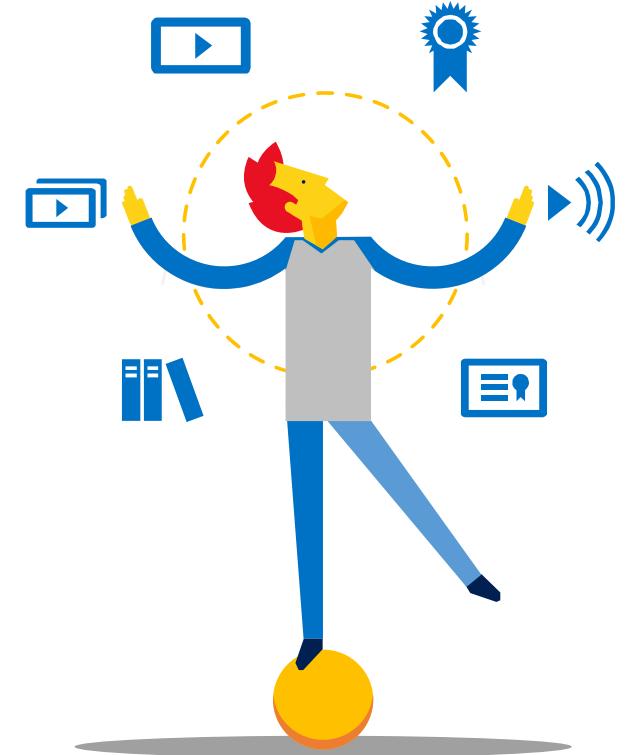


Microsoft Virtual Academy

Exam Overview: Exam 70-473: Designing and Implementing Cloud Data Platform Solutions



The Exam

Available as of December 2015 and updated on a quarterly basis. Last major updates were made June of 2017

Audience

IT Professionals who are:

Able to identify trade-offs and to make decisions for designing public and hybrid cloud solutions.

Able to define the appropriate infrastructure and platform solutions to meet requirements (functional, operations, and deployment)

Check latest updates/changes to the exam:

<https://www.microsoft.com/en-ca/learning/exam-70-473.aspx>

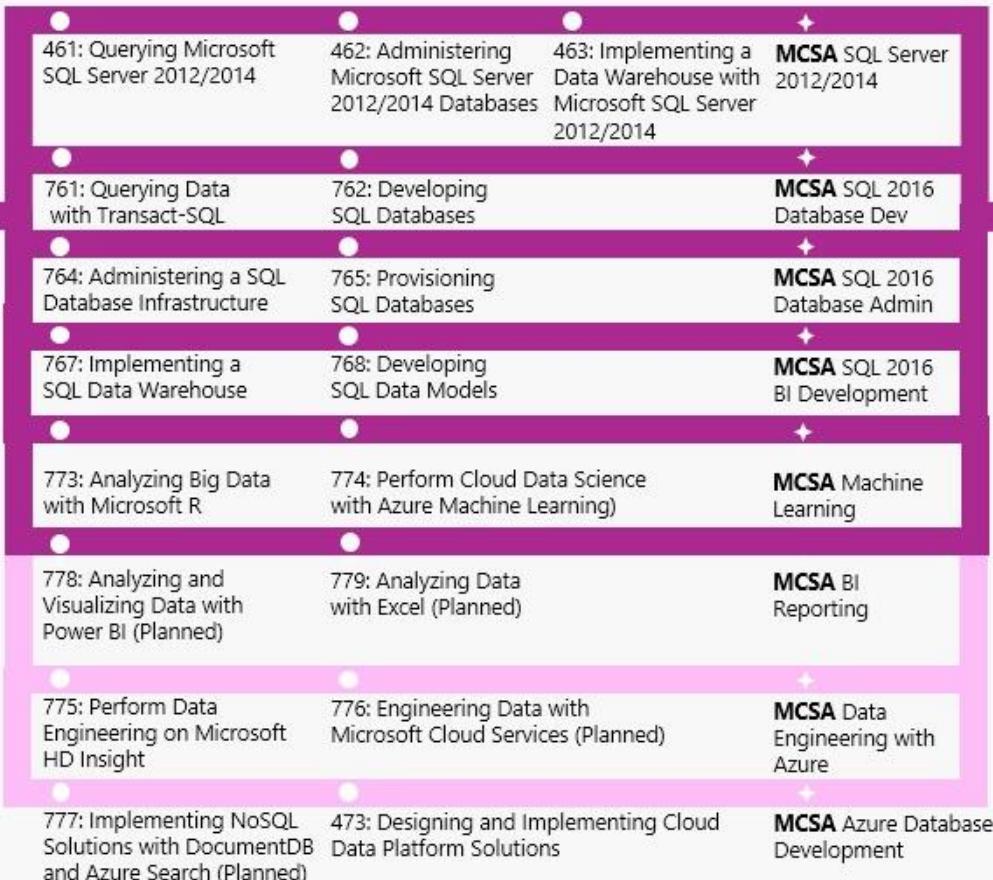
Microsoft Certified Solutions Associate (MCSA)

- Background
 - A specialist who knows “how” to work with an advanced product
 - For those with access to hands-on practical experience.
 - Someone with 1-2 years’ experience
 - Azure Subscription – for hands-on
- Simplified Tracks
 - Requires passing 2 exams
 - Only one more exam to reach MCSE
 - MCSA certification is required in order to become an MCSE.

MCSE Data Management & Analytics

DATA MANAGEMENT & ANALYTICS PATH

DATA MGMT & ANALYTICS



Available: FY18 Q1
5/22/17 – Reflects updates to some MCSA names as well as the MCSE elective pool exams list.

MCSE
Data Mgmt & Analytics
Earned: 2017
Elective

ELECTIVE EXAM POOL

- 473: Designing and Implementing Cloud Data Platform Solutions
- 475: Designing and Implementing Big Data Analytics Solutions
- 464: Developing Microsoft SQL Server Databases
- 465: Designing Database Solutions for SQL Server
- 466: Implementing Data Models and Reports with Microsoft SQL Server
- 467: Designing Business Intelligence Solutions with Microsoft SQL Server
- 762: Developing SQL Databases
- 767: Implementing a SQL Data Warehouse
- 768: Developing SQL Data Models
- 773: Analyzing Big Data with Microsoft R
- 774: Perform Cloud Data Science with Azure Machine Learning
- 775: Perform Data Engineering on Microsoft Azure HDInsight

Important Technology Cert program

- Cloud focused
 - Many exams focus on cloud-based technologies
 - When relevant, hybrid environments are part of solutions
- Constant Updates
 - Exams are continuously updated to reflect changes in the cloud
 - Exam are updated every several months
- New Exam Releases
 - Many new exams released, focusing on cloud platforms, data science, and many new product releases.

Exam Basics

- 40-60 questions (51 in my case)
- 2.5 hours to complete the exam
- Can review questions (for most question types)
- Can include up to 20 different types of questions
- 700 is passing
- 700 <> 70%

"Microsoft reserves the right to update content for any reason at any time to maintain the validity and relevance of our certifications. This includes, but is not limited to, incorporating functionality and features related to technology revisions and service packs into exam content."

- <https://www.microsoft.com/en-us/learning/certification-exam-policies.aspx>

Exam Topics

1. Design and implement database solutions for Microsoft SQL Server and Microsoft Azure SQL Database (20–25%)
2. Design and Implement Security (25–30%)
3. Design for high availability, disaster recovery and scalability (25–30%)
4. Monitor and manage database implementations in Azure (25–30%)

1. Design and implement database solutions for Microsoft SQL Server and Microsoft Azure SQL Database (20–25%)

Objectives

Design a hybrid SQL Server solution

Design a Disaster Recovery topology for a hybrid deployment

Design a data storage architecture

Design a security architecture

Design a data load strategy (bcp, no ETL tools)

Design a data synchronization strategy (Azure Data Sync)

Objectives (continued)

Implement SQL Server on Azure Virtual Machines (VMs)

Provision SQL Server in an Azure VM

Configure firewall rules

Configure and optimize storage

Migrate an on-premises database to Microsoft Azure

Configure and optimize VM sizes by workload

Objectives (continued)

Design a database solution on Azure SQL Database and SQL Server in Azure

Design a solution architecture

Design a Geo/DR topology

Design a security architecture

Design a data load strategy

Determine the appropriate service tier

Determine the appropriate deployment scenario

Determine IaaS vs PaaS

Determine application access in Azure

Objectives (continued)

~~Design and implement data warehousing on Azure~~

~~Design a data warehousing solution on Azure~~

~~Design a data load strategy and topology~~

~~Configure SQL Data Warehouse~~

~~Migrate an on-premises database to SQL Data Warehouse~~

Objectives (continued)

Design and implement MySQL and PostgreSQL database solutions in Azure

Design security

Design a data load strategy

Determine the appropriate service tier

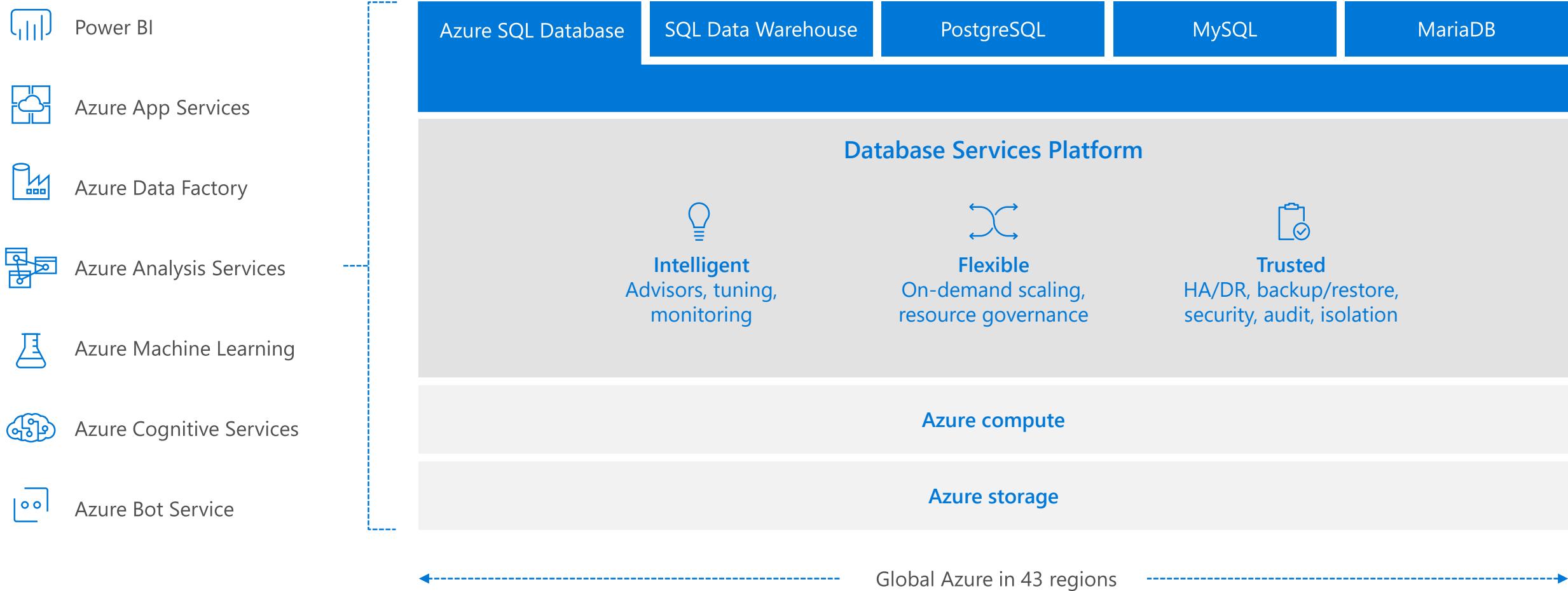
Provision databases and servers

Configure firewall rules

Migrate to Azure

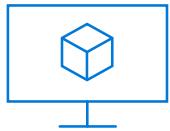
Configure for scale and performance

AZURE RELATIONAL DATABASE PLATFORM



MICROSOFT AZURE DATABASE SERVICES

Leverage a single programming surface for a consistent experience



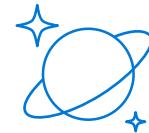
SQL Server on Azure Virtual Machines

SQL Server installed and hosted in the cloud



Azure SQL Database

Intelligent relational cloud database service



Azure Cosmos DB

Globally distributed, multi-model database service



Azure SQL Data Warehouse

Elastic, enterprise-class data warehouse as a service



Azure Database for MySQL

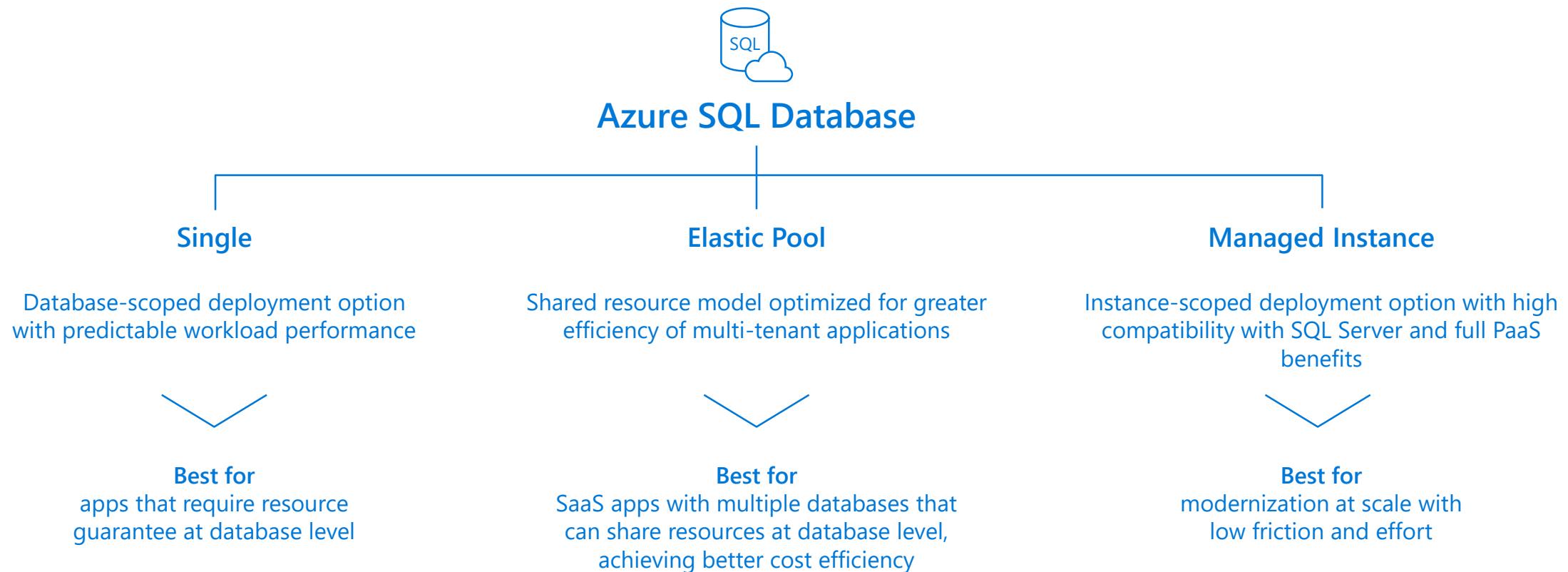
Managed MySQL database service for app devs



