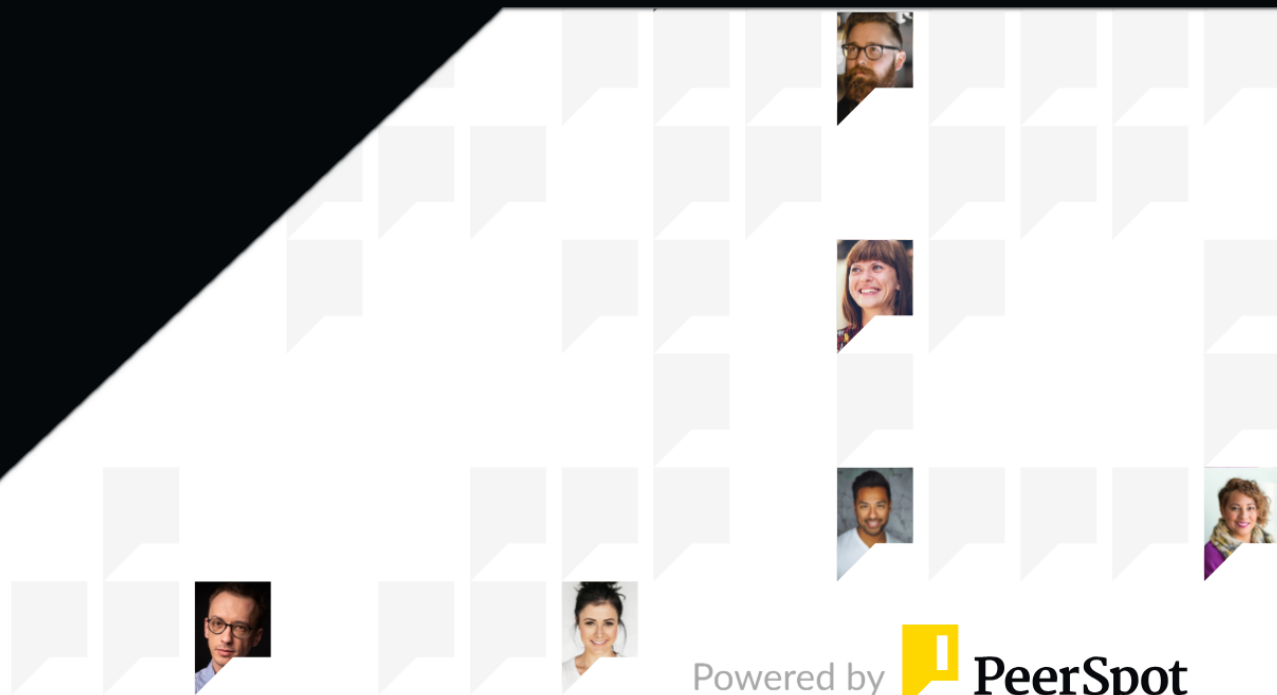




Redgate Flyway

Reviews, tips, and advice from real users



Powered by  **PeerSpot**

Contents

Product Recap..... 3 - 4

Valuable Features..... 5 - 13

Other Solutions Considered..... 14 - 15

ROI..... 16 - 18

Use Case..... 19 - 23

Setup..... 24

Customer Service and Support..... 25

Other Advice..... 26 - 30

Trends..... 31 - 32

About PeerSpot..... 33 - 34

Product Recap



Redgate Flyway

Redgate Flyway Recap

Redgate Flyway delivers agile database migrations, optimizing version control and deployment workflow. Its approach is designed to align efficiently with DevOps practices, enabling teams to handle diverse database environments with precision and speed.

Redgate Flyway offers robust migration tools adapted for flexible database configurations, integrating seamlessly into continuous delivery pipelines. Known for its consistent performance, Flyway simplifies complex database transformations, ensuring dependable version control. This compatibility encourages swift adaptation in changing development landscapes, making it integral to modern database management strategies.

