CASE STUDY | HAYS TRAVEL

Introducing true team development with SQL Toolbelt Essentials

“We were having probably one big deployment incident a week at the worst point.”

As the Group Business Analytics & Reporting Manager at Hays Travel, Paul Cresham has a challenging role. With an annual turnover of £800 million (~US$1.07 billion), Hays Travel is the UK’s largest independent travel agent and books thousands of holidays through hundreds of tour operators every week.

Behind the scenes, Paul leads a small team of application and database developers who are responsible for the in-house applications which control the many different elements of the business, and the enterprise data warehouse and related databases which store and process the huge amount of ever-changing data the company generates.

This is a crucial part of the business because they support processes such as the supply and payments channel, and the applications the sales people access in stores when speaking with customers.

Using a typical Microsoft stack, with a mixture of Visual Studio, SQL Server Management Studio (SSMS) and Team Foundation Server (TFS), the team works in a DevOps style, sometimes developing the applications and databases side by side, other times working independently.

Because much of the data is stored in the enterprise data warehouse, the team uses a shared development approach, moving changes from a Test environment to the Live environment when necessary.

As might be expected, with the databases being updated regularly and a constant stream of new features, fixes, and general development work, the variety of demands on Paul’s team prevents them having a regular release schedule.

In one way, this encourages collaboration within the team to gain an understanding of what everyone is working on at any given time. In another way, however, it was causing problems because, while the team was skilled at moving changes from the Test to the Live environment quickly, there was no control over those changes, and some were made directly to the databases in the Live environment to meet urgent deadlines.

This inevitably led to problems occurring with failed deployments which, in turn, resulted in the business operations of Hays Travel being affected, often on a weekly basis. At this point, it was recognized that the development process had to change.

“Everything on our front side was source controlled with TFS, but nothing at all on the SQL side.”

Paul Cresham and his team knew they had to refine their database development practices, as did the management of Hays Travel itself. While making the best use of the resources they had, the failed deployments were a result of hot fixes being made to databases in the Live environment, as well as changes not being migrated properly from the Test environment because something was missed.

Paul recognized that the root cause of the problems being encountered was the way changes to the databases were being deployed. While he actively encouraged the developers to write database migration scripts by hand because it gave more control over deployments, the multiple streams of development work meant there was no control over the different changes going through.

This was compounded by hot fixes being made to the Live environment. Because they weren’t replicated in the Test environment, there was no way of keeping track of the changes, or knowing what the differences were between the two. So when a change was deployed, it would sometimes fail because the environment it had been tested in didn’t match the environment it was being deployed to.

Paul already understood that part of the solution was to version control database development changes in the same way the team was version controlling application development with TFS. That way, when changes were committed to version control, everyone in the team would be working on the same codebase. He tried the TFS integration in the new version of SSMS and discovered it wasn’t user-friendly, which discouraged developers from adopting it.

His second challenge was being able to identify accurately the precise differences between the databases in the Test environment and the Live environment. Only by knowing this would the team be able to confidently deploy changes and keep the business of Hays Travel on track.

Having already heard of tools from Redgate like SQL Source Control and SQL Compare, he looked at SQL Toolbelt Essentials, which includes these tools as well as seven others.

> “It’s improved the speed at which the team develops significantly, and saved about half a day a week.”

The main goals of introducing SQL Toolbelt Essentials were to version control database changes, properly manage the migrations between the Test and Live environments, identify the exact differences between those environments to avoid failed deployments and, ultimately, create a more transparent release management process.

When the team trialed SQL Toolbelt Essentials, they were instantly sold on how it could help them meet those ambitions.

Because SQL Source Control plugs into and works inside SSMS, it was easy to start committing and sharing code changes across the team without needing to alter the way they worked. With full support for TFS, it also meant that application and database development could be version controlled alongside each other.

Similarly, the introduction of SQL Compare enabled the database schemas in the Test and Live environments to be compared down to individual lines of SQL. Its ability to create deployment scripts with every dependency listed in the right order also, importantly, gave the team confidence that they could deploy database changes without errors or fears of downtime.

