INTRODUCTION TO MICROSOFT AZURE SQL DATABASE

BY HERVE ROGGERO

SUMMARY

This white paper introduces you to Microsoft Azure SQL Database (previously SQL Azure) and offers an overview of this Microsoft cloud database and related services, including Data Sync and Reporting.

INTRODUCTION

As Microsoft continues to invest heavily in its Microsoft Azure platform, many DBAs and developers are beginning to understand one of the fundamental values of cloud computing: Flexibility. In order to gain the necessary flexibility, Microsoft made certain infrastructure and architecture decisions that may be surprising at first, leading some DBAs and developers to question the readiness of the Microsoft Azure SQL Database platform. When we look closer and understand the benefits of those tradeoffs, however, Microsoft Azure SQL Database becomes a very attractive offering.

Note that for the remainder of this paper, “SQL Database” refers to the Microsoft Azure SQL Database platform.

The purpose of this white paper is to explain how the SQL Database platform works, to analyze some of its strengths, and to provide an overview of the Azure services that leverage the SQL Database platform. In addition to gaining a better understanding of the technology, you will also better understand how certain companies are able to leverage SQL Database for their business.

WHAT IT IS AND WHAT IT IS NOT

First, let's address some of the most fundamental questions about SQL Database. As an SQL Server DBA or a .NET programmer, you may think that SQL Database is nothing more than SQL Server in the cloud, and that most of your current knowledge of SQL Server applies. Although SQL Database is a subset of SQL Server, with similar concepts, it is more accurate to think of SQL Database as the relational engine of SQL Server in the cloud.

Is SQL Database conceptually the same as SQL Server?

No. SQL Server is a server hosting multiple databases on a single machine. SQL Database is an Internet DNS address holding a reference to your databases that are actually stored on different servers. SQL Database determines which server to connect you to using the database name you provide during the connection request.

By disassociating the concept of a database server and its databases, Microsoft has essentially created the foundation for a massively distributed database system. Your SQL Database server could be hosting 10 individual databases, all on different physical servers.

This architecture allows Microsoft to build new features that will be difficult, if at all possible, to build on SQL Server. For example the CREATE DATABASE command has a new option: COPY AS. This option allows you to create a clone of an existing SQL Database on another server:

```
CREATE DATABASE mydb2 AS COPY OF mydb
```

At first glance, this statement looks simple and almost unworthy of any further analysis. However, consider what would need to take place on an SQL Server environment to make this statement work: Find an available SQL Server on the network ready to accept a database; create a database with the proper disk configuration settings; and finally backup and restore the database to ensure transactional consistency. As you can see, this simple statement demonstrates the power and simplicity that cloud computing can enable.

Is SQL Database Highly Available?

Yes. SQL Database has automatic failover capabilities ensuring high availability. There are a total of three sets of your data: The primary database, and two standby copies. If the primary database becomes unavailable, one of the standby databases will take over almost instantly. Although an application may experience a temporary disconnection, users may not even realize that database switch has taken place if the application is designed with properconnection retry logic.

Is my data safe?

Yes. SQL Database comes with a few interesting features that offer unique protection capabilities. For example, SQL Database has a built-in Denial-of-Service attack detection system that blocks certain connections if it detects too many failed login attempts. SQL Database also comes with a built-in firewall that allows you to configure which IP Addresses are allowed to connect to your databases. These features significantly improve access control over your databases. SQL Database also requires SSL to be used at all times throughout the duration of your sessions, ensuring privacy for your data in flight.

You should know however that SQL Database does not yet offer strong encryption for your data at rest. Transparent Data Encryption (TDE) and certificates are not supported.

Can I move my database in the cloud?

It depends. If your database uses features that are not supported on SQL Database, you will not be able to migrate your database without making changes. For example your database may use the WITH ENCRYPTION option, which is not supported in SQL Database.