What are the key features of Redgate Flyway?

- **Versioned Migrations:** Maintain clear version history with scripts ensuring predictable results.
- **Repeatable Migrations:** Run specific scripts recurrently until successful execution.
- **Baseline:** Establish a baseline version for consistent future migrations.
- **Repair:** Detect and correct discrepancies in the schema history.

What benefits should users examine in reviews?

- **Scalability:** Adaptive management of increasing database sizes and complexities.
- **Efficiency:** Streamlined processes reduce deployment times and errors.
- **Consistency:** Reliable execution to maintain database integrity.
- **Flexibility:** Compatible with multiple database platforms.

In industries like finance and e-commerce, Redgate Flyway supports rapid deployment, ensuring compliance with stringent data regulations while maintaining high performance. Its adaptability also benefits tech startups managing frequent schema changes within tight deadlines.

Valuable Features

Excerpts from real customer reviews on PeerSpot:



“Now using Redgate Flyway, we do not need to do any of this procedure, so this whole process has been eliminated, almost a day is saved, and on release day there are 99% chances that the scripts will run successfully without issues.”



Hassan F

Full Stack Developer at DPL



“Redgate Flyway has positively impacted my organization by enabling automated database migrations, so there are no manual updates to databases on each deployment to each environment.”



Martin Munez

Business Development Manager at Ford Motor Company



“Redgate Flyway has positively impacted my organization by maintaining consistency among developers.”



Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees



“I have seen a return on investment with Redgate Flyway in that time was saved significantly for local development.”



Verified user

Senior Software Engineer at a comms service provider with 51-200 employees



“Redgate Flyway streamlines the process of database migration and significantly reduces the time for data migration from twenty to twenty-five days down to five to seven days, thanks to the initial time we took for learning.”



ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees



“Redgate Flyway has impacted us very positively because we don't have to manually cover all those things.”



Verified user

Software Engineer at a tech vendor with 11-50 employees

What users had to say about valuable features:

“The best features Redgate Flyway offers include protecting our database so that not everybody can randomly change it. Additionally, database migration, database migration events, management, and managing database objects as code objects in version control are excellent.

“Redgate Flyway has positively impacted my organization by enabling automated database migrations, so there are no manual updates to databases on each deployment to each environment. Team onboarding is easier as any developer can get a local database working in minimal time, thanks to database migration from scratch..”

Martin Munez

Business Development Manager at Ford Motor Company

[Read full review](#) 

“The best features that Redgate Flyway offers include support from the command prompt, which is the greatest advantage I have. Command prompt support in Redgate Flyway helps me because the integration with CI/CD happens seamlessly with the help of the command prompt, and for our local migration, we also use the command prompt. Because we use local Docker instances to maintain our Postgres databases, the command prompt is helping significantly.

An additional feature of Redgate Flyway is the flyway_schema_history table that is created whenever migration happens. That table helps a great deal to understand in which part of the migration a particular migration has failed previously and serves useful debug purposes..”

Verified user

Senior Software Engineer at a comms service provider with 51-200 employees

[Read full review](#) 

“Redgate Flyway provides versioning as a primary benefit, allowing us to have numbered versions and easily switch from one version to another. We write the script once, and it performs most of the manual tasks such as version control and rollback in case of any errors independently.

Redgate Flyway's version control of database changes has significantly helped in ensuring repeatable deployments. As we continuously develop and migrate schema changes while verifying that the migration is working correctly, we progressively create new versions and keep our front-end and API teams informed of changes.

Redgate Flyway streamlines the process of database migration and significantly reduces the time for data migration from twenty to twenty-five days down to five to seven days, thanks to the initial time we took for learning. Now we can complete migrations in two or three days, which is a substantial impact.

