InBrief

Daniel Howard – Senior Researcher

Redgate SQL Provision

The company

Redgate provides software that enables what it calls *'Compliant Database DevOps'*, meaning that it helps you to leverage your data inside a DevOps toolchain in a way that is compliant with all applicable regulations and policies. The majority of Redgate's products work exclusively with SQL Server, although several are also available for Oracle and MySQL databases. Moreover, many Redgate products support Azure and AWS environments in addition to on-premises SQL Server. In 2017, Redgate acquired Net2000, a data masking vendor, along with their products DataBee (now discontinued) and Data Masker.

The company was founded in 1999 and is based in Cambridge, UK. It has additional offices in the US and Australia. Redgate's customer base numbers over 202,000 and includes 91% of the Fortune 100. It is also a gold partner with both Microsoft and Oracle.

> " SQL Provision has given us the ability to mask data and push it out to multiple locations almost instantly. That saves hours compared to the way we used to refresh. "
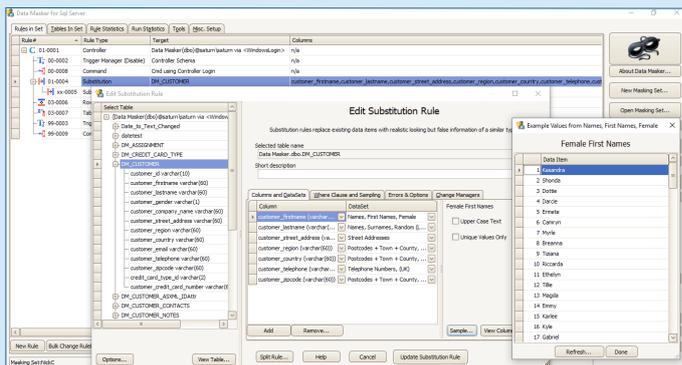>
> **KEPRO**

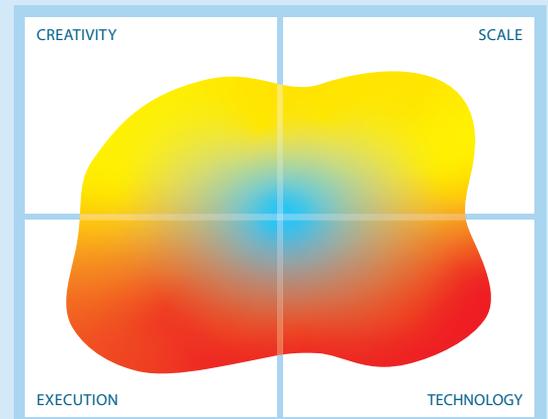*Figure 1 – Editing a masking rule in Data Masker*

What is it?

SQL Provision is Redgate's solution for (compliant) test data management. It consists of two products: Data Masker, a static data masking product that the company has inherited from Net2000, and SQL Clone, a preexisting Redgate product that provides database cloning and provisioning. Together, they allow you to create and distribute masked, production-like copies of your data wherever it is required, principally for development and testing

The image in this Mutable Quadrant is derived from 13 high level metrics, the more the image covers a section the better. Execution metrics relate to the company, Technology to the product, Creativity to both technical and business innovation and Scale covers the potential business and market impact.

purposes. In this way, SQL Provision provides a complete test data management solution. They are also available as separate products, if required.

SQL Clone runs exclusively on SQL Server; Data Masker will run on either SQL Server or Oracle.

What does it do?

Data Masker is a static data masking product that can mask millions of rows an hour via the application of sets of masking rules. It maintains the credibility of masked data by ensuring that correlated values (for instance, age and date of birth) remain consistent after masking. This is done via substitution rules using a correlated data set that contains potential values to substitute. This can be seen in *Figure 1*. For instance, your data set could contain random surnames, or it could contain random US zip codes with accompanying state, county and town names appropriate to that zip code. These data sets are available out of the box or can be user defined. In addition, this can be done between databases or even database instances. Data Masker also has the ability to generate synthetic data in a limited capacity. This capability is best utilised when production data is either unavailable or incomplete.