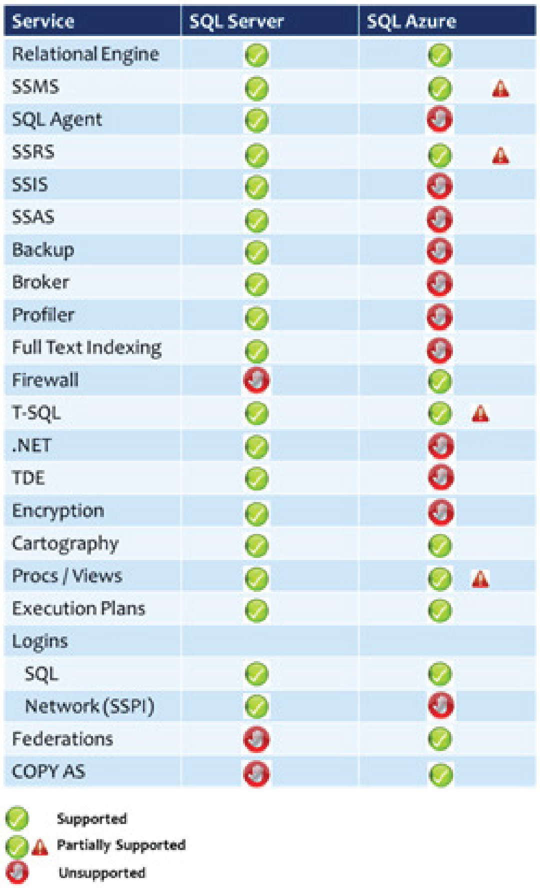
There are other differences that are more difficult to detect because they will not generate an error. The best way to find out is to run the ***Generate and Publish Script*** wizard against your database using SQL Server Management Studio. Make sure to select “SQL Azure” as the destination database engine (***Script for the Database Engine*** option) and observe the output generated by the scripting engine. Incompatibilities will raise errors during the creation of the script.

Can I backup my database in the cloud?

At the moment, Microsoft provides very few options for backup in the cloud, although this is likely to improve as the Microsoft Azure platform evolves. The best way to obtain a backup of your database is to execute the CREATE DATABASE command with the AS COPY OF option as discussed previously. This command creates a transactionally consistent copy of your database which can then be used as the basis for your backup operation. However, the BACKUP command is not available in SQL Database. As a result you will need to rely on the Import/Export function available in the Microsoft Azure management portal to extract the schema and data of your database.

What are some of the features not currently supported by SQL Database?

SQL Database offers many features available in SQL Server. However, there are some key differences that you should know about. For example, SQL Database does not offer a built-in SQL Agent. This means that if you are considering creating jobs on a regular basis against an SQL Database you will need to rely on an on-premise SQL Agent or other scheduler to execute the operations for you.



Service	SQL Server	SQL Azure
Relational Engine	✓	✓
SSMS	✓	✓ ⚠
SQL Agent	✓	✗
SSRS	✓	✓ ⚠
SSIS	✓	✗
SSAS	✓	✗
Backup	✓	✗
Broker	✓	✗
Profiler	✓	✗
Full Text Indexing	✓	✗
Firewall	✗	✓
T-SQL	✓	✓ ⚠
.NET	✓	✗
TDE	✓	✗
Encryption	✓	✗
Cartography	✓	✓
Procs / Views	✓	✓ ⚠
Execution Plans	✓	✓
Logins		
SQL	✓	✓
Network (SSPI)	✓	✗
Federations	✗	✓
COPY AS	✗	✓

✓ Supported
✓ ⚠ Partially Supported
✗ Unsupported

You will also notice that SQL Database does not support Network logins (SSPI Context). This means that all your databases in SQL Database rely on SQL logins.

How do I manage my SQL Databases?

You can use SQL Server Management Studio to manage your SQL Databases. However, you should note that SSMS is limited in its ability to support SQL Database because many of the menu options are not available.

You can also use the SQL Database management portal to design and query your cloud databases. To access the management portal, simply open a browser and type the URL of your database as such:

```
<sqldatabaseServerName>.database.windows.net
```

The SQL Database Server Name is assigned to you by Microsoft when you first create a database

What is Federations and how does it help with scalability?