Azure Database for PostgreSQL

Managed PostgreSQL database service for app devs

AZURE SQL DATABASE DEPLOYMENT OPTIONS



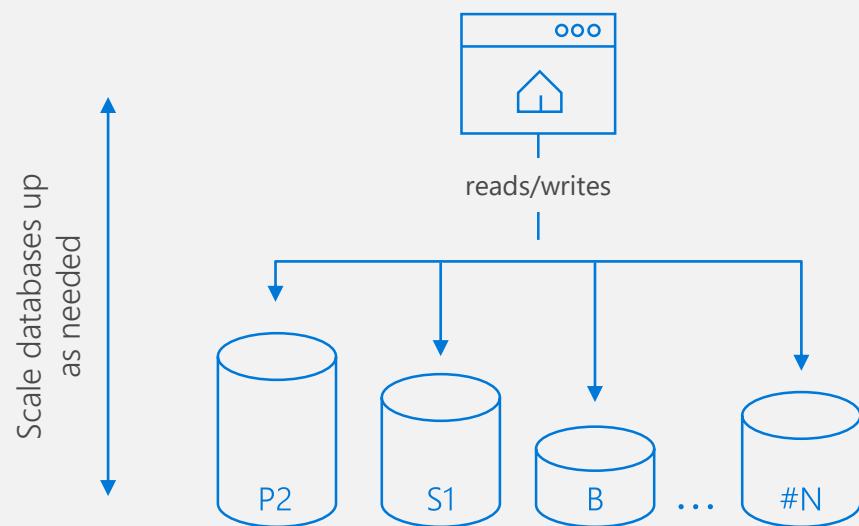
SINGLE DATABASE VS ELASTIC POOL

Predictable workloads

Single databases or partitioned data across multiple databases

Scale between service tiers and performance levels as capacity needs fluctuate

Single database or partitioned databases

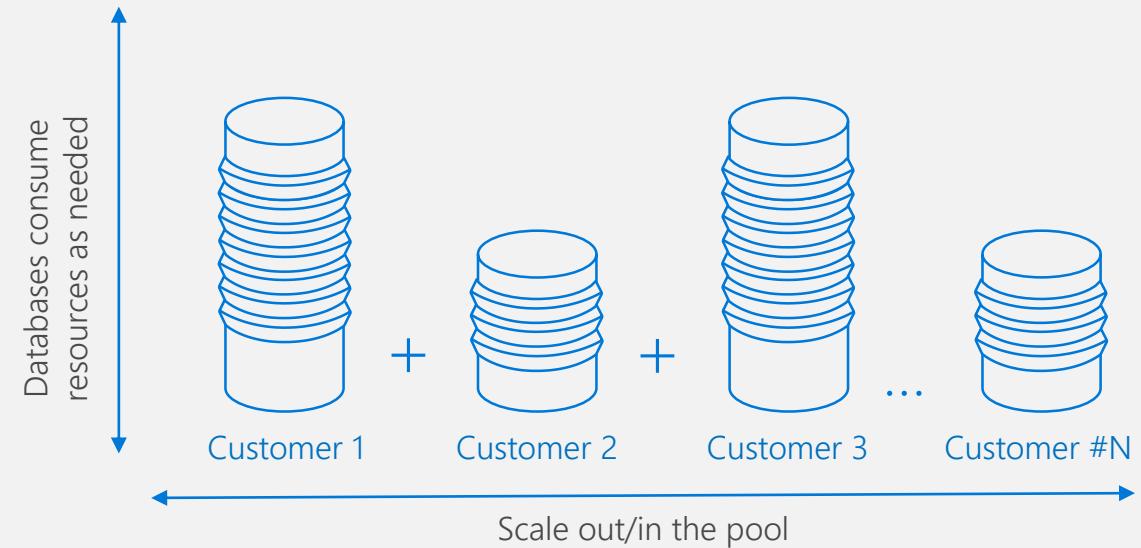


Unpredictable workloads

For large numbers of databases with unpredictable performance demands

Pool resources are shared between these databases

Elastic Database Pool



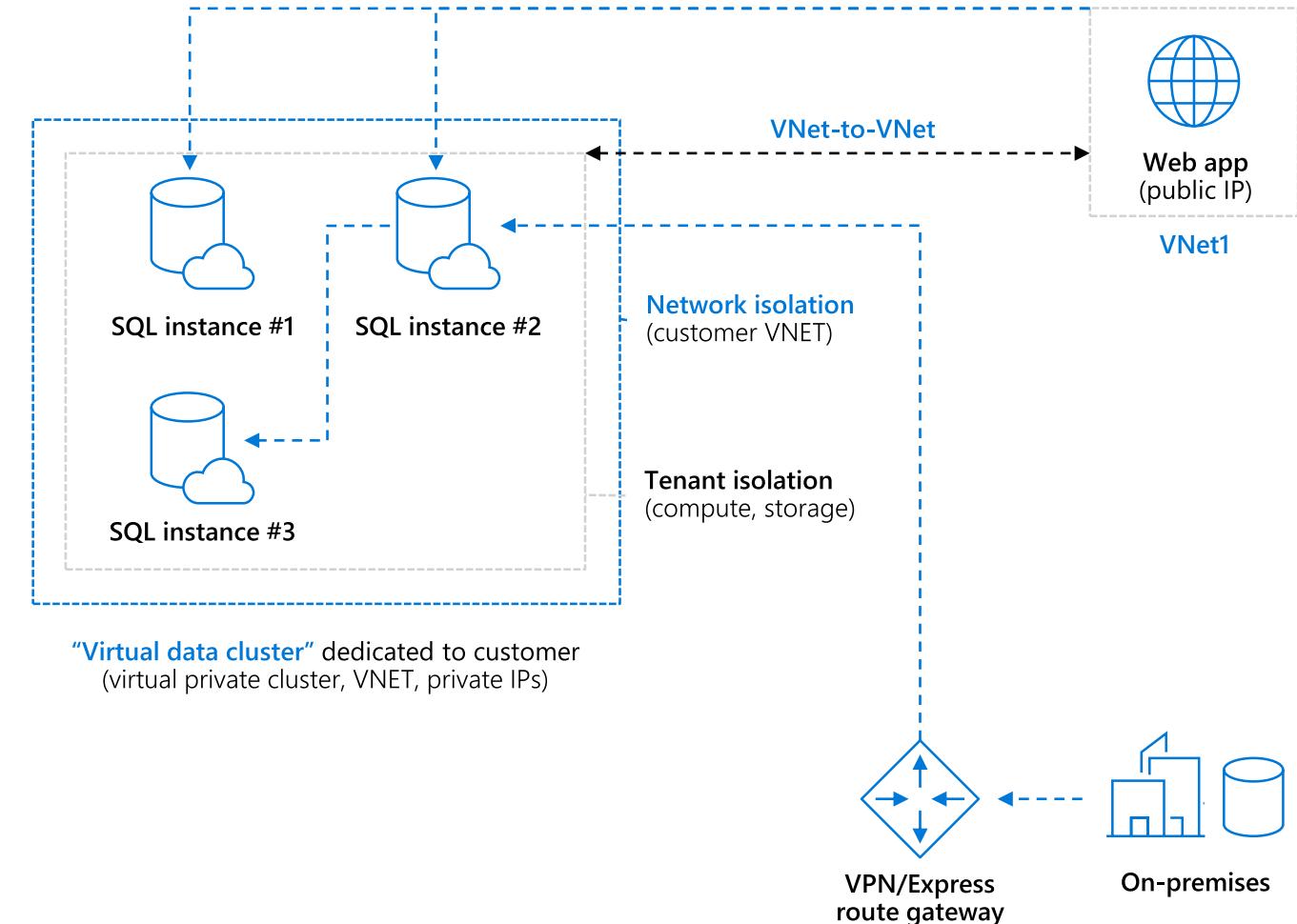
DEDICATED RESOURCES THROUGH CUSTOMER ISOLATION

Enable full isolation from other tenants without resource sharing

Promote secure communication over private IPs with VNET integration



VNET support in SQL Database Managed Instance



DATA PLATFORM CONTINUUM

Shared lower cost



Dedicated higher cost

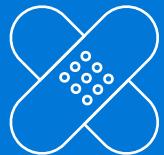
Higher administration

Lower administration

KEY BENEFITS OF AZURE SQL DATABASE



Independently scale compute and storage to match both performance and financial needs



High availability and disaster recovery with 99.99% uptime availability SLA and active-geo replication, point-in-time restore, and geo-restore



Up to 100x performance improvements with support for In-Memory Columnstore queries



Improved monitoring and troubleshooting with Extended Events and visibility into more than 100 new table views

101010
010101
101010

Support for key programmability functions to drive more robust application design

FOCUS ON YOUR BUSINESS

Your work so far	How PaaS helps
Hardware purchasing and management	Built-in scale on-demand
Protect data with backups (with health checks and retention)	Built-in point-in-time restore
High availability implementation	Built-in 99.99% SLA and auto-failover
Disaster recovery implementation	Built-in geo-redundancy and geo-replication
Ensure compliance with standards on your own	Built-in easy to use features
Secure your data from malicious users and mistakes	Built-in easy to use features
Role out updates and upgrades	Built-in updates and upgrades
Monitor, troubleshoot, and manage at scale	Built-in easy to use features
Tune and maintain for predictable performance	Built-in easy to use features

We take care of your database chores

HADR Deployment Options (Azure VMs)

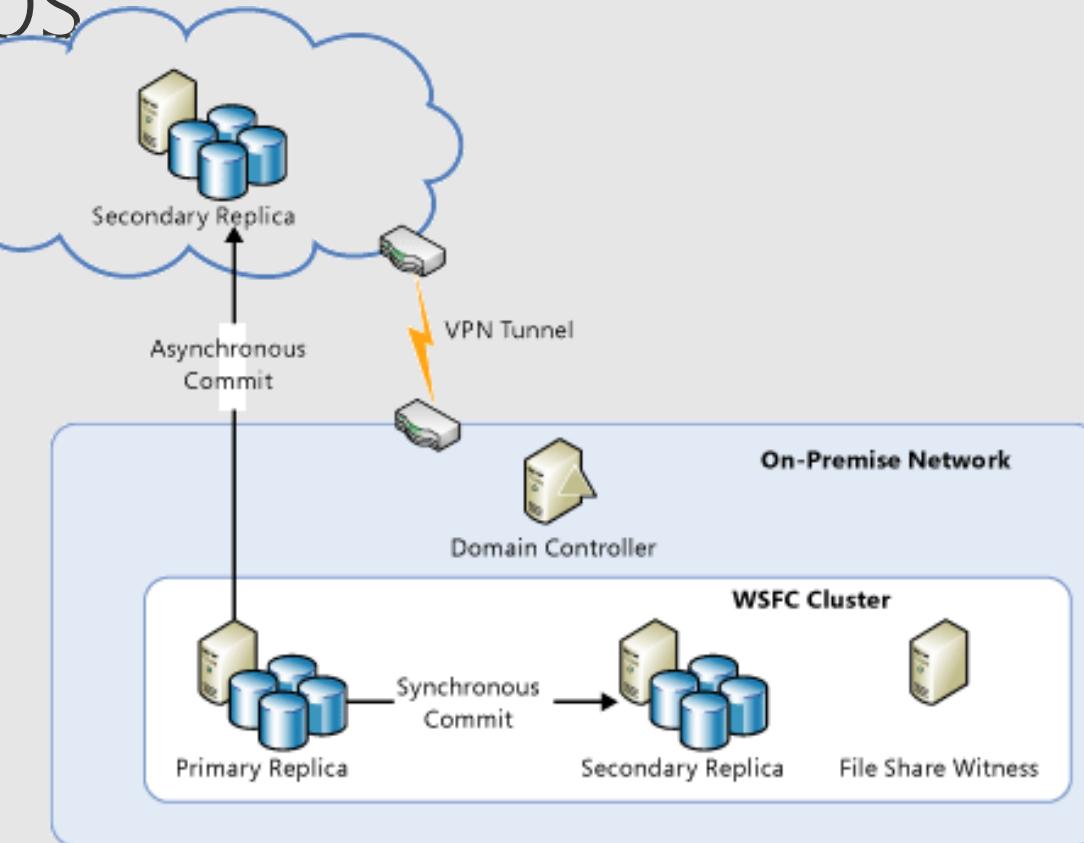
SQL Server HADR technologies supported in Azure:

- Always On Availability Groups
- Always On Failover Cluster Instances (FCI)
- Log Shipping
- SQL Server Backup and Restore with Azure Blob Storage Service
- ~~Database Mirroring – Deprecated in SQL Server 2016~~

<https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-windows-sql-high-availability-dr/>