The team also discovered they could use SQL Data Compare to track the static reference data in the Live environment, which slowly changes over time, and use it to update the Test environment to ensure every element in both is in sync.

> Redgate’s SQL Toolbelt has many fantastic features, with SQL Prompt being the most useful to me. I’ve saved so much time altering procedures, tables and views using the shortcuts, rather than having to find in the object explorer and then modify. Having a dialog box pop up when executing insert / update / delete statements without a where clause is a lifesaver!”
>
> **Robert Calvert**

There were two unexpected surprises waiting for the team inside SQL Toolbelt Essentials as well.

SQL Search is a small but very useful add-in for SSMS and Visual Studio that the team have found invaluable because it lets them search for SQL across databases and find everything that references it in seconds.

SQL Prompt, meanwhile, is the advanced IntelliSense-style code completion tool that lets users write, format, share and refactor SQL effortlessly. It has dramatically improved the pace at which the team develop, helped to standardize processes and, by taking the chore out of writing SQL, made database development far easier.

Together, the tools within SQL Toolbelt Essentials have also speeded up the team’s development time significantly, while increasing confidence in the quality of that development.

“The next step will be moving towards automated release management by using the whole SQL Toolbelt.”

Introducing SQL Toolbelt Essentials has enabled Paul and his team at Hays Travel to resolve every deployment issue they were facing. Importantly, it has also given the team new energy in the way they approach database development.

> Redgate’s SQL Toolbelt Essentials has been a hit with the team and has had a huge positive impact on how we work. SQL Prompt takes the chore out of many tasks, by prompting for linked server databases, automatically completing insert statements, and my favorite: automatically creating your GROUP BY clauses. SQL Search makes finding dependencies a breeze, and SQL Compare is used to manage the differences between test and live environments, and also deploy changes at the press of a button, removing the need in most cases to manage (often multiple) migration scripts.”
>
> **Paul Cresham**

Most database deployments can now be handled with the migration scripts produced by SQL Compare and are streamlined and error-free. Where a more complex migration script is required, Paul uses SQL Source Control to produce the script, which he can then review and update as required.

But while SQL Source Control and SQL Compare have been the tools that have introduced the processes required to safely develop and deploy database changes, SQL Prompt and SQL Search are called out by every member of the team as the tools that offer the most value in their everyday work.

SQL Doc is the next tool Paul wants to introduce because, like any development team, getting people to document databases is always a challenge. SQL Doc eliminates manual documentation tasks, satisfies audit requirements and, importantly, does so in a couple of clicks from within SSMS.

Paul is also now looking towards his longer term goal of introducing automated release management. His team is growing and, having seen the huge difference the version control and comparison tools have made, he is already considering the advantages of introducing the whole SQL Toolbelt.

The additional tools it offers like DLM Automation and ReadyRoll will enable the team to Include the databases in automated processes like continuous integration, alongside the applications, and set up a reliable, repeatable deployment process for fast, frequent database updates.

Everything you need to work with SQL Server

The SQL Toolbelt Essentials gives developers and DBAs nine essential SQL Server tools in one installer. This means that when you are working with SQL Server databases, you can finally have **the most powerful set of tools right there when you need them**.

These are the nine essential SQL Server tools to get you started:

SQL Compare

Save time comparing and deploying SQL Server database schemas with SQL Compare. Find and fix errors, and create deployment scripts in minutes.

SQL Data Compare

SQL Data Compare is the industry-standard tool for comparing and deploying SQL Server database contents.

SQL Source Control

Connect your databases to your source control system

SQL Prompt

An add-in for SQL Server Management Studio and Visual Studio, SQL Prompt strips away the repetition of coding.

SQL Doc

Generates the latest database documentation automatically as part of your CI process.

SQL Test

Allows developers to easily create, run, and manage tSQLt unit tests on databases.

SQL Data Generator

Generates realistic test data based on your existing schema.

DLM Dashboard

DLM Dashboard tracks your database schemas and alerts you when they change.

SQL Search

An add-in for SQL Server Management Studio and Visual Studio, SQL Search lets you find fragments of SQL instantly.

Try the SQL Toolbelt free: www.red-gate.com/sql-toolbelt