Azure SQL Database

Ranjit Karni

VPark Innovations
Ranjit.balu@gmail.com

Ph.No: 9676976662

Introduction

- **SQL Database** is a cloud based relational database **service** that is built on SQL Server technologies. It supports T-SQL commands, tables, indexes, views, primary keys, stored procedures, triggers, roles, functions etc.

- SQL Database delivers predictable performance, scalability with no downtime, business continuity and data protection – all with **near-zero administration**.

- You can focus on rapid app development and accelerating your time to market, rather than managing virtual machines and infrastructure.

- Because it's based on the SQL Server engine, SQL Database **supports existing SQL Server tools, libraries and APIs**, which makes it easier for you to move and extend to the cloud

- SQL databases is available in **Basic, Standard and Premium** service tiers.

- Each service tier offers different levels of performance and capabilities to support lightweight to heavyweight database workloads.

- You can build your first app on a small database for a few bucks a month, then change the service tier manually or programmatically at any time as your app goes viral worldwide, **without downtime to your app** or your customers.

Benefits of SQL Database

- **High Availability**: For each SQL Database created on Windows Azure, there are **three** replicas of the database

- **On Demand**: One can quickly provision the database when needed with a few mouse clicks

- **Reduced Management Overhead**: It allows you to extend your business applications into the cloud by building on core SQL Server functionality while letting Windows Azure support staff handle the maintenance and patching tasks.

SQL Database Top Features

- Tables, views, indexes, roles, stored procedures, triggers and user defined functions

- Constraints

- Transaction

- Temp tables

- Basic functions (Aggregates, math, string, date/time)

- Constants

- Cursors

- Index management and index rebuilding

- Local temporary tables

- Reserved keywords

- Statistics management

- Table variables

- Transact-SQL language elements such as create/drop databases, create/alter/drop tables, create/alter/drop users and logons

Features not supported in SQL Database

- Windows Authentication

- Not all T-SQL commands supported

- User-defined types

- Access to System Tables

- Common Language Runtime (CLR)

- Database File Placement

- Database Mirroring

- Distributed Queries

- Distributed Transactions

- Filegroup management

- Global temporary tables

- Support for SSAS (Instead use Data Factory), SSRS (Power BI), SSIS (Separate Service)

- Support for Replication or SQL Server Service Broker

- Support Backup and Restore

Azure SQL Database Managed Instance

- It is a new deployment model of Azure SQL Database near **100% compatibility** with the latest SQL server on-premises (Enterprise Edition) Database Engine

- It provides a **native virtual network (Vnet)** implementation that addresses common security concerns and a business model **favorable for on-premises** SQL Server customers.

- Managed Instance allows existing SQL Server customers to **lift and shift** their on-premises applications to the cloud with minimal application and database changes

- Managed Instance preserves all **PaaS capabilities** (automatic patching and version updates, automated backups, high availability), that drastically reduces management overhead and TCO.

Database Level Firewall Setting

- To create database-level firewall setting for only IP 0.0.0.4

Execute sp_set_database_firewall_rule N’Example DB Setting 1’, ‘0.0.0.4’, ‘0.0.0.4’;

To update database-level firewall setting to create a range of allowed IP addresses

Execute sp_set_database_firewall_rule N’Example DB Setting 1’, ‘0.0.0.4’, ‘0.0.0.6’;

Methods to Migrate On-Premise to Azure SQL Database

- Migration tools help in handling repetitive tasks of migrating (Ex: Migrating columns, identify field etc)

- Consider what data to move, prepare for the move and use tools for the migration

- Any data of a RDBMS can be moved to SQL Azure

Tools for migration of databases are:

- Generate SQL Script using SQL Server Management Studio (SSMS)

- Microsoft SQL Server Migration Assistant (SSMA) – It can move data from (Oracle, MySQL, Sybase, IBM-db2, Microsoft SQL Server)

Planning the deployment of an Azure SQL Database

- Your first step should be to determine whether any service **specific functional limitations** would require that you choose an IaaS-based implementation of SQL Server or another relational database management system.

- When you plan an Azure SQL Database deployment, your primary consideration should be database's **intended workload**.

- Also, you should consider the scalability limits of Azure SQL Database. These might include maximum **supported database size or performance**, with respect to transactional throughput, maximum concurrent requests, maximum sessions and maximum concurrent logins.

- In SQL Database, the relative measure of a database's ability to handle resource demands is expressed in **Database Transaction Units (DTUs).** DTUs provide a way to describe the relative capacity of a performance level based on a blended measure of CPU, memory, reads and writes.

Service Tiers and Target Workloads

Basic:

- Best suited for a **small database**, supporting typically one single active operation at a given time.
- Example include databases used for **development or testing**, or small-scale infrequently used applications.
- For infrequent access and less demanding workloads.

Standard:

- The go-to option for cloud applications with low to medium IO performance requirements, supporting **multiple concurrent queries**.
- Examples include workgroup or web applications.