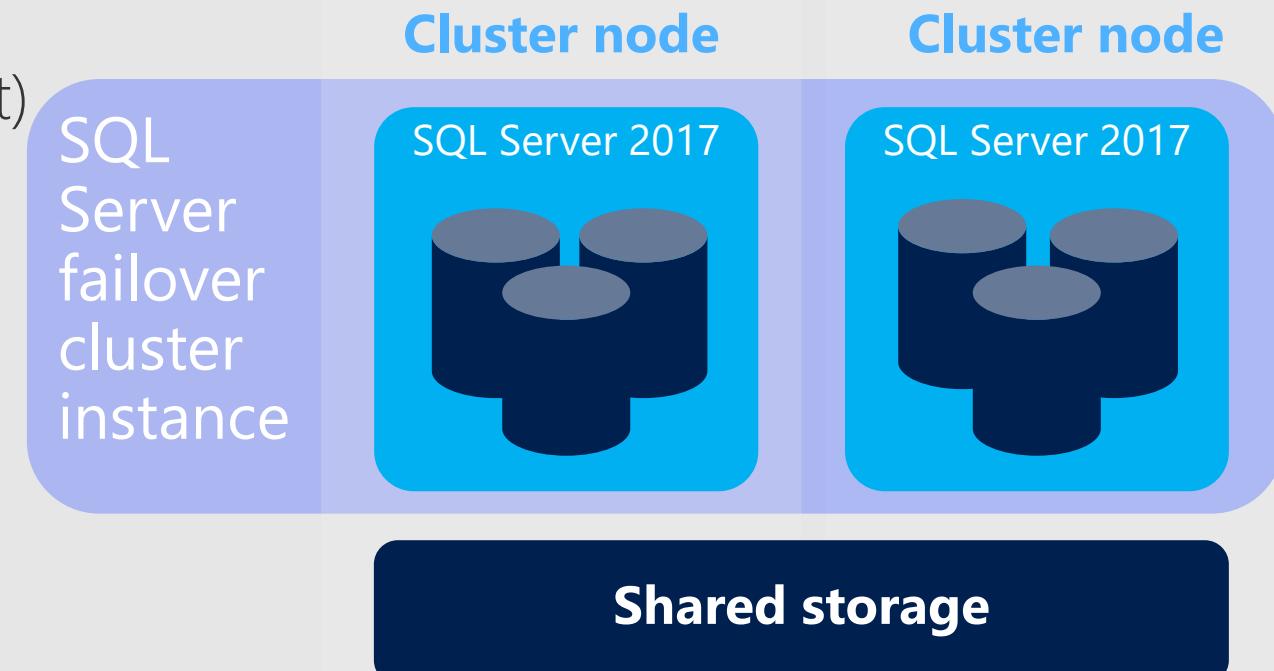
Always On Availability Groups

- HA & DR Solution, maximizes availability of group of user databases (database level)
- Replicas run SQL Instance that maintains local copy of databases
- Clients connect to availability group listener (no connection string change on failover)
- Readable Secondary (ApplicationIntent=ReadOnly)
- Availability Modes:
 - **Asynchronous-commit mode.** When replicas are distributed over considerable distances. Manual Failover. DR Solution.
 - **Synchronous-commit mode.** HA over performance, at the cost of increased transaction latency. Automatic Failover. Better RPO, RTO. HA Solution



Failover Cluster Instances (FCI)

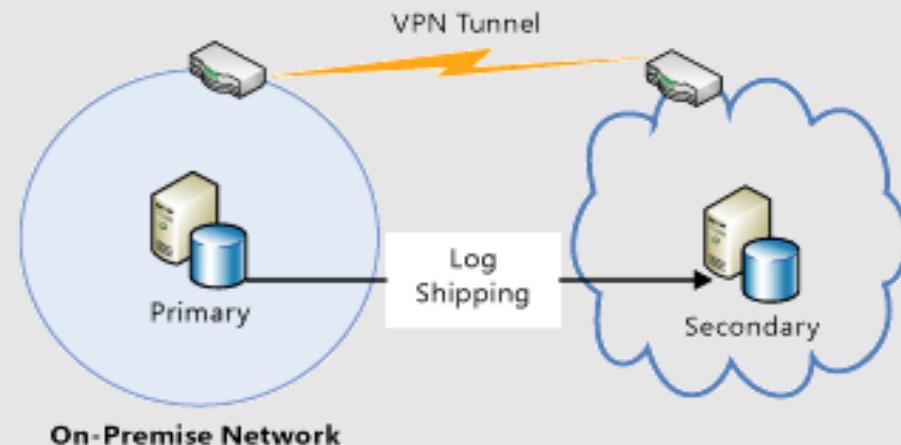
- Single instance of SQL Server installed across Windows Server Failover Clustering (WSFC) nodes (Instance Level)
- FCI appears to be an instance of SQL Server running on a single computer
- Requires Shared Storage (E.g. Windows Server 2016 Storage Spaces Direct (S2D) / SAN)
- No Readable Secondary (Secondary instance starts after resource group ownership transfer)
- HA Solution (Shared Storage requirement)



Others

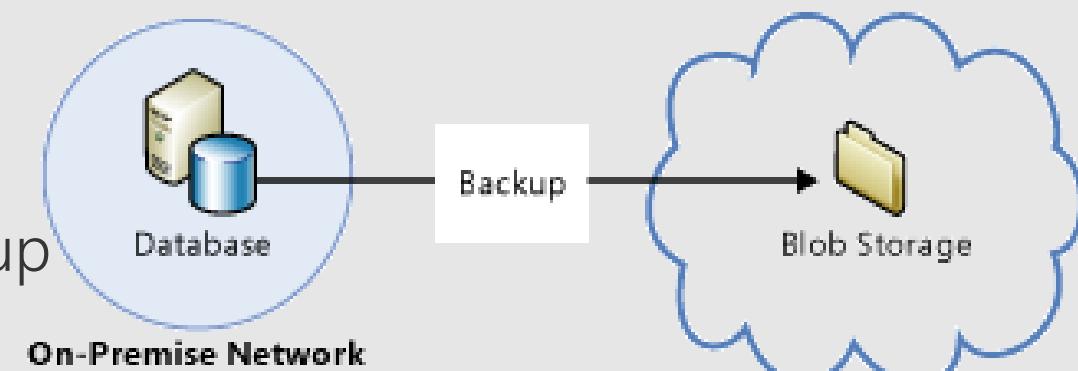
Log Shipping

- Automatically send transaction log backups from a primary database to one or more secondary databases on separate secondary server instances.
- Supports limited read-only access to secondary databases (during the interval between restore jobs).
- Provides warm secondary which is a specified duration behind primary



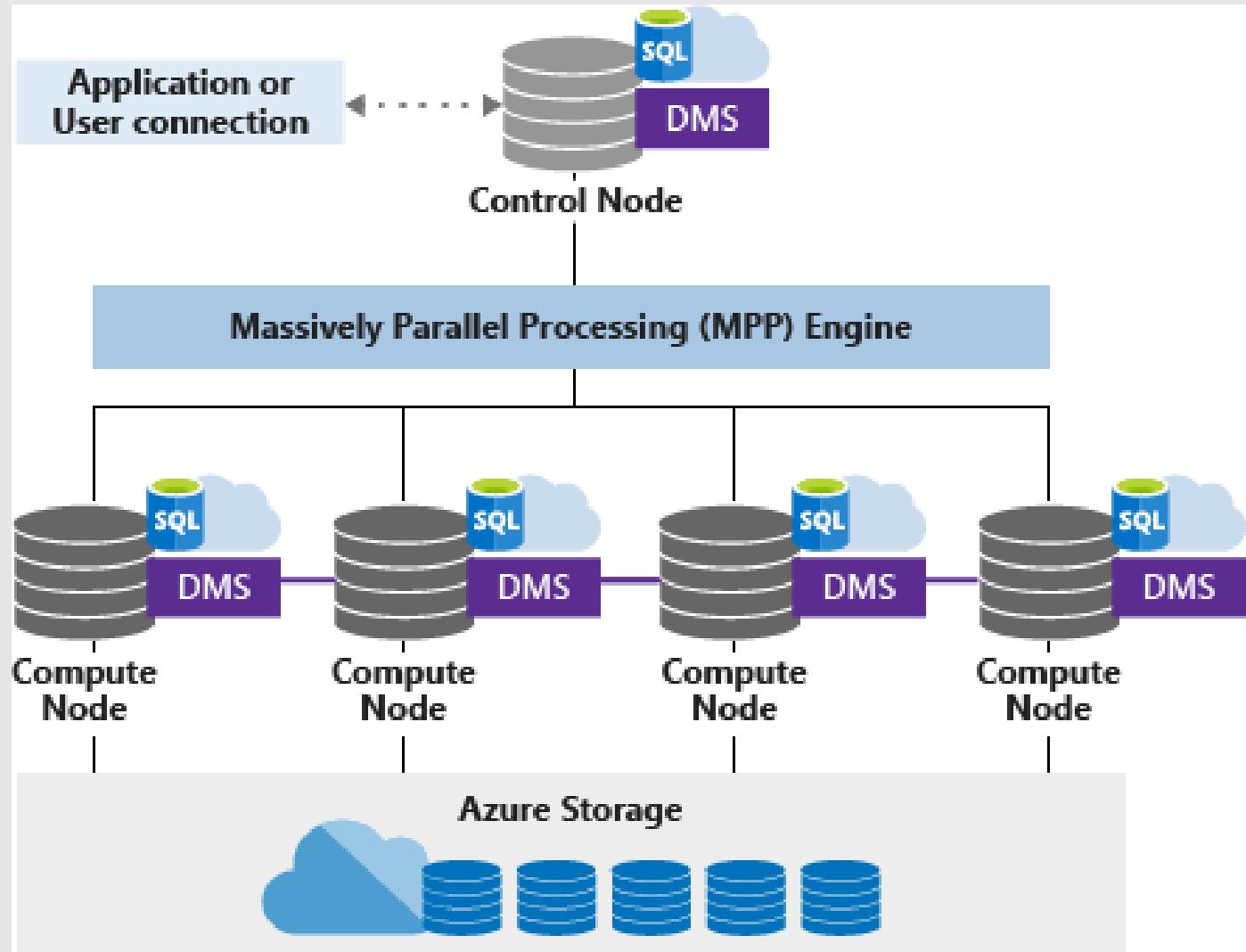
Backup and Restore with Blob Storage

- On-premises production databases backed up directly to Azure blob storage for disaster recovery.



Azure SQL DW – What to expect

- Cloud-based Enterprise Data Warehouse (EDW) that leverages Massively Parallel Processing (MPP) architecture
- Separation of compute and storage
- Pause compute capacity while leaving data intact, so you only pay for storage.
- Resume compute capacity during operational hours



Azure SQL DW – What to expect

- Data Warehouse Units (DWUs) – Blended measure of CPU, memory, and IO
- Higher DWUs imply better performance
- Database Throughput Unit (DTU) are a similar performance measure but for Azure SQL DB service
- Data Warehousing solution when storage needs go beyond 4 TB
- Create Database:
 - <https://docs.microsoft.com/en-us/sql/t-sql/statements/create-database-transact-sql?view=azure-sqldw-latest>

```
CREATE DATABASE database_name [ COLLATE collation_name ]
(
    [ MAXSIZE = {
        250 | 500 | 750 | 1024 | 5120 | 10240 | 20480 | 30720
        | 40960 | 51200 | 61440 | 71680 | 81920 | 92160 | 102400
        | 153600 | 204800 | 245760
    } GB ,
]
    EDITION = 'datawarehouse',
    SERVICE_OBJECTIVE = {
        'DW100' | 'DW200' | 'DW300' | 'DW400' | 'DW500' | 'DW600'
        | 'DW1000' | 'DW1200' | 'DW1500' | 'DW2000' | 'DW3000' | 'DW6000'
        | 'DW1000c' | 'DW1500c' | 'DW2000c' | 'DW2500c' | 'DW3000c' | 'DW5000c'
        | 'DW6000c' | 'DW7500c' | 'DW10000c' | 'DW15000c' | 'DW30000c'
    }
)
[;]
```

IaaS vs PaaS

	Azure SQL Database	SQL Server in an Azure Virtual Machine
Best for:	New cloud-designed applications that have time constraints in development and marketing.	Existing applications that require fast migration to the cloud with minimal changes. Rapid development and test scenarios when you do not want to buy on-premises non-production SQL Server hardware.
	Teams that need built-in high availability, disaster recovery, and upgrade for the database.	Teams that can configure and manage high availability, disaster recovery, and patching for SQL Server. Some provided automated features dramatically simplify this.
	Teams that do not want to manage the underlying operating system and configuration settings.	You need a customized environment with full administrative rights.
	Databases of up to 4 TB, or larger databases that can be horizontally or vertically partitioned using a scale-out pattern.	SQL Server instances with up to 64 TB of storage. The instance can support as many databases as needed.
Resources:	You do not want to employ IT resources for configuration and management of the underlying infrastructure, but want to focus on the application layer.	You have some IT resources for configuration and management. Some provided automated features dramatically simplify this.
Total cost of ownership:	Eliminates hardware costs and reduces administrative costs.	Eliminates hardware costs.
Business continuity:	In addition to built-in fault tolerance infrastructure capabilities, Azure SQL Database provides features, such as automated backups , Point-In-Time Restore , geo-restore , and active geo-replication to increase business continuity. For more information, see SQL Database business continuity overview .	SQL Server on Azure VMs lets you set up a high availability and disaster recovery solution for your database's specific needs. Therefore, you can have a system that is highly optimized for your application. You can test and run failovers by yourself when needed. For more information, see High Availability and Disaster Recovery for SQL Server on Azure Virtual Machines .
Hybrid cloud:	Your on-premises application can access data in Azure SQL Database.	With SQL Server on Azure VMs, you can have applications that run partly in the cloud and partly on-premises. For example, you can extend your on-premises network and Active Directory Domain to the cloud via Azure Virtual Network . In addition, you can store on-premises data files in Azure Storage using SQL Server Data Files in Azure . For more information, see Introduction to SQL Server 2014 Hybrid Cloud .
	Supports SQL Server transactional replication as a subscriber to replicate data.	Fully supports SQL Server transactional replication , AlwaysOn Availability Groups , Integration Services, and Log Shipping to replicate data. Also, traditional SQL Server backups are fully supported

Performance guidelines for SQL on IaaS (VM)

Area	Optimizations
VM size	DS3_v2 or higher for SQL Enterprise edition.
Storage	<p>Use Premium Storage. Standard storage is only recommended for dev/test.</p> <p>Keep the storage account and SQL Server VM in the same region.</p> <p>Disable Azure geo-redundant storage (geo-replication) on the storage account.</p>
Disks	<p>Use a minimum of 2 P30 disks (1 for log files and 1 for data files including TempDB).</p> <p>Avoid using operating system or temporary disks for database storage or logging.</p> <p>Enable read caching on the disk(s) hosting the data files and TempDB data files.</p> <p>Do not enable caching on disk(s) hosting the log file.</p> <p>Important: Stop the SQL Server service when changing the cache settings for an Azure VM disk.</p> <p>Stripe multiple Azure data disks to get increased IO throughput.</p>
Feature-specific	Back up directly to blob storage.