Redgate Flyway has played a critical role in accelerating our software development cycle by helping us implement continuous deployment through the creation of different schema versions and their sequential deployment, which would be very difficult with traditional methods..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

“The best features Redgate Flyway offers are its usefulness when working with other people, as it maintains consistency in database changes, and it is helpful when shifting databases because Redgate Flyway provides the migration script that can be run in different environments without needing to rely on Spring auto updates to create those tables. It is a well-structured tool for managing table creations.

Managing migrations across different environments has been quite easy for my team to adapt to and use Redgate Flyway without any challenges. We recently moved from a Docker database to an actual database in a different environment, and it was easy for us to share the migrations. Once they were done running them, we shifted connectivity to their environment and everything worked without issues.

Redgate Flyway has positively impacted my organization by maintaining consistency among developers. One real added feature is using migration tools, which allows rollbacks although I think rollback has some bottlenecks. I acknowledge that Spring Boot does not have a migration tool by default unlike Entity Framework in C#, but having Redgate Flyway is becoming a default tool for migration in Spring..”

Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees

[Read full review](#) 

“The best features Redgate Flyway offers include ease of use. The second is that it is simple with not a lot of dependency required. Third is the version controlling mechanism.

Regarding the link mechanism, when we execute Redgate Flyway, it generates a hash value. It keeps that hash value, and when the next migration comes into existence, it links that particular previous value and executes it, which shows how the ease of use has benefited me and my team.

Redgate Flyway has impacted us very positively because we don't have to manually cover all those things. The one main improvement that we have made is that previously we were not using Redgate Flyway, so we had to manually migrate the scripts from one command to another command with the same versioning. Now we don't need to configure all those things. We let Redgate Flyway handle all those things and we need to write the SQL command in those particular files with the correct versioning.

It definitely saves a lot of time because we don't need to write manual scripts in all those environments. You keep that in your local, and when you push that particular Redgate Flyway script or your code to the production environment, it is automatically executed because it keeps tracking the previous version. This was a great mechanism that Redgate Flyway follows. Also, it manually reduces a lot of errors because when we write the SQL command and if we want to alter any table or change the database schema in a specific order, Redgate Flyway always takes care of that..”

Verified user

Software Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

“The best features that Redgate Flyway offers, if I had to pick a few that really stand out, would be multi-environment support. On the migrations tab, I do not need to go to an environment and change settings or anything. I simply change the branches of the environment and it shows me what is available and what has been run on a certain environment. The environment feature is very user-friendly and helpful, so I would keep it at the top of my list. The feature of changing branches on the migrations tab is very helpful.

An example of how Redgate Flyway specifically helped with discrepancies is that previously we did not have any tool recording database changes. We work on an Agile Scrum pattern, so we have to do deployments frequently, within every two to three weeks or sometimes four weeks. Previously, we had code repositories for front-end and back-end, but for the database side, we did not have any repository. We were not saving database-related changes in any GitHub or AWS CodeCommit repositories. Every time, we have a Jira board where developers update their scripts. For example, if I work on a ticket and update a stored procedure, I must mention the stored procedure on the ticket. When deployment time arrives, the release manager must pull out or scan all the tickets and extract the objects. For example, if we deploy 10 Jira tickets from a sprint in the next release, we must go through all 10 tickets and see the post-deployments of their tickets. Then we extracted the objects from the development environment, deployed on stage, and then deployed on production. In this scenario, many objects and discrepancies occurred. Sometimes a developer or the release manager would forget the object to take to production. Now, after using Redgate Flyway, I have restricted access as the release manager of my team. I manage the release for my team and have restricted developer access to environments other than the development environment. If developers want to take anything to the next environment such as demo, staging, or production, they must make a script. When they create a script, it is in our record. Now, after using Redgate Flyway, we do not need to scan all the tickets on Jira or see the post-deployments of each ticket. We simply view the Redgate Flyway script showing what has been run from this to this version, and what pending deployments need to be run on production. In this way, it has helped tremendously.