Premium:

- Designed for **high transactional volume** with **high IO performance** requirements, supporting many **concurrent users**.
- Examples are databases supporting mission critical applications.

Premium RS:

- Designed for **IO-intensive workloads** that do **not require the highest availability** guarantees.
- Examples include testing high-performance workloads, or an analytical workload where the database is not the system of record.

Elastic Pools

- Elastic pools provide a **simple cost effective solution** to manage the performance goals for **multiple databases** (hosted on the same logical server) that have widely **varying and unpredictable** usage patterns.

- **Elastic DTUs** (eDTUs) are used elastic databases in an elastic pool.

- A pool is given a set number of eDTUs, for a set price. Within the pool, individual databases are given the flexibility to auto-scale within set parameters.

- Provisioning resources for the entire pool rather than for single databases simplifies your management tasks.

- Under heavy load, a database can consume more eDTUs to meet demand. Databases under light loads consume less, and databases under no load consume no eDTUs.

- Additional eDTUs can be added to an existing pool with no **database downtime** or no impact on the databases in the elastic pool. Similarly, if extra eDTUs are no longer needed they can be removed from an existing pool at any point in time.

- You can add or subtract databases to the pool. If a database is predictably under-utilizing resources, move it out.

- 1eDTU = 1.5 DTUs

Which Databases go in a Pool

- Databases that are great candidates for elastic pools typically have periods of activity and other periods of inactivity.

- Databases with varying activity over time are great candidates for elastic pools because they are not all active at the same time and can share eDTUs.

- Not all databases fit this pattern. Databases that have a more constant resource demand are better suited to the **Basic**, **Standard and premium** service tiers where resources are individually assigned

- While the eDTU unit price for a pool is 1.5x greater than the DTU unit price for a single database, **pool eDTUs can be shared by many databases and fewer total eDTUs are needed**.

Configure Azure SQL Database Auditing

- Azure SQL Database Auditing tracks database events and writes them to an audit log in your Azure Storage account.

- Auditing can help you maintain regulatory compliance, understand database activity and gain insight into discrepancies and anomalies that could indicate business concerns or suspected security violations.

SQL Database Auditing allows you to:

- **Retain** an audit trail of selected events. You can define categories of database actions to be audited.

- **Report** on database activity. You can use preconfigured reports and a dashboard to get started quickly with activity and event reporting.

- **Analyze** reports. You can find suspicious events, unusual activity and trends.

There are two **Auditing Types**:

- **Blob Auditing:** Logs are written to Azure Blob Storage. This is a newer auditing method, which provides **higher performance** supports **higher granularity object-level auditing** and is **more cost effective**.

- **Table Auditing:** Logs are written to Azure Table Storage (This option is now Deprecates)

Export and Import of Database

- In Azure SQL Database, you **cannot** directly use the database and transaction log backup capabilities of SQL Server.

- Historically, this was remediated by periodically exporting a copy of each database that you want to protect and storing the copy in a **.bacpac** file in a storage account.

- In the event of a SQL database or server failure, you could then create a new SQL database server, if necessary and import the copy of the database from the exported file.

Business Continuity

- SQL Database **automatically** creates a database backups and uses Azure read-access geo redundant storage (RA-GRS) to provide **geo-redundancy**.

- SQL Database automatically performs a combination of **full database backups weekly, differential database backups hourly** and **transactional log backups every five minutes** to protect your business from data loss.

- The full and differential database backups are also replicated to a paired data center for protection against a data center outage.

- These backups are created automatically. SQL Database provides up to 200% of your maximum provisioned database storage as backup storage at no additional cost.

- If your database exceeds the provided backup storage, you can choose to **reduce the retention period** by contacting Azure Support.

- Another option is to pay for extra backup storage that is billed at the standard Read-Access Geographically Redundant Storage (RA-GRS) rate.

- Each SQL Database backup has a retention period that is based on the service-tier of the database. The retention period for a database is:

Basic Service tier – 7 Days

Standard Service tier – 35 Days

Premium Service tier – 35 Days

Active Geo-Replication

- Active Geo-Replication enables you to configure up to **four readable secondary databases** in the same or different data center locations (regions).

- Secondary databases are available for **querying and for failover** in the case of a data center outage or the inability to connect to the primary database.

- If the primary database goes offline unexpectedly or you need to take it offline for maintenance activities, you can quickly promote a secondary to become the primary (**also called a failover**) and configure applications to connect to the newly promoted primary.

- With a planned failover, there is no data loss.

- With an unplanned failover, there may be some small amount of data loss for very recent transactions due to the nature of asynchronous replication. After a failover, you can later failback – either according to a plan or when the data center comes back online.

- In all cases, users experience a small amount of downtime and need to reconnect

- It is used to reduce recovery time and limit data loss associated with a recovery

- The secondary database must be on the same service tier as the primary, so migrating your primary database to a different service tier requires you to either terminate the geo-replication link and rename the secondary database or simply drop it

Questions?

Ranjit Karni

Vpark Innovations

Ranjit.balu@gmail.com

Ph No: +91-9676976662