AZURE SQL DATABASE



Azure SQL Database

Single

Database-scoped deployment option
with predictable workload performance

Elastic Pool

Shared resource model optimized for greater
efficiency of multi-tenant applications

Managed Instance (lift & Shift)

Instance-scoped deployment option with high
compatibility with SQL Server and full PaaS
benefits

Best for
apps that require resource
guarantee at database level

Best for
SaaS apps with multiple databases that
can share resources at database level,
achieving better cost efficiency

Best for
modernization at scale with
low friction and effort

Provisioning Azure SQL DB – PowerShell

Click Try It in the upper right corner of a code block.

Azure PowerShell

Copy

Try It

```
# Create a server with a system wide unique server name
$server = New-AzureRmSqlServer -ResourceGroupName $resourcegroupname
  -ServerName $servername
  -Location $location
  -SqlAdministratorCredentials $(New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList

# Create a server firewall rule that allows access from the specified IP range
$serverfirewallrule = New-AzureRmSqlServerFirewallRule -ResourceGroupName $resourcegroupname
  -ServerName $servername
  -FirewallRuleName "AllowedIPs" -StartIpAddress $startip -EndIpAddress $endip

# Create a blank database with an S0 performance level
$database = New-AzureRmSqlDatabase -ResourceGroupName $resourcegroupname
  -ServerName $servername
  -DatabaseName $databasename
  -RequestedServiceObjectiveName "S0"
  -SampleName "AdventureworksLT"

# Clean up deployment
# Remove-AzureRmResourceGroup -ResourceGroupName $resourcegroupname
```

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-powershell-samples>

PAY ONLY FOR WHAT YOU NEED

DTUs			vCores	
Basic	Standard	Premium	General Purpose	Business Critical
Small databases particularly those in development phases	General purpose databases with moderate performance requirements	Mission-critical databases with high performance and high-availability requirements	Data applications with basic IO and basic availability requirements	Business critical data applications with fast IO and high availability requirements



Elastic scale and performance: Three service tiers within DTU-based model, and two tiers within vCore-based model let you scale up and down based on throughput needs, and offer better resource isolation and an improved billing experience



Business continuity and data protection: A spectrum of business-continuity features across tiers lets you dial up control over data recovery and failover



Familiar and fully-managed: Near-complete SQL Server compatibility and unprecedented efficiencies as your applications scale with a near-zero maintenance service and a variety of familiar management tools and programmatic APIs

DTU-BASED SINGLETON SERVICE TIERS

	Basic	Standard									Premium																																							
		S0	S1	S2	S3	S4	S6	S7	S9	S12	P1	P2	P4	P6	P11	P15																																		
Built for	Light transactional workloads	Medium transactional workloads									Heavy transactional workloads																																							
Available SLA	99.99%*																																																	
Database max size	2 GB	250 GB		1 TB				4 TB																																										
Point-in-time restore ("oops" recovery)	Any point within 7 days	Any point within 35 days																																																
Business continuity	Active geo-replication, up to four readable secondary backups																																																	
Security	Auditing, row-level security, dynamic data masking, Advanced Threat Protection																																																	
Performance objectives	Transactions per hour	Transactions per minute									Transactions per second																																							
Database transaction units (DTUs)	5	10	20	50	100	200	400	800	1,600	3,000	125	250	500	1,000	1,750	4,000																																		

\$ ← → \$\$\$

*The 99.99% availability SLA does not apply to the existing Web and Business editions, which will continue to be supported at 99.9% availability.

VCORE MODEL

Database Transaction Unit (DTU) model

Pre-packaged, bundled unit that represents the database power

Designed for predictable performance, but somewhat inflexible and limited in options

DTU sizing offers simplicity of choice

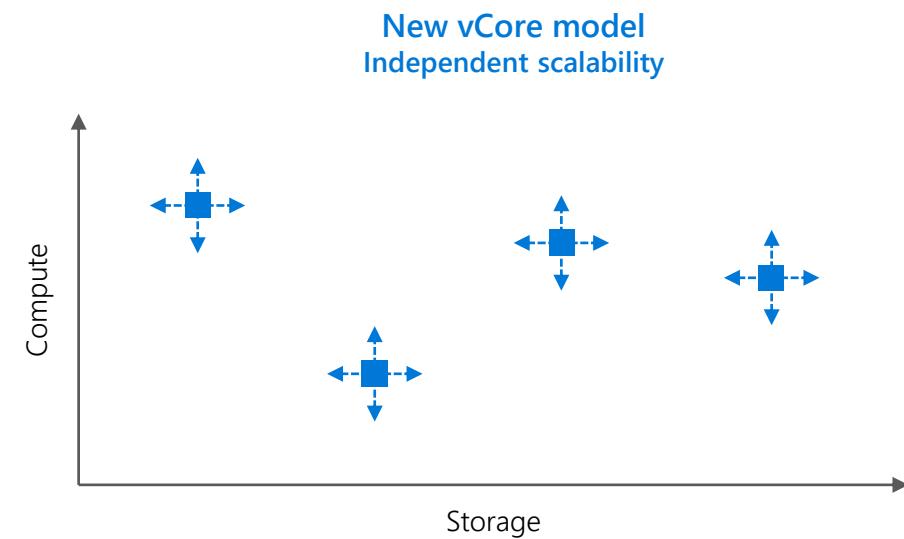
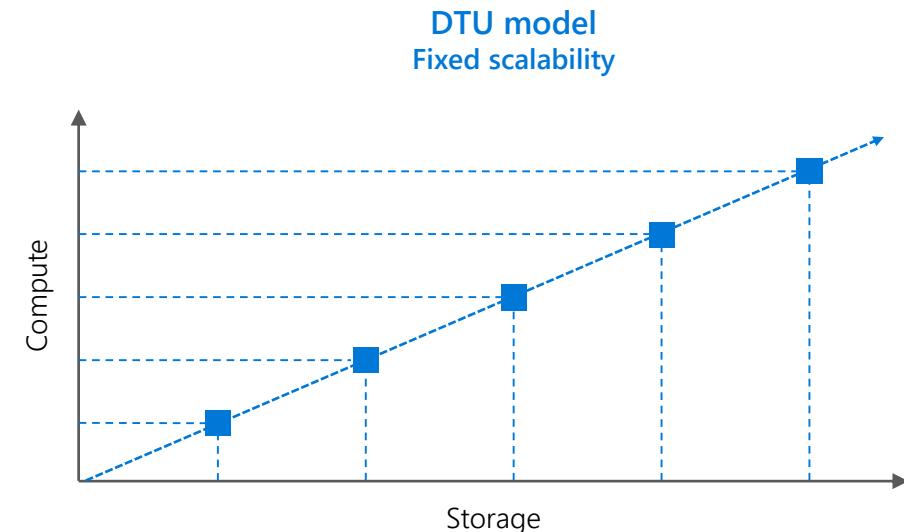
vCore model

À la carte approach deconstructs the DTU model into separate parts

Customers can select compute and storage independently

Allows customers to right-size their compute requirements in the cloud

vCore sizing offers flexibility of choice



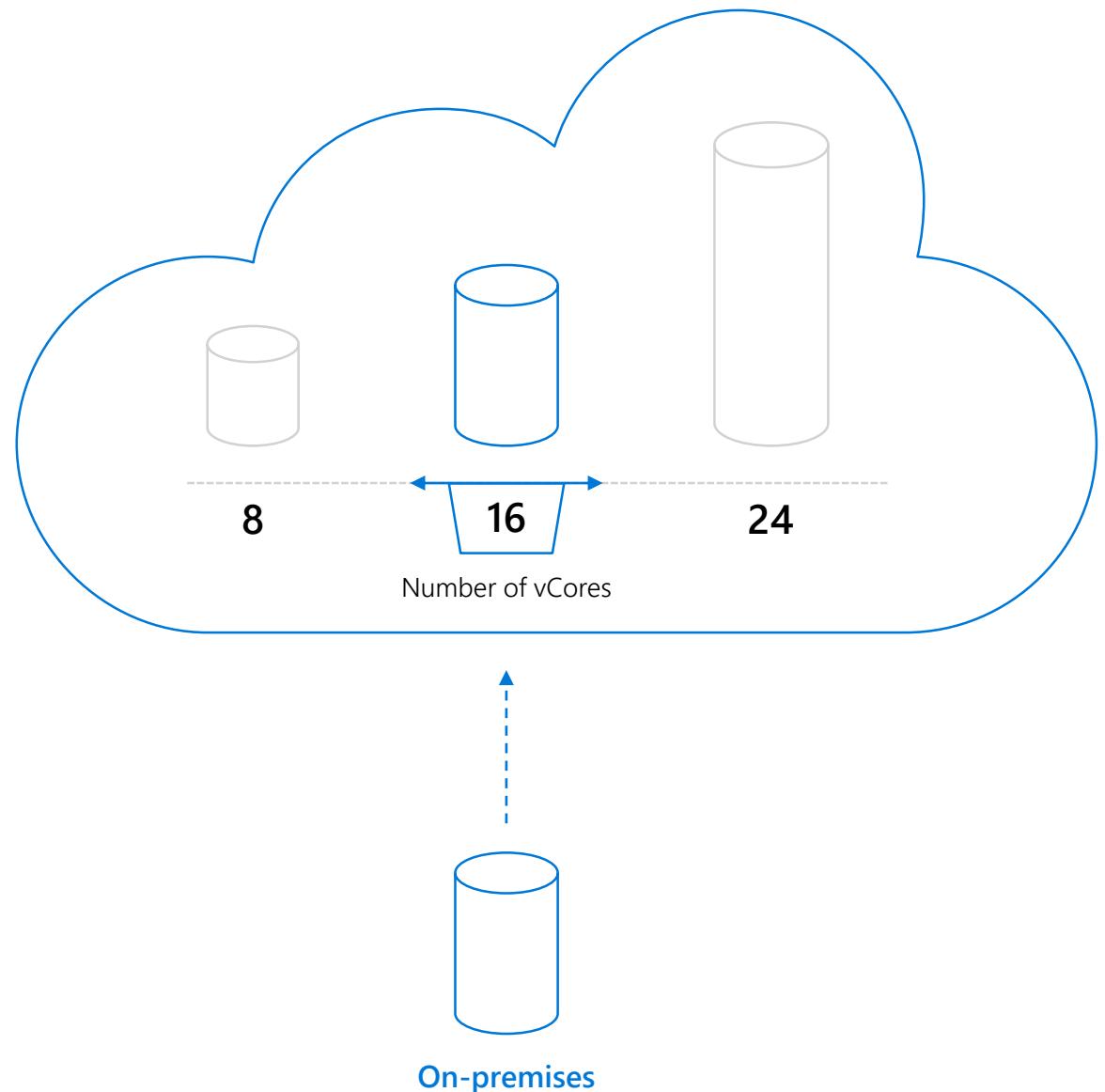
BENEFITS OF VIRTUAL CORES

Choice between vCores and DTUs in Azure SQL
Database as a unit of measure for available CPU

Understand your compute requirements in the cloud
vs. what you use on-premises today

Easier to right-size the destination environment by
removing the guesswork of DTUs

1 on-premises core = 1 vCore on Gen4 hardware



PostgreSQL & MySQL

- Migrate your PostgreSQL database using dump and restore
- You can use pg_dump to extract a PostgreSQL database into a dump file and pg_restore to restore the PostgreSQL database from an archive file created by pg_dump.
- <https://docs.microsoft.com/en-us/azure/postgresql/howto-migrate-using-dump-and-restore>

Lab 01

Provision SQL IaaS & Azure SQL DB

<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sql/quickstart-sql-vm-create-portal>

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-get-started-portal>

2. Design and Implement security (25–30%)

Objectives

Design and implement SQL Server Database security

Configure firewalls

Manage logins, users, and roles

Assign permissions

Configure auditing

Configure transparent database encryption

Configure row-level security

Configure data encryption

Configure data masking

Configure Always Encrypted

Objectives (continued)