I can share that the migrations tab and branch changing helped my team in a specific situation during our second last sprint. Two developers were working on the same object, and one change needed to be deployed on stage while another change needed to be deployed on the demo environment, which is our QA level. Our QA and demo are the same environment, and then we have stage and production. We have three environments other than development. Previously, without Redgate Flyway, what could have happened is that we would take the stored procedure from demo if we needed to deploy it on stage and take it directly to stage. This was our previous practice where we would go to the database explorer, take the stored procedure, and move it to the next environment with the ticket. Now with Redgate Flyway, we have different versions of that stored procedure. We simply took the version of the stored procedure that needed to be on stage, and the second version that needed to be on the QA level remained there. Redgate Flyway helped in this case, and we have many cases..”

Hassan F

Full Stack Developer at DPL

[Read full review](#) 

Other Solutions Considered

“Some teams were using different migration tools previously, but I do not know their names exactly. Since we migrated to a newer version of SQL, those tools were not working efficiently..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

“I previously used Liquibase and switched to Redgate Flyway because I had to follow the company's requirements, as most companies I have worked with use Redgate Flyway..”

Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees

[Read full review](#) 

“Before choosing Redgate Flyway, we evaluated options such as Liquibase and Alembic, among others.

We decided to go with Redgate Flyway because it uses plain SQL scripts without complex XML or YAML, has version control for database changes, and works well with CI/CD. Although the cons include the absence of automatic rollback and potential merge conflicts, the versioning control was the major factor for our decision..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

ROI

Real user quotes about their ROI:

“I have seen a return on investment as the migration event subscriptions allow my team to monitor production deployments more carefully and detect errors before deploying breaking changes to a productive database..”

Martin Munez

Business Development Manager at Ford Motor Company

[Read full review](#) 

“I see a return on investment with Redgate Flyway, as it is leading compared to Liquibase and others, and most teams I have worked with utilize Redgate Flyway, which is definitely pointing in the right direction..”

Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees

[Read full review](#) 

“I have seen a return on investment with Redgate Flyway in that time was saved significantly for local development. Even for local development, we had to maintain the different versions, so I would definitely say that time was saved. For example, a normal process would have taken one day, but the migration script made it possible within twenty to thirty minutes..”

Verified user

Senior Software Engineer at a comms service provider with 51-200 employees

[Read full review](#) 

“I can share specific metrics regarding how much time was saved during releases since using Redgate Flyway. During our normal release, there are around 20 to 30 Jira tickets that we take to production, typically representing two sprints or sometimes three sprints. Previously, the release manager and developers had to sit together and check all the post-deployments of these tickets, then extract their object names into an Excel file or some other file. They created a filter file to filter out stored procedures from the staging environment and pulled all objects such as stored procedures, tables, and alter commands. They kept everything on a separate file and consolidated it. On deployment day, they ran it on the production environment. The extraction of objects, filtering from the database, creating a file, maintaining an Excel file, and communicating with each developer used to take around four to five hours or more, and could sometimes take a whole day for developers and the release manager as well, especially if a developer questioned whether a change was theirs. Now using Redgate Flyway, we do not need to do any of this procedure, so this whole process has been eliminated. Almost a day is saved through Redgate Flyway. On release day, when the release database scripts run using Redgate Flyway, there are 99% chances that the scripts will run successfully without issues. Over the last seven to eight months, we have not faced any issues and the client has praised us. Previously, it was a normal practice to take the script to the production environment and run it, and it was normal to have errors including syntax errors or object errors. We would fix these errors and two to three hours or sometimes four hours of developer time would be spent on release day. Overall, around 10 to 15 hours have been saved from using Redgate Flyway, along with reduced human effort..”

Hassan F

Full Stack Developer at DPL

[Read full review](#) 

Use Case

“My main use case for Redgate Flyway is for migration and migration of scripts, primarily to maintain our databases. I do not have anything unique to add about my use case with Redgate Flyway; it is almost the same as other standard use cases..”