Data masking	★★★★	Sensitive data discovery	★★★★⯪	
Data virtualisation	★★★★	Synthetic data generation	★★★	
Ease of use	★★★★⯪	Test data provisioning	★★★★⯪	

> " We've cut the time for database provisioning by more than 85%. "
> **Paymentsense**

Notably, masking in Data Masker always retains relational integrity and includes the capability to mask primary or foreign keys without a join operation. Data Masker also provides a column finder that allows you to search your database based on column name. It automatically generates reports whenever a masking rule is run, making the masking process fully auditable, and allows you to either apply your masking rules immediately via the application itself or export them for use elsewhere. The latter capability is particularly important for interacting with the second half of SQL Provision, SQL Clone.

Figure 2 – The SQL Clone dashboard

SQL Clone (as shown in **Figure 2**) allows you to create and centrally manage images and virtualised clones of your production data. In this case, images are complete point-in-time copies of a database, taken from either a live server or a backup. As images are often quite large (since they are complete copies) they are usually stored centrally. Importantly, you can modify your image during its creation using either SQL scripts or sets of masking rules exported from Data Masker. In the latter case, the masking rules will be applied to your data before the image is created, meaning that the created image will be masked. Moreover, since Data Masker always maintains referential integrity, it will be referentially intact. In addition, if multiple sets of masking rules are used, you can choose the order to apply them in.

Clones, on the other hand, are derived from an image and only store the changes between themselves and the images they were derived from. Since they only store differences, they are small in size (usually less than a hundred megabytes) and can be created and deleted very quickly and at will, wherever they need to be at any particular time. For the purposes of test data management, this means that your testers can provision a clone of a masked image to their local machines whenever they need test data. This can be accomplished using self-service, meaning that your testers can retrieve test data without having to wait on an administrator. The product also provides PowerShell-driven automation for creating and updating images and clones, allowing you to ensure that your testers always have up to date test data.

Why should you care?

Redgate's entire approach hinges on two concerns: compliance (with existing mandates, such as GDPR, as well as forthcoming regulations) and DevOps. Effective test data management is essential for achieving both of these. Desensitising your test data is necessary for compliance with a variety of mandates, as well ensuring data privacy and security (protection from data breaches, for instance), while timely provisioning of test data – delivering the right test data to the right place at the right time – is an important component of any DevOps pipeline. Consequently, SQL Provision provides both of these capabilities.
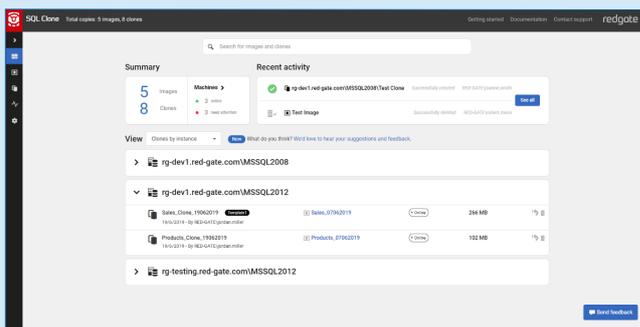
Moreover, SQL Provision does so using a combination of database cloning and data masking. This has some clear advantages over competing approaches, such as data subsetting or synthetic data generation, most of all that it guarantees that your test data will be representative. It also makes provisioning that test data fast and easy. What's more, Redgate is uniquely positioned in offering a solution based on database cloning to the mid-market, whereas competing products tend to be targeted at the high-end.

The Bottom Line

SQL Provision is the only solution of its kind aimed at the mid-market. It lacks some of the sophistication of more expensive test data management products, particularly when it comes to masking and discovery, but on the other hand, do you really need to pay for all those bells and whistles – and are they worth giving up the advantages of Redgate's approach? If your answer is *'no'*, and you're using SQL Server, then SQL Provision may be an appropriate choice.

FOR FURTHER INFORMATION AND RESEARCH CLICK HERE