Design and Implement Azure SQL Database security

Configure firewalls

Manage logins, users, and roles

Assign permissions

Configure auditing

Configure row-level security

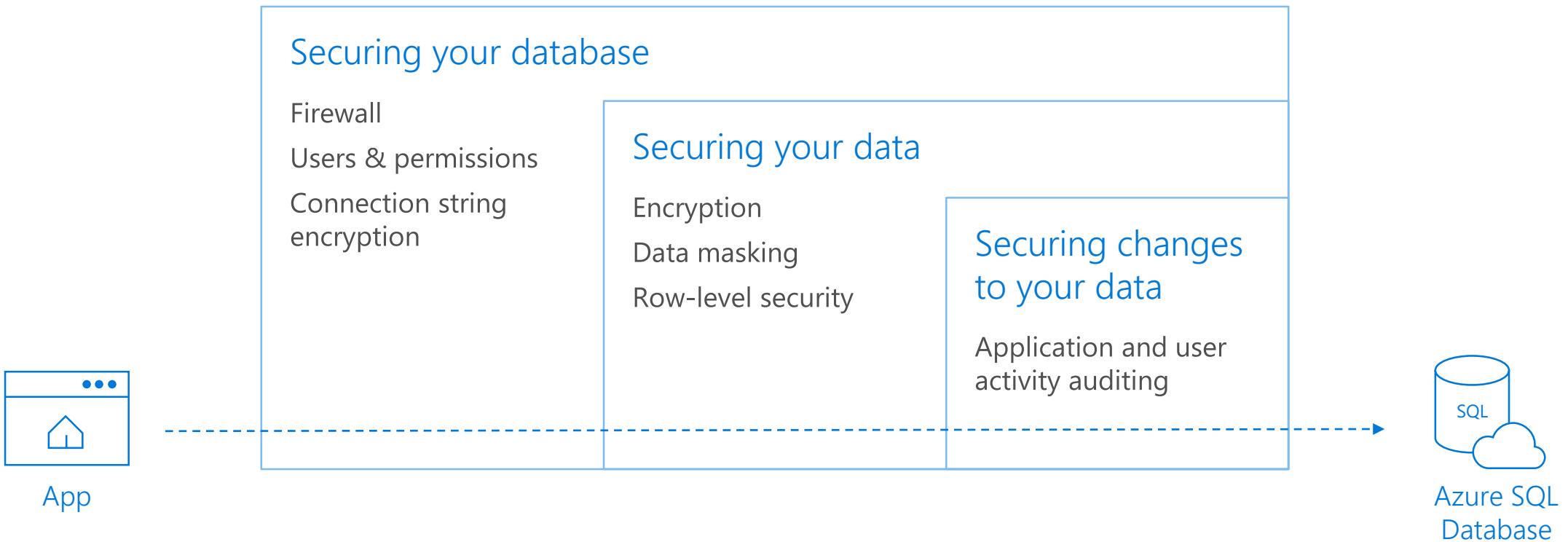
Configure data encryption

Configure data masking

Configure Always Encrypted

Configure Automatic Threat Detection

LAYERED APPROACH TO SECURITY

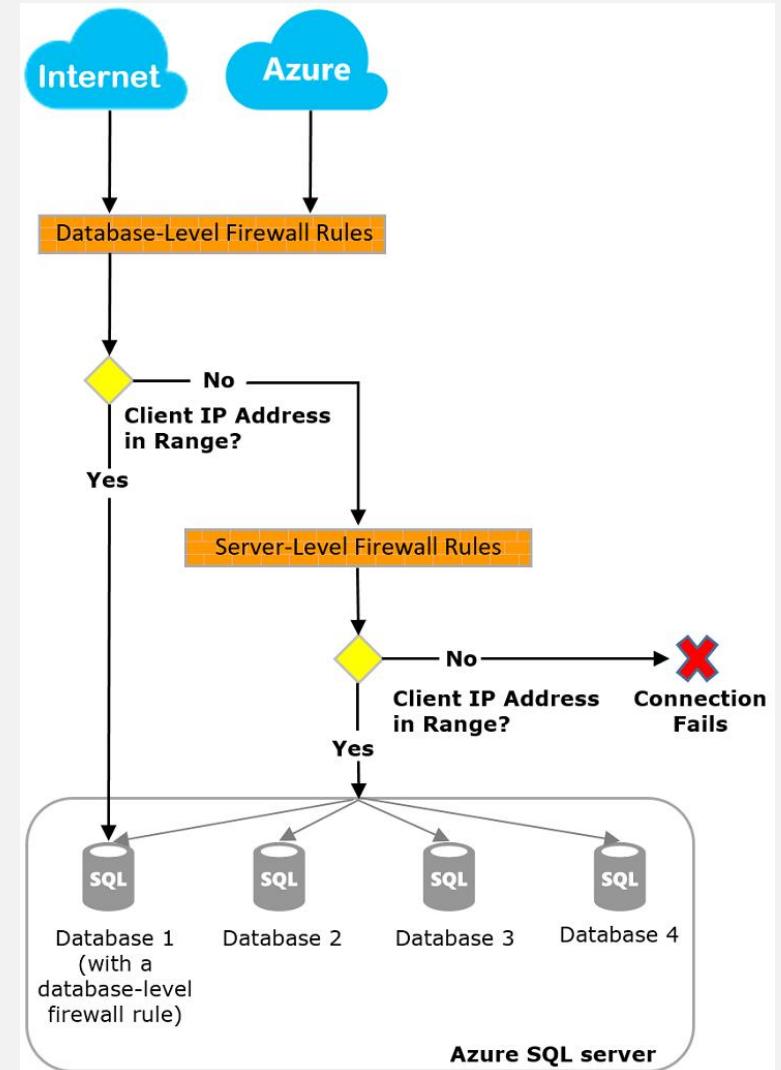


SECURING YOUR DATABASE WITH FIREWALLS

Initially, all access to your Azure SQL Database server is blocked by the firewall

In order to begin using your Azure SQL Database server, you must go to the Management Portal

Database Level Firewall rules can only be created through T-SQL ([sp_set_database_firewall_rule](#))



FIREWALL CONFIGURATION EXAMPLE

By default, Azure blocks all external connections to port 1433

Enable in the following ways in Azure portal:

Classic portal: Server level – **Configure** page

New portal: **Settings > Firewall > Firewall settings** blade

RULE NAME	START IP	END IP	
ClientIPAddress_2015-02-02...	50.200.196.194	50.200.196.194	...
ClientIPAddress_2014-12-05...	131.107.160.14	131.107.160.14	...
ClientIPAddress_2015-58-29...	67.170.61.126	67.170.61.126	...

Firewall configuration using PowerShell / T-SQL

```
# PS Enable Azure connections
PS C:\> New-AzureSqlDatabaseServerFirewallRule ` 
    -ServerName "Contoso" ` 
    -AllowAllAzureServices ` 
    -RuleName "myRule2"

-- PS Allow external IP access to SQL Database
PS C:\> New-AzureSqlDatabaseServerFirewallRule ` 
    -ServerName "Contoso" ` 
    -RuleName "myRule1" ` 
    -StartIpAddress 12.1.1.1 ` 
    -EndIpAddress 12.1.1.2

-- T-SQL Enable Azure connections
sp_set_firewall_rule N'Allow Windows Azure',
    '0.0.0.0','0.0.0.0'

-- T-SQL Allow external IP access to SQL Database
sp_set_firewall_rule N'myRule1',
    '12.1.1.1','12.1.1.2'
```

Manage SQL Database firewall rules using code.

Windows PowerShell Azure cmdlets

New-AzureSqlDatabaseServerFirewallRule
Get-AzureSqlDatabaseServerFirewallRule
Set-AzureSqlDatabaseServerFirewallRule

Transact SQL

sp_set_firewall_rule
sp_set_database_firewall_rule
sp_delete_firewall_rule
sp_delete_database_firewall_rule

Firewall configuration using REST API

POST

```
https://management.core.windows.net:8443/{subscriptionId}/services/sqlservers/servers/Contoso/firewallrules
```

REQUEST BODY

```
<ServiceResource  
xmlns="http://schemas.microsoft.com/windowsazure">  
<Name>myRule1</Name>  
  <StartIPAddress> 12.1.1.1 </StartIPAddress>  
  <EndIPAddress> 12.1.1.1 </EndIPAddress>  
</ServiceResource>
```

DELETE

```
https://management.core.windows.net:8443/{subscriptionId}/services/sqlservers/servers/Contoso/firewallrules/myRule1
```

GET

```
https://management.core.windows.net:8443/{subscriptionId}/services/sqlservers/servers/Contoso/firewallrules
```

Managing firewall rules through REST API must be authenticated. For information, see [Authenticating Service Management Requests](#).

Server-level rules can be created, updated, or deleted using REST API

To create or update a server-level firewall rule, execute the POST method

To remove an existing server-level firewall rule, execute the DELETE method

To list firewall rules, execute the GET

ROW LEVEL SECURITY - CREATE A SECURITY POLICY

Create a security policy for row-level security

The following examples demonstrate the use of the CREATE SECURITY POLICY syntax

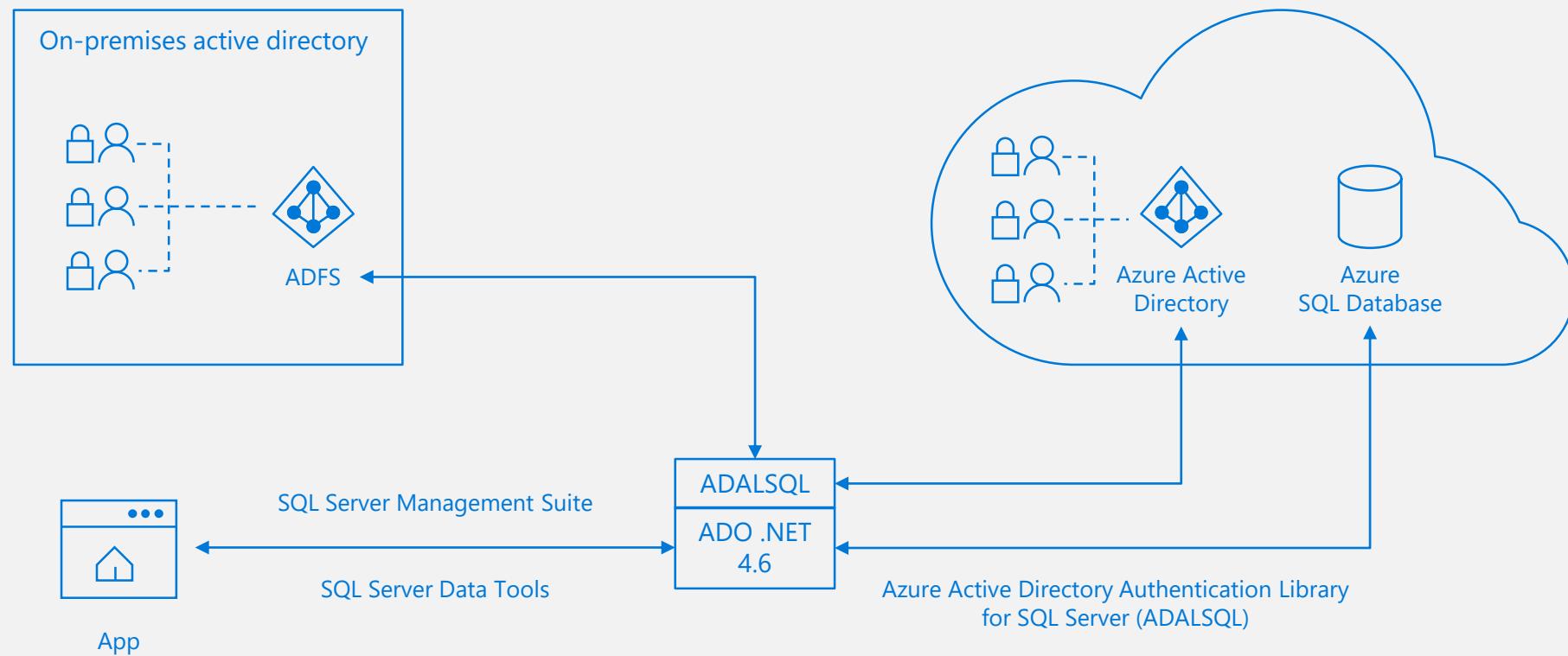
For an example of a complete security policy scenario, see [Row-Level Security](#)

```
-- Create a new schema and predicate function,  
which will use the user name to filter rows.  
CREATE FUNCTION  
Security.fn_securitypredicate(@SalesRep AS sysname)  
    RETURNS TABLE  
WITH SCHEMABINDING  
AS  
    RETURN SELECT 1 AS fn_securitypredicate_result  
WHERE @SalesRep = USER_NAME() OR USER_NAME() =  
'Manager';  
GO
```

```
-- The following syntax creates a security policy  
with a filter predicate for the Sales table  
CREATE SECURITY POLICY SalesFilter  
ADD FILTER PREDICATE  
Security.fn_securitypredicate(SalesRep)  
ON dbo.Sales  
WITH (STATE = ON);
```

TRUST ARCHITECTURE

Azure Active Directory and Azure SQL Database



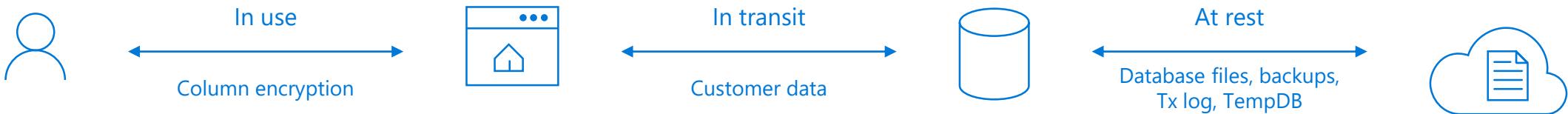
Overview SQL Database Security Administration

Point of Difference	On-premises/Azure VM SQL Server	Microsoft Azure SQL Database
Where you manage server-level security	The Security folder in SQL Server Management Studio's Object Explorer	The master database
Server-level security role for creating logins	securityadmin fixed server role	loginmanager database role in the master database (recommended to use contained dbs instead)
Commands for managing logins	CREATE LOGIN ALTER LOGIN DROP LOGIN	CREATE LOGIN ALTER LOGIN DROP LOGIN (parameter limitations and must be connected to master db)
View that shows all logins	sys.syslogins (sys.sql_logins for SQL Server authentication logins)	sys.sql_logins (You must be connected to the master database)
Server-level role for creating databases	dbcreator fixed database role	dbmanager database role in the master database
Database-level roles	db_owner , db_ddladmin , db_datareader	db_owner , db_ddladmin , db_datareader

<https://azure.microsoft.com/en-us/documentation/articles/sql-database-manage-logins/>

TYPES OF DATA ENCRYPTION

Data encryption	Encryption technology	Customer value
In transit	Transport Layer Security (TLS) from the client to the server	Protects data between client and server against snooping and man-in-the-middle attacks *Azure SQL Database is phasing out Secure Sockets Layer (SSL) 3.0 and TLS 1.0 in favor of TLS 1.2
At rest	Transparent Data Encryption (TDE) for Azure SQL Database	Protects data on the disk Key management is done by Azure, which makes it easier to obtain compliance
In use (end-to-end)	Always Encrypted for client-side column encryption	Data is protected end-to-end, but the application is aware of encrypted columns This is used in the absence of data masking and TDE for compliance-related scenarios