Verified user

Senior Software Engineer at a comms service provider with 51-200 employees

[Read full review](#) 

“Redgate Flyway is a migration tool that we use for our SQL migration. We have several customers' databases that must be migrated to the new schema, and we use Redgate Flyway scripts to migrate from the older schema to the newer schema. This approach helps us keep different versions of the schema without losing data simultaneously..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

“Redgate Flyway is my primary tool for database migrations, especially for solutions based on the Java programming language.

“A specific example of how I use Redgate Flyway for database migrations in a Java-based solution is that it has an intuitive way to operate within the Java ecosystem. It has multiple functions and events to monitor migration progress, making it very useful for solutions that have several migrations with breaking changes. In a short period of time, I can subscribe to each event during the migration and observe the entire progress, detect any problems, and add behaviors at any time.

“I use Redgate Flyway in every single environment. In my opinion, it is a very good tool for every scenario and is not restricted to development environments only..”

Martin Munez

Business Development Manager at Ford Motor Company

[Read full review](#) 

“My main use case for Redgate Flyway is that it is a migration version control tool. I prefer Redgate Flyway because instead of manually running SQL scripts on different environments like dev, QA, and production, Redgate Flyway helps automate this process. It tracks which scripts have already been executed and only runs new migration scripts. It ensures every environment has the same database schema, which is very helpful for all the people on the team. It also keeps a history of database changes.

In my first company, I used Redgate Flyway for a project where we were integrating MakeMyTrip APIs. In that particular project, we had to create a lot of tables, and I generally used Redgate Flyway for that. The main purpose of using it was that we had to make a lot of schema changes. Instead of doing it manually in the PostgreSQL database, we can use Redgate Flyway. What I did was version the file name, then provide the SQL language in that file, and run it. It keeps executing from there..”

Verified user

Software Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

“My main use case for Redgate Flyway is to maintain the database structure, especially when coding with other developers, and I view it as a version control system for databases, similar to Git.

A specific example of how I have used Redgate Flyway to maintain the database structure is in starting out a project, where I have used two flavors of it, writing raw SQL queries and also using the Java class version of SQL queries. I started from scratch because we needed to ensure that other teams see the update of the database since we are all coding differently. Once you pull, you notice there is a new migration and have to run that migration to maintain the update of the database. We built it from scratch and used the raw SQL queries.

Redgate Flyway is used at every level of the project. If there is a new column added, I have to update the SQL query migration. If there is any constraint change, table creation, or indexing to add, I have to include that in the Redgate Flyway migration. Once that is pushed, our database team gets a notification that the tables or the database have a new entry, and they also do reviews. Once anybody pulls the code, that person also gets the new migration change..”

Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees

[Read full review](#) 

“The main use case for Redgate Flyway is that previously we had to manually manage our different environment database scripts. We are using a .NET application with a Dapper back-end, and we have numerous SQL objects including stored procedures, views, table scripts, and DDL and DML scripts each time we deploy. We needed to manage scripts for the development environment, then for the staging environment, then for production, and that caused too much discrepancy and human effort during the release. After learning about Redgate Flyway, we started using it and have managed multiple environments within it. Redgate Flyway provides support for different branches, allowing us to manage our scripts with each script recorded with a version number. This has reduced the discrepancy in our deployments and accelerated our procedure significantly. The human effort required has become considerably less.

Regarding how Redgate Flyway fits into my workflow, as a team, we always wanted to automate or record everything we do and maintain transparency. From the database side, we needed a tool that transparently shows what developers have done and what we will take to production deployment. Redgate Flyway aligned with our needs in this way. Our next target is to create pipelines for Redgate Flyway, but it is currently working well for us. We have set up different environments in Redgate Flyway and it is functioning properly. We have a repository on GitHub and we push and pull from there..”

Hassan F

Full Stack Developer at DPL

[Read full review](#) 

Setup

The setup process involves configuring and preparing the product or service for use, which may include tasks such as installation, account creation, initial configuration, and troubleshooting any issues that may arise. Below you can find real user quotes about the setup process.

