



CASE STUDY

Streamlining database deployments with Redgate



BGL Insurance chose Redgate to introduce DevOps to its database development process. The move resolved many deployment issues by establishing the same kind of CI and CD pipelines that were in use in application development.

Contents

The Customer

In partnership with several of the best-known brands in UK financial services and through its own brands, BGL Insurance provides a range of insurance products, supported by state-of-the-art, award winning contact centers and market leading digital platforms.

The Challenge

Maintaining and updating the databases that support the huge variety of insurance services and products offered was an ongoing challenge for the Data Solutions Team, with numerous out-of-hours calls to resolve problems with releases.

The Solution

The team introduced DevOps to the database development process and set up the same kind of Continuous Integration and Continuous Deployment pipelines that were in use in application development.

The Results

Automating the release process has made it faster and much more reliable, and moving development issues further to the left in the development pipeline has saved time and energy for support staff.

“There were numerous out of hours calls and situations where a deployment overwrote someone else’s previous deployment.”

Luke Rust, Senior Data Engineer

The Customer

A leading digital distributor of insurance to more than three million customers, BGL Insurance provides a range of motor, home and life insurance products in partnership with several of the best-known brands in UK financial services and through own brands Budget Insurance, Dial Direct and Beagle Street, all supported by state-of-the-art, award winning contact centers and market leading digital platforms.

With almost 2,000 colleagues across three sites in Peterborough, Sunderland and Wakefield, BGL Insurance uses its passion, scale and expertise for the benefit of its customers, offering unrivalled value, ease and confidence.

The success of BGL Insurance is, in many ways, down to its advanced software development practices. The ongoing goal of its 250-strong IT team, BGL Tech, is to challenge, create, deliver and evolve its services and products. The team is responsible for developing and supporting BGL’s contact centers and market leading digital platforms, as well as providing tech services to other business functions such as Information Security and IT Operations.

Luke Rust is the Senior Data Engineer for the Data Solutions Team in BGL Tech which manages, maintains and updates over 70 SQL Server databases across the different development and production environments.



In addition to this, the team of 12 Data Engineers extract insurance quote and policy data from the core management system and deliver business-critical data sets and reporting to a wide range of data experts around the business, including BI teams, Data Scientists and partners.

The reports are key to BGL Insurance being able to make fast decisions about the ever-changing conditions that affect insurance policies, quotes and product comparisons in order to maintain a competitive advantage. As a consequence, internal stakeholders need the confidence that the data is updated and provided quickly, regularly, accurately and on time. Any delays in processing and providing the data because of database deployment errors has a knock-on effect in other areas of the business.

“Releases are now deployed the same day they are signed off, normally within ten minutes of approval, with no possibility of overwriting code.”

Luke Rust, Senior Data Engineer

The Challenge

As might be imagined, the huge variety of services and products offered by the many financial services companies BGL Insurance partners with are developed and improved constantly. Alongside this, there are data points coming in, changing and being revised all of the time. This in turn makes maintaining and updating the databases that support and serve them an ongoing challenge for the Data Solutions Team.

To handle this, the team were performing manual database deployments using hand-rolled migration scripts, along with rollback scripts in case they were required. Releases typically occurred during working hours from Monday to Friday and there were numerous out-of-hours calls with Data Engineers having to return to work in the early hours to create fixes.

Quality checks were also difficult to validate and this often led to situations where deployments overwrote previous database changes. All of this meant that deploying code required a buddy system with developers overseeing each other, which was a huge strain on time and resources.



To resolve the issues, Luke Rust and his team were tasked with introducing DevOps to the database development process and setting up the same kind of Continuous Integration and Continuous Deployment pipelines that were in use in application development. As part of this, a due diligence review was undertaken of the three major database development vendors.

Redgate was chosen because its solution was more comprehensive, it was easier to get the team up-to-speed with the technology, and a wealth of training materials from Redgate was available. The team were also already familiar with Redgate's broad portfolio of database development tools and the superior level of support that was offered.

“It’s saved out-of-hour callouts significantly, along with reducing delays to producing business critical data to data experts around BGL.”

Luke Rust, Senior Data Engineer

The Solution

In order to get the team on board with the new development and deployment process, the project to introduce database DevOps was planned in two stages.

The first stage was to introduce Redgate version control tooling to the database code, which provides an object level history of all the changes being implemented. The team also improved the reliability of manual deployments by adopting Redgate’s database comparison tool, the industry standard for comparing and deploying SQL Server database schemas quickly and accurately. This involved moving from shared to dedicated development environments, with each developer having their own database copy to develop and test changes against.

Once the team were comfortable using version control, the next step was to integrate database changes into the CI/CD pipelines. Redgate database deployment technology was plugged into the existing TeamCity, Octopus Deploy and Nexus Repository Manager pipeline. This triggers a CI process to build, test and validate changes whenever they are checked into version control, before automatically deploying the release to relevant environments.



Releases can now be deployed the same day they have been signed off by pull requests, normally within ten minutes of approval being given. This is only limited by the time it takes to merge, build and do the release checks. This is an important step for compliance because, as Luke comments:

"It also provides a complete log of changes, a history of every release, and a record of every script that was created to make a change and what the outcome was."

While there are still manual intervention steps in place to ensure the correct changes are being deployed, the team aim to remove some of these once the software has fully bedded in.



The Results

The Data Solutions Team has seen a number of major – and timely – improvements in the way they work. As Luke comments:

“The buddy system has been dropped, removing a significant drain on resources, the number of failed deployments has fallen markedly, and the quality of development has risen, even with a 30% increase in staff.”

Releases are now much faster because instead of manually opening each migration file to check it, waiting for it to run and then running the next file, it is all handled by the CD software. Failing TeamCity builds have also highlighted code issues before they were deployed to production, which has saved out-of-hour callouts significantly, along with delays to producing reports for the MI and BI teams. Working on the same code base is no longer an issue and there have been at least three occasions where the new process has prevented code overwriting someone else's work.

The team have also managed to move development issues further to the left in the development pipeline, saving time and energy for support staff. Before a change has a pull request created, it must have a valid TeamCity build. This additional quality check has meant problems are picked up earlier and are not deployed to production.



Importantly, the added visibility of changes has allowed the team to scale up far easier. As Luke concludes:

"The team has nearly doubled in size over the last few years and it would have been difficult to integrate all of the changes the Data Engineers now make without the additional software."

The next step is to introduce Redgate's on-demand database cloning capability to the process, which creates full copies of SQL Server databases and backups in seconds, using only around 40 MB of disk space per clone. This will save significant time in the provisioning of dedicated development environments for members of the team and enable them to take full advantage of Redgate's solution.