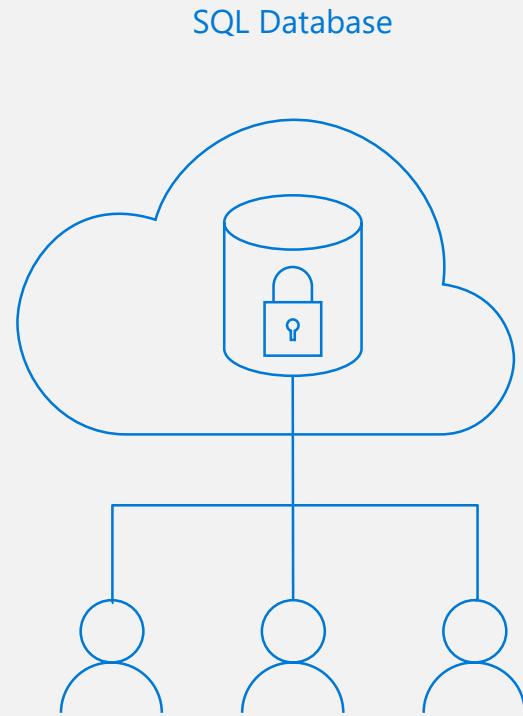
TRANSPARENT DATA ENCRYPTION

All customer data encrypted at rest

Encryption keys managed by Azure

Application changes kept to a minimum

Support for equality operations (including joins) on encrypted data



ALWAYS ENCRYPTED

Overview

Protect data at rest and in motion, on premises and in the cloud

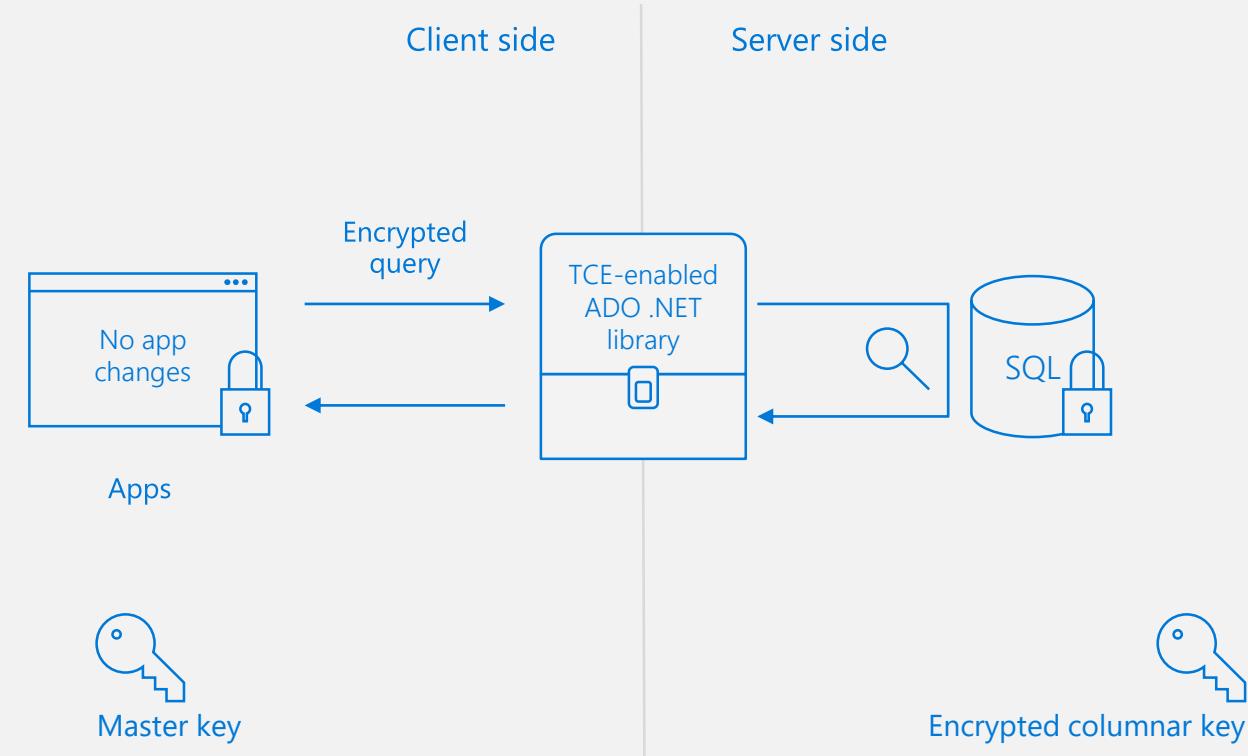
Transparent client-side encryption, while SQL Server executes T-SQL queries on encrypted data

Benefits

Sensitive data remains encrypted and queryable at all times on-premises and in the cloud

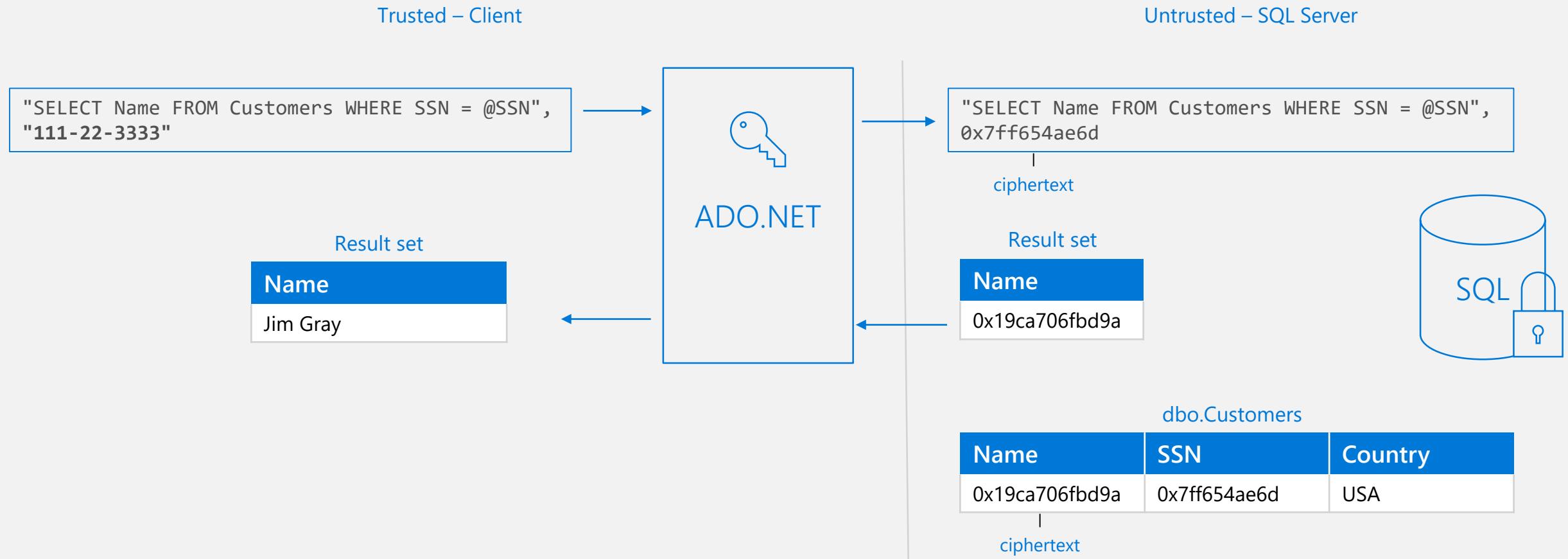
Unauthorized users never have access to data or keys

No application changes



HOW ALWAYS ENCRYPTED WORKS

Encrypted sensitive data and its corresponding keys are never seen in plaintext in SQL Server



ALWAYS ENCRYPTED - PROTECT DATA FROM HIGH-PRIVILEGED, UNAUTHORIZED USERS

Client-side encryption

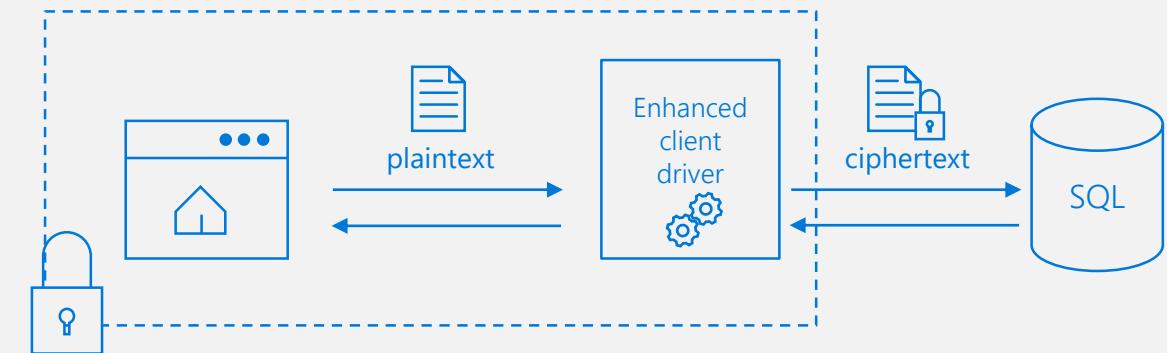
Sensitive data and related encryption keys are never revealed to the database engine

Encryption transparency

Client driver transparently encrypts query parameters and decrypts encrypted results

Queries on encrypted data

Support for equality comparison on columns encrypted using deterministic encryption



Always Encrypted

Deterministic vs Randomized

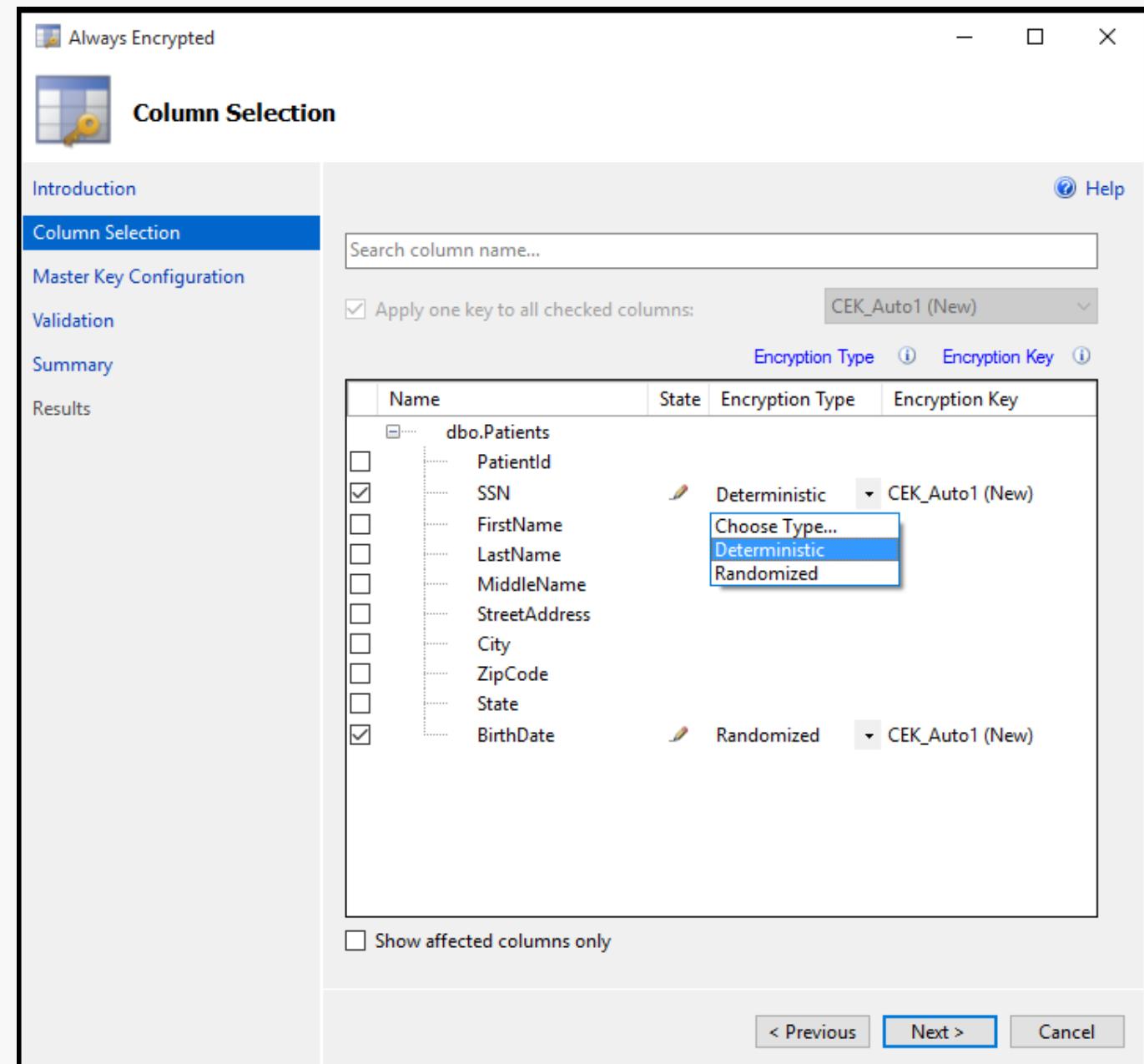
Deterministic encryption must use a column collation with a binary2 sort order for character columns

Deterministic is weaker for small sets of possible values (ie. yes/no)

Column master keys

Column encryption keys

<https://msdn.microsoft.com/library/mt459280.aspx>



T-SQL Implementation

Column master key

- Key-protecting key that encrypts one or more column encryption keys.
- Stored in external trusted key stores

Column encryption key

- Encrypts the column

```
CREATE COLUMN MASTER KEY MyCMK
WITH (
    KEY_STORE_PROVIDER_NAME = 'MSSQL_CERTIFICATE_STORE',
    KEY_PATH = 'Current User/Personal/f2260f28d909d21c642a3d8e0b45a830e79a1420'
);

-----
CREATE COLUMN ENCRYPTION KEY MyCEK
WITH VALUES
(
    COLUMN_MASTER_KEY = MyCMK,
    ALGORITHM = 'RSA_OAEP',
    ENCRYPTED_VALUE = 0x01700000016C006F00630061006C006D0061006300680069006E0065002F006D0079002F00320066006100
);

-----
CREATE TABLE Customers (
    CustName nvarchar(60)
        COLLATE Latin1_General_BIN2 ENCRYPTED WITH (COLUMN_ENCRYPTION_KEY = MyCEK,
        ENCRYPTION_TYPE = RANDOMIZED,
        ALGORITHM = 'AEAD_AES_256_CBC_HMAC_SHA_256'),
    SSN varchar(11)
        COLLATE Latin1_General_BIN2 ENCRYPTED WITH (COLUMN_ENCRYPTION_KEY = MyCEK,
        ENCRYPTION_TYPE = DETERMINISTIC ,
        ALGORITHM = 'AEAD_AES_256_CBC_HMAC_SHA_256'),
    Age int NULL
);
GO
```

DYNAMIC DATA MASKING

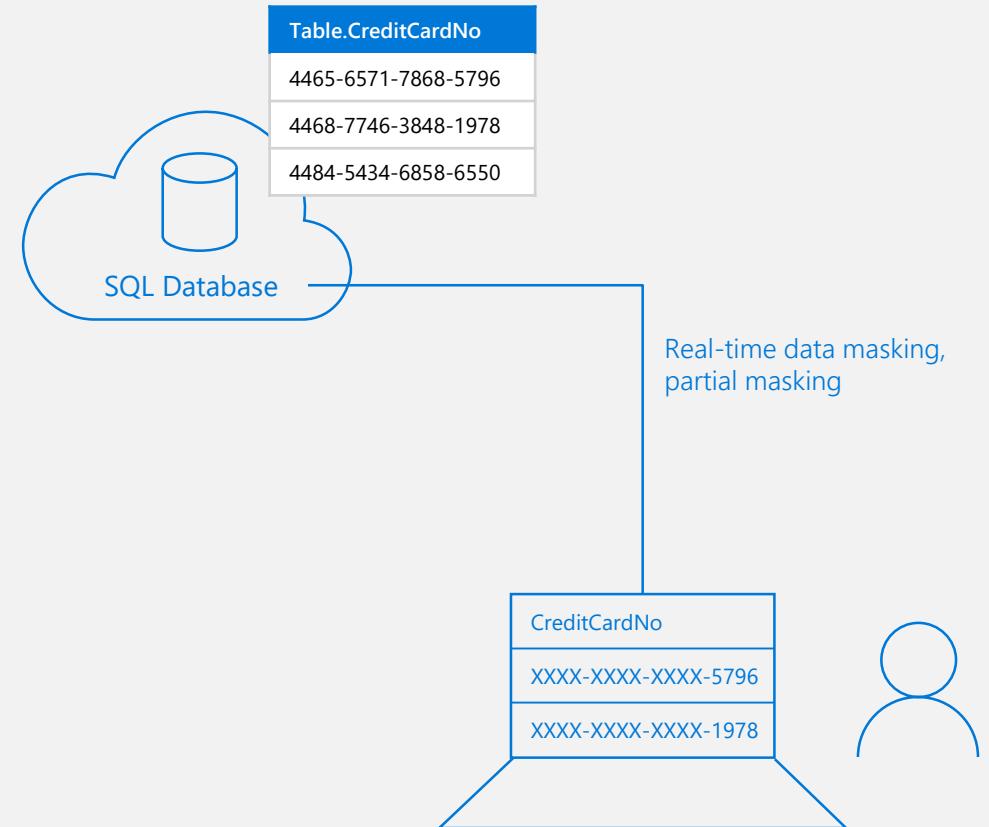
Prevent abuse of sensitive data by hiding it from users

Policy-driven at table and column level, for a defined set of users

Data masking applied in real-time to query results based on policy

Multiple masking functions available, such as full or partial, for various sensitive data categories (credit card numbers, SSN, etc.)