“The initial setup of Redgate Flyway was challenging because we had to integrate a database without any history and establish the initial version. This was difficult due to multiple tables and the need for environment setup, alongside the team needing to learn it, and version conflicts arising when multiple people were working..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

“Regarding maintaining consistency and rollbacks with Redgate Flyway, I have luckily not had many issues. In the past, I sometimes had to delete all migrations trying to recreate them. I do not think it is really smooth when you have already started the project and want to now introduce Redgate Flyway halfway through. While I cannot remember the exact scenario, I know it was not smooth. It is usually better to use it from the beginning of the project, though it is a bit more complicated in between projects..”

Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees

[Read full review](#) 

Customer Service and Support

“We generally use resources such as Stack Overflow for support and have not escalated any questions to technical support during our database migrations with Redgate Flyway..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

Other Advice

“My advice for organizations considering Redgate Flyway is to look for agentic AI support for regular tasks, particularly in writing basic scripts, as such assistance could be very beneficial. I rate Redgate Flyway overall as an eight as a product and a solution..”

ShivamJaiswal

Software Development Engineer 2 at a tech vendor with 10,001+ employees

[Read full review](#) 

“My advice to others looking into using Redgate Flyway is to keep track of the versioning because it is very important in Redgate Flyway. After that, you are all set. As a software engineer who is using Redgate Flyway, I can conclude that it is a really good tool. I gave this review a rating of nine out of ten..”

Verified user

Software Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

“Redgate Flyway is a tool that works, and I have not had many issues with it. My advice to others looking into using Redgate Flyway is that it is a good tool, especially when the development team is large and you want to maintain database integrity, consistent updates, and also handle migrations, as Redgate Flyway has become a default tool to use.

Regarding Redgate Flyway's artificial intelligence capabilities, I have not really examined artificial intelligence with Redgate Flyway. Security is a general issue, but I have not faced any security issues with Redgate Flyway and have not looked into Redgate Flyway's artificial intelligence capabilities. I have not actually looked at it or its accuracy and reliability of output recently, but it is something I would like to explore..”

Zurum Ogbonda

Senior Java Software Engineer at a comms service provider with 10,001+ employees

[Read full review](#) 

“My advice to others looking into using Redgate Flyway is to focus on database migrations for Java-based solutions. It has one of the best integrations with Java as it is a database migration tool. You can do it with the community edition, which has no pricing involved, and it works flawlessly with Maven and Gradle. It is not an expensive tool to use the next level of features, and it is worth the money. I would recommend reaching that edition level, as the object mapping feature becomes really handy.

“I would not recommend Redgate Flyway for any Microsoft-based solutions or .NET, as it is not compatible at all based on my experience. This tool is only worthwhile with Java applications.

“I would rate this product a nine out of ten..”

Martin Munez

Business Development Manager at Ford Motor Company

[Read full review](#) 

“I would say I have never used Redgate Flyway in production; we only use it in our local test instances or some QA environments where we test it. I have never worried about the performance or much else, but it is definitely saving us a lot of time. To maintain the local database is not that easy, and to maintain the different instances of the same version of the database is quite hard where multiple teams handle different instances, so I think it helps a great deal.

Currently, I deploy Redgate Flyway only to maintain our local test instance using a [Docker](#) setup; apart from that, we are not using it anywhere. I have only used Redgate Flyway with Oracle and Postgres, and for these two, it has worked pretty much fine until now.

My advice to others looking into using Redgate Flyway is to try to be as straightforward as possible. Do not try to do unnecessary changes on the databases when you are using Flyway Migrate because that would unnecessarily corrupt your database in the eyes of Flyway Migrate. The database is actually pretty much fine, and Flyway Migrate assumes that it is corrupted, which causes it to fail the migration and ask you to repair it, which in turn will create multiple problems.