A new scalability feature, Federations was recently added to SQL Database. This feature allows an administrator to store related tables in a collection of databases, all stored on different servers. Administrators can then manage those databases independently of the application and transparently perform split operations to further distribute the data.

Although splitting the data and managing the federated environment is transparent to the application, the application needs to be developed accordingly. Adopting federations is a design decision because certain restrictions apply. For example, federated tables do not support Identity columns.

Other important factors come into play when considering the use of federations, such as managing a distributed database schema, enforcing referential integrity across databases at the application level and disaster recovery.

Nevertheless, federations offer a relational scalability platform that can be useful to certain applications in need of unpredictable and increasing scalability.

RELATED SQL SERVICES

SQL Data Sync

If you use multiple SQL Databases you may need to keep certain tables synchronized so that changes in one SQL Database are replicated to another. Or you may also need to keep an on-premise database synchronized with an SQL Database. The SQL Data Sync service allows you to create a replication topology between SQL Databases, and optionally with on-premise SQL Server databases.

Configuring SQL Data Sync services is simple and performed entirely in the Microsoft Azure management portal. The replication of data can be bidirectional if needed and you can choose to synchronize certain columns of a table, and even apply a row filter. Keep in mind however adding an SQL Server database to the topology requires installing an agent on the SQL Server machine.

Microsoft Azure SQL Reporting

If you are interested in running reports against your SQL Databases you can create and publish reports in Microsoft Azure SQL Reporting, a cloud reporting service based on SQL Server Reporting Services.

This cloud reporting service makes it easy to deploy and manage reports that can be accessed by users securely and embed them in existing applications. If you are currently using SQL Server Reporting Service, you will find it easy to publish existing reports and run them by simply typing the report URL into a browser. As you would expect, the major advantage of this service is its ability to scale and offer high availability.

Note that existing reports may need to be modified slightly to conform to the limitations imposed by Microsoft Azure SQL Reporting.

HOW COMPANIES ARE USING SQL DATABASE

Because of its ease of management and automatic high availability, SQL Database is becoming a very popular database platform for many organizations. The following implementations provide a few common usage patterns for using SQL Database.

Traditional Relational Database

As you might expect, a popular usage of SQL Database is to implement the primary relational data storage of an application in order to leverage its high availability and scalability options. In this implementation, the SQL Database is considered critical because it must be available for the main applications to function. As a result backup, data synchronization and reporting may become critical functions of this implementation.

Synchronization Database

Another interesting use case for SQL Database is to leverage its natural network accessibility. As a cloud database, it can be used to synchronize or centralize records of distributed applications. For example a traditional office management application may have its own SQL Server database, but cannot easily reach out to other office locations to obtain a comprehensive list of customers. These applications can leverage SQL Data Sync and SQL Database to create a centralized database that can be accessed at any time by the application to check and synchronize records from other locations.

Edge Database

Another use case that leverages cloud databases is the need to push certain records to make them available on small form factor devices or a website. Traditional WinForm applications installed in smaller offices and remote locations lack the ability to expose their latest information online. By leveraging SQL Data Sync and SQL Database, legacy applications can push a subset of their data easily so that is can be consumed by phone devices for example.

CONCLUSION

Understanding the primary differences between SQL Server and SQL Database is the first step to building cloud computing applications that can scale and leverage the flexibility of cloud computing. I presented some of the primary differences between those platforms and gave you an overview of the services that leverage SQL Databases, including SQL Data Sync and SQL Reporting.

We also discussed some of the popular use cases for leveraging SQL Databases. Many organizations are adopting the cloud computing as their primary computing environment and others find ways to use Microsoft Azure to enhance their current offerings and create solutions that would otherwise be difficult to achieve.

ABOUT THE AUTHOR

HERVE ROGGERO

Herve Roggero, SQL Azure MVP, is the founder of Blue Syntax Consulting, a company specialized in cloud computing products and services.

Herve's experience includes software development, architecture, database administration and senior management with both global corporations and startup companies. Herve holds multiple certifications, including an MCDBA, MCSE, MCSD. He also holds an master's in business administration from Indiana University.

Herve is the co-author of PRO SQL Azure from Apress.