Easy configuration in new Azure Portal

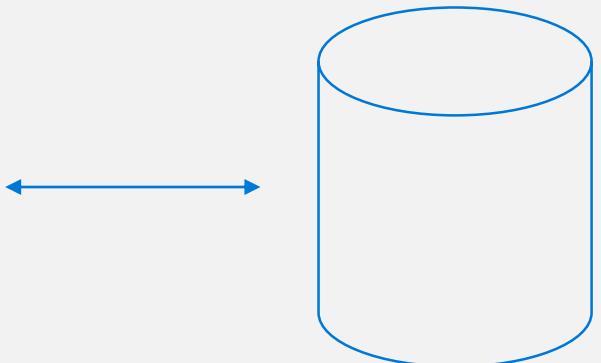


DYNAMIC DATA MASKING

1. Security officer defines dynamic data masking policy in T-SQL over sensitive data in the Employee table
2. The app user selects from the Employee table
3. The dynamic data masking policy obfuscates the sensitive data in the query results



Business app



```
ALTER TABLE [Employee] ALTER COLUMN [SocialSecurityNumber]
ADD MASKED WITH (FUNCTION = 'SSN()')

ALTER TABLE [Employee] ALTER COLUMN [Email]
ADD MASKED WITH (FUNCTION = 'EMAIL()')

ALTER TABLE [Employee] ALTER COLUMN [Salary]
ADD MASKED WITH (FUNCTION = 'RANDOM(1,20000)')

GRANT UNMASK to admin1
```



Security officer

```
SELECT [Name],
       [SocialSecurityNumber],
       [Email],
       [Salary]
  FROM [Employee]
```

other logon

	First Name	Social Security Number	Email	Salary
1	LILA	XXX-XX-XX37	IXX@XXXX.net	8940
2	JAMIE	XXX-XX-XX14	jXX@XXXX.com	19582
3	SHELLEY	XXX-XX-XX28	sXX@XXXX.net	3713
4	MARCELLA	XXX-XX-XX65	mXX@XXXX.net	11572
5	GILBERT	XXX-XX-XX87	gXX@XXXX.net	4487

admin1 logon

	First Name	Social Security Num...	Email	Salary
1	LILA	758-10-9637	lila.bamett@comcast.net	1012794
2	JAMIE	113-29-4314	jamie.brown@ntlworld.com	1025713
3	SHELLEY	550-72-2028	shelley.lynn@charter.net	1040131
4	MARCELLA	903-94-5665	marcella.estrada@comcast.net	1040753
5	GILBERT	376-79-4787	gilbert.juarez@verizon.net	1041308

SQL Database Dynamic Data Masking

Configured by the Azure Database admin, server admin, or security officer roles

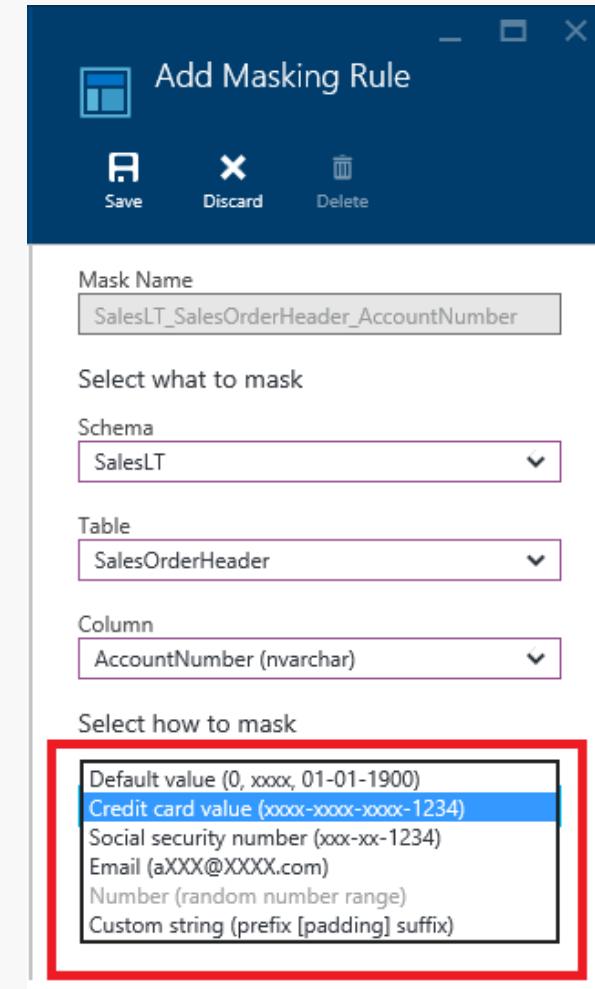
Masking Function

Default, Credit card, Social Security number, Email, Random, Custom text

Masking Field Format

Custom text

Exposed Prefix	Padding String	Exposed Suffix
3	X*X*X	2



<https://azure.microsoft.com/en-us/documentation/articles/sql-database-dynamic-data-masking-get-started/>

Practice Question 1

You are an administrator setting up a new Azure SQL Database that will be deployed on a new server object.

After creating the new server and database, you are unable to connect to it with Management Studio from your workstation.

You need to be able to establish the connection.

What should you do? (any two options)

- A. Add your current public IP address to the Azure SQL Database's firewall settings.
- B. Add your current public IP address to the Azure SQL Database server's firewall settings.
- C. Add and configure Auditing for the Azure SQL Database. False
- D. Add a new user to the Azure SQL Database server's User Access Administrator role.

*Courtesy of MeasureUp

<https://azure.microsoft.com/en-us/documentation/articles/sql-database-firewall-configure/>

Practice Question 1

You are an administrator setting up a new Azure SQL Database that will be deployed on a new server object.

After creating the new server and database, you are unable to connect to it with Management Studio from your workstation.

You need to be able to establish the connection.

What should you do?

- A. Add your current public IP address to the Azure SQL Database's firewall settings.
- B. Add your current public IP address to the Azure SQL Database server's firewall settings.
- C. Add and configure Auditing for the Azure SQL Database.
- D. Add a new user to the Azure SQL Database server's User Access Administrator role.

*Courtesy of MeasureUp

<https://azure.microsoft.com/en-us/documentation/articles/sql-database-firewall-configure/>

Lab 02

Configure RLS & Always Encrypted

<https://docs.microsoft.com/en-us/sql/relational-databases/security/row-level-security?view=sql-server-2017#Typical>
<https://www.red-gate.com/simple-talk/sql/database-administration/sql-server-encryption-always-encrypted/>

3. Design for high availability, disaster recovery, and scalability (25–30%)

Objectives

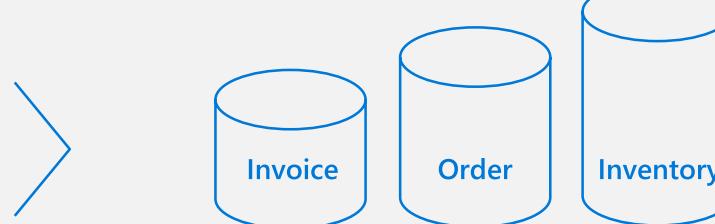
- Design and implement high availability solutions
 - Design a high availability solution topology
 - Design a high-availability solution for SQL in Azure VMs
- Design and implement scalable solutions
 - Design a scale-out solution
 - Implement multi-master scenarios with database replication
 - Implement elastic scale for Azure SQL Database
- Design and implement SQL Database data recovery
 - Implement self-service restore
 - Copy and export databases
 - Implement long-term retention backups

COMMON DATABASE SCALABILITY PATTERNS

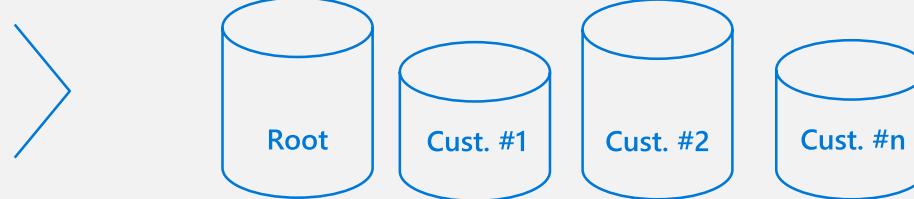
Single large database



Vertically partitioned



1 tenant: 1 database (SaaS ISV)



Hybrid partitioning scheme



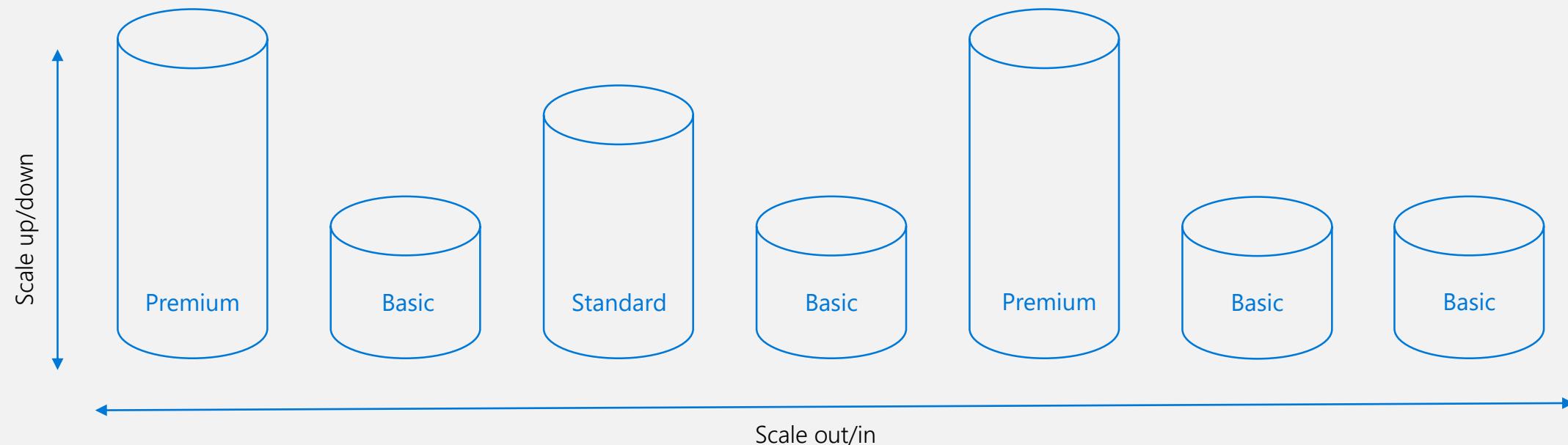
SCALABILITY OPTIONS IN AZURE SQL DATABASE

Vertical: scale up or scale down

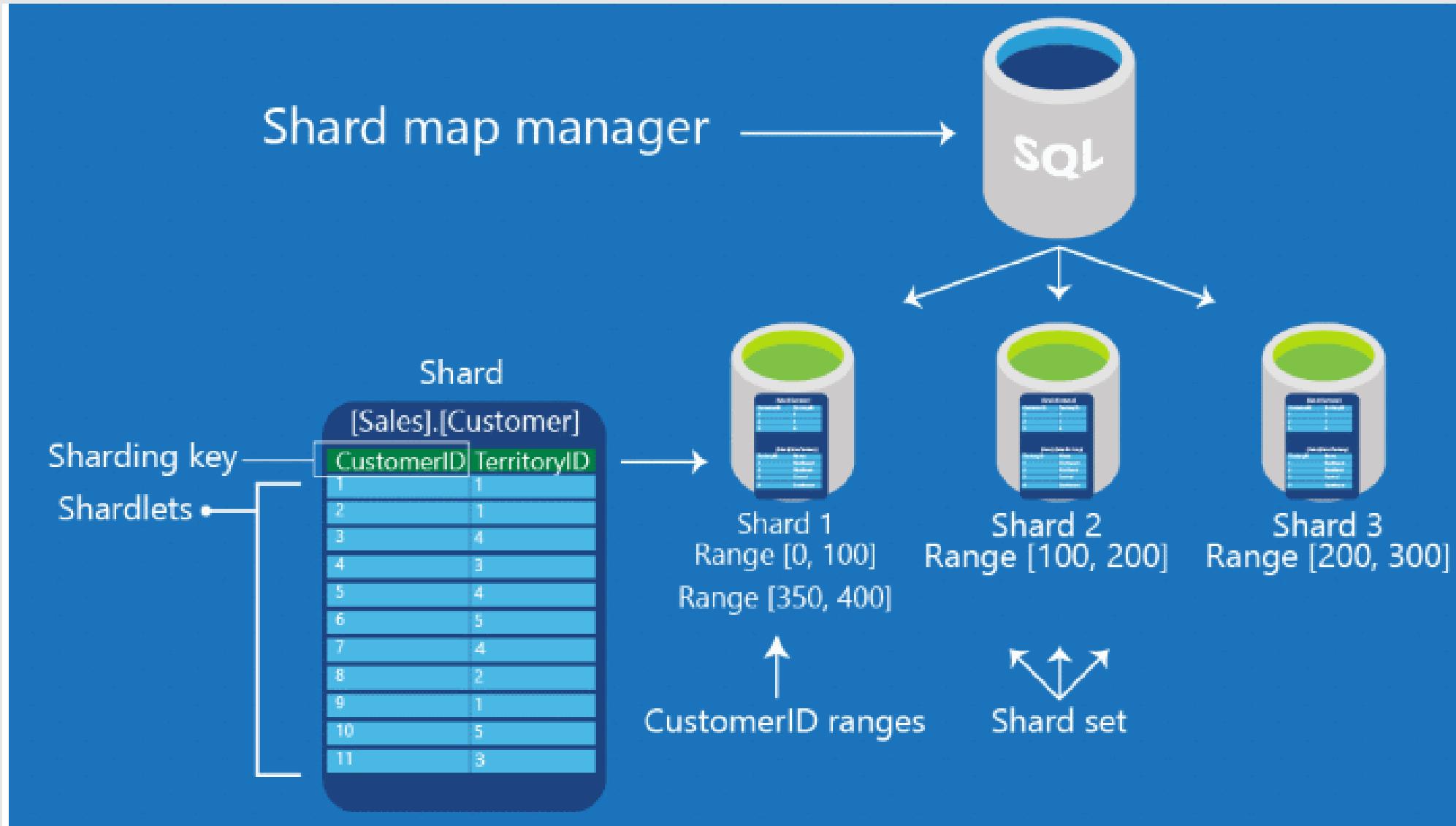
Change service tiers for a given database as capacity needs fluctuate

Horizontal: scale out or scale in

Add or remove databases (sharded and/or in a pool) as more or less capacity is needed



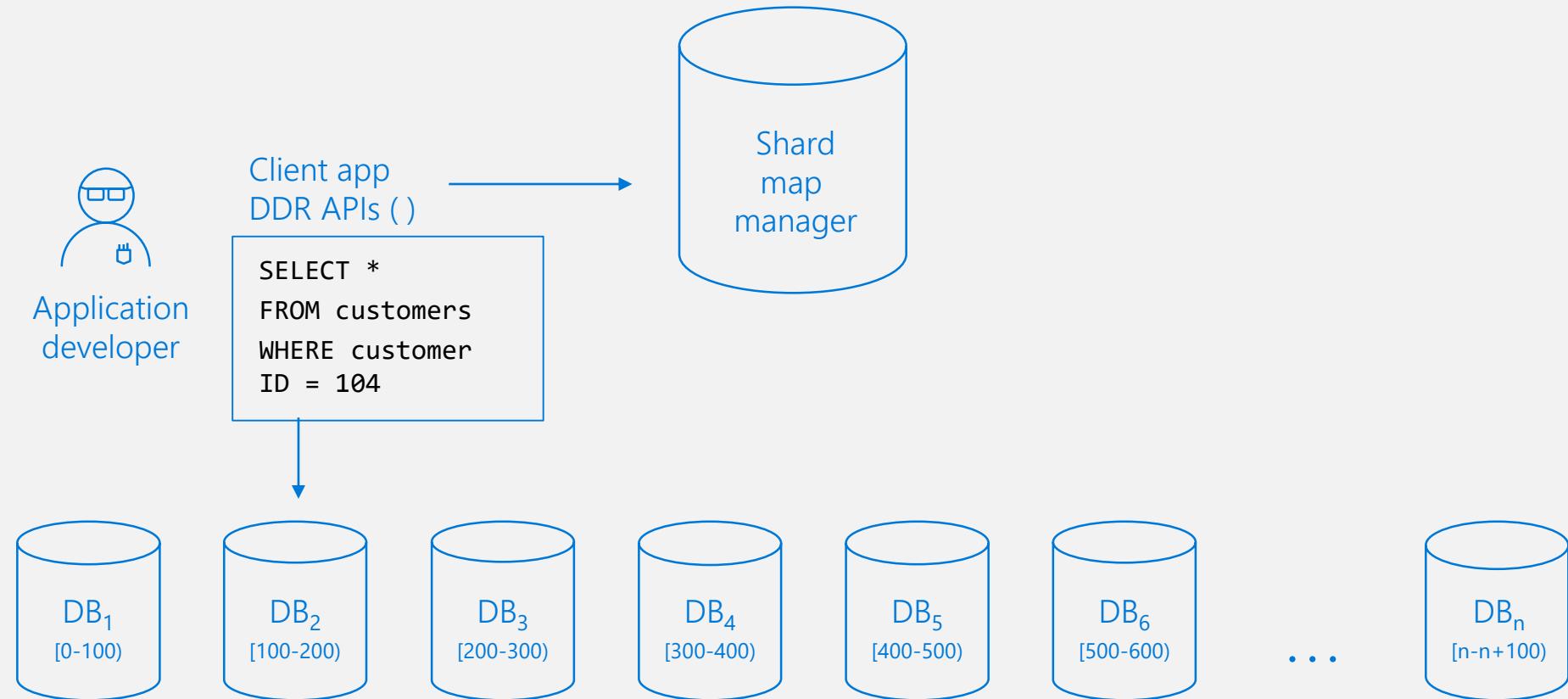
Scale out databases with the shard map manager



DATA DEPENDENT ROUTING (DDR)

Scenario:

Query a shard with a specific shardlet key



DATA DEPENDENT ROUTING (DDR)

```
// Get a routed connection for a given shardingKey
using (SqlConnection conn = ShardMap.OpenConnectionForKey(
    shardingKey,
    connectionString /* Credentials Only */ ,
    ConnectionOptions.Validate /* Validate */ ));
{
    using (SqlCommand cmd = new SqlCommand())
    {
        cmd.Connection = conn;
        cmd.CommandText = "SELECT dbNameField, TestIntField, TestBigIntField FROM ShardedTable";

        SqlDataReader sdr = cmd.ExecuteReader();

        // Now consume results from the data reader...
    }
}
```

Cross Database Operations

Elastic Database Jobs

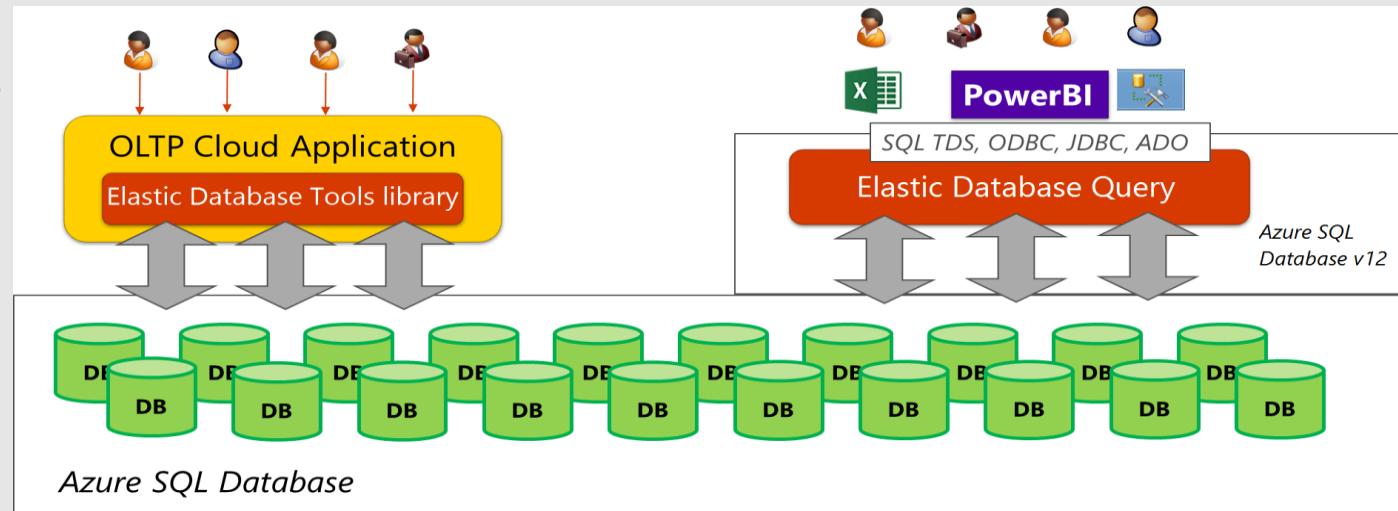
- Run one or more T-SQL scripts in parallel, across a large number of databases, on a schedule or on-demand.
- Use Case: Database Management, Data aggregation, Job Scheduling

Elastic Database Queries

- Run a read-only Transact-SQL query that spans multiple databases in Azure SQL Database.
- Use Case: Reporting/Analytics

Elastic Transactions

- Make atomic changes to data stored in several different SQL Databases.
- Focuses on client-side development experiences in C# and .NET

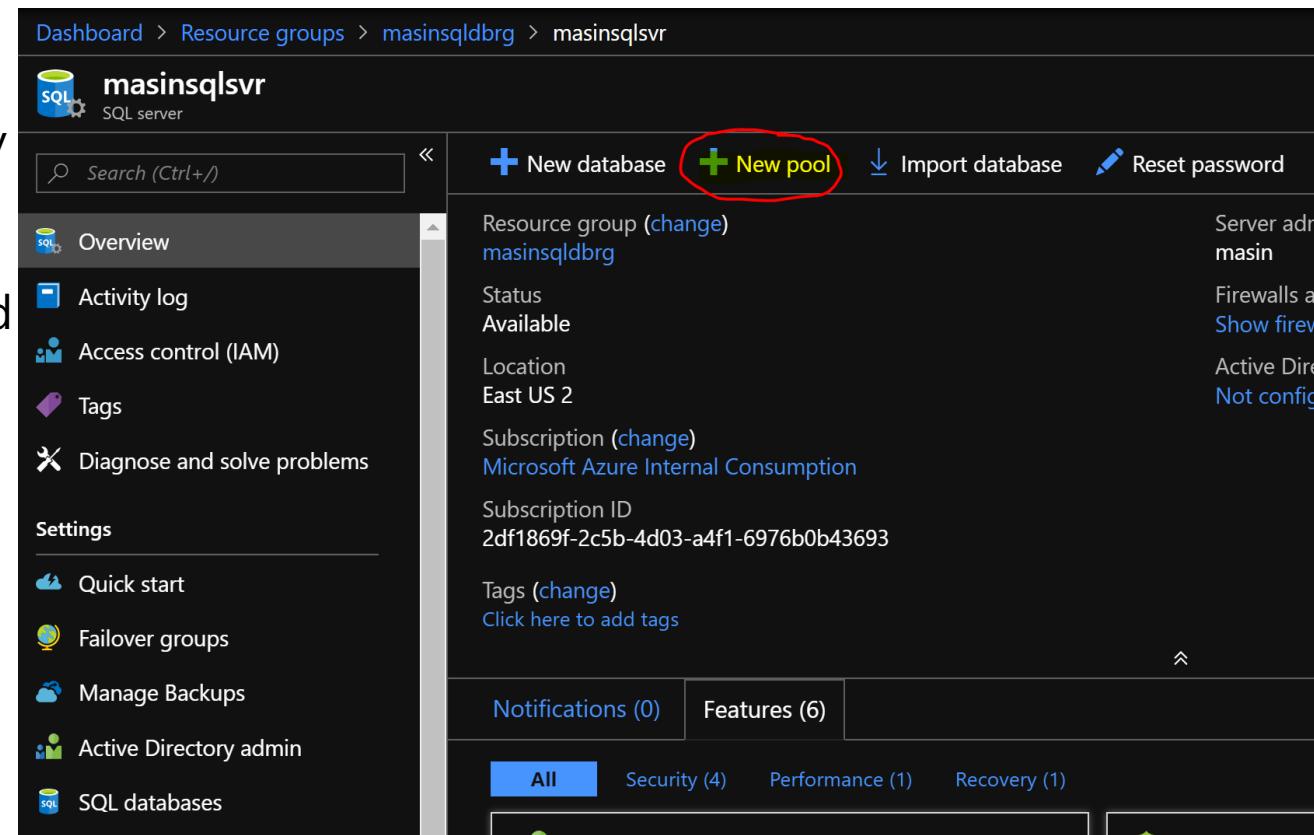


ADDING DATABASES TO ELASTIC POOL

Create an elastic pool by navigating to an existing SQL server and clicking **Create pool**

Configure the resources and pricing of the pool, by clicking **Configure pool**.

Select a service tier, add databases to the pool, and configure the resource limits for the pool and its databases.

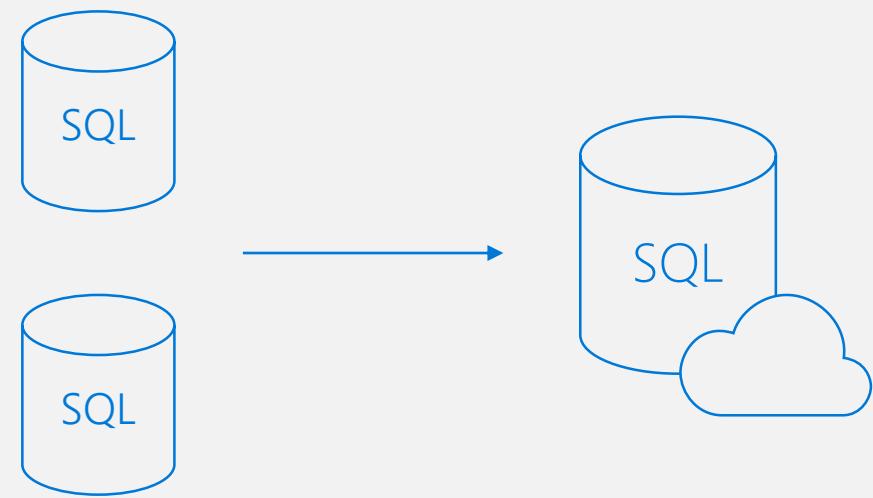


MIGRATING TO AZURE SQL DATABASE

Migrate an on-premises SQL Server database to Azure SQL Database

Simplify migration:

- To migrate a non-SQL Server database, including Microsoft Access, Sybase, MySQL Oracle, and DB2 to Azure SQL Database, see [SQL Server Migration Assistant](#).