Verified user

Senior Software Engineer at a comms service provider with 51-200 employees

[Read full review](#) 

“There is nothing specific to add regarding needed improvements.

The advice I would give to others looking into using Redgate Flyway is to integrate it properly through pipelines and manage it properly. If you are using Redgate Flyway, you must not update any objects or scripts manually so that you can achieve 100% of what Redgate Flyway can provide.

Regarding additional thoughts about Redgate Flyway before we conclude, I cannot think of anything right now. Everything is working well currently.

I would rate Redgate Flyway overall as an eight on a scale of one to ten.

The reason I chose an eight rating is that Redgate Flyway is providing tremendous support and helping in different scenarios, but there could be some improvements. For example, if we want to attach it through code pipelines or connect it with our code, something could be added. Currently, we are using Dapper, but if we go to Entity Framework Core, there could be some connection. We have a connection with our entire system including the front-end, the back-end, the repository, and the [AWS](#) infrastructure, but Redgate Flyway seems to be an isolated system in itself. It is not connected with anything else and is just for the database. If it could connect with other tools as well, that would be beneficial. However, this is just my opinion and I cannot comment on the science behind it.

Regarding Redgate Flyway's security and governance, they are normal. We have other tools and applications that are enterprise-grade providing general security and governance. I cannot say that it is highly secure, but the features are good. The security aspect, where we have to log in through our [SSO](#) and corporate emails, is quite better. It is similar to other tools we are using, such as GitHub and [AWS](#). The intelligence of Redgate Flyway is quite better, but there is nothing such as bots working for me. I would not say that this is great in AI or artificial intelligence..”

Hassan F

Full Stack Developer at DPL

[Read full review](#) 

Top Industries

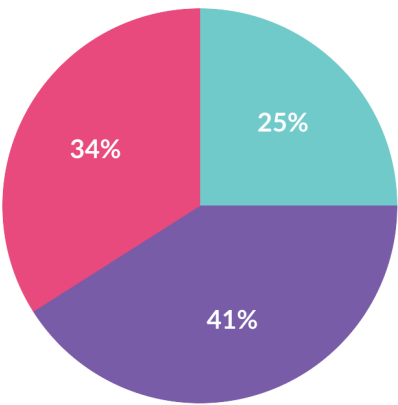
by visitors reading reviews



Company Size

by reviewers

by visitors reading reviews



Large Enterprise Midsized Enterprise Small Business

About this buyer's guide

Thanks for downloading this PeerSpot report.

The summaries, overviews and recaps in this report are all based on real user feedback and reviews collected by PeerSpot's team. Every reviewer on PeerSpot has been authenticated with our triple authentication process. This is done to ensure that every review provided is an unbiased review from a real user.

Get a custom version of this report... Personalized for you!

Please note that this is a generic report based on reviews and opinions from the collective PeerSpot community. We offer a [customized report](#) of solutions recommended for you based on:

- Your industry
- Company size
- Which solutions you're already considering

The customized report will include recommendations for you based on what other people like you are using and researching.

Answer a few questions in our short wizard to get your customized report.

[Get your personalized report here](#)

About PeerSpot

PeerSpot is the leading review site for cloud, AI, and business software. We created PeerSpot to provide a trusted platform to share information about software, applications, and services. Since 2012, over 22 million people have used PeerSpot to choose the right software for their business.

PeerSpot helps tech professionals by providing:

- A list of products recommended by real users
- In-depth reviews, including pros and cons
- Specific information to help you choose the best vendor for your needs

Use PeerSpot to:

- Read and post reviews of products
- Access over 30,000 buyer's guides and comparison reports
- Request or share information about functionality, quality, and pricing

Join PeerSpot to connect with peers to help you:

- Get immediate answers to questions
- Validate vendor claims
- Exchange tips for getting the best deals with vendor

Visit PeerSpot: www.peerspot.com

PeerSpot

244 5th Avenue, Suite R-230 • New York, NY 10001

reports@peerspot.com

+1 646.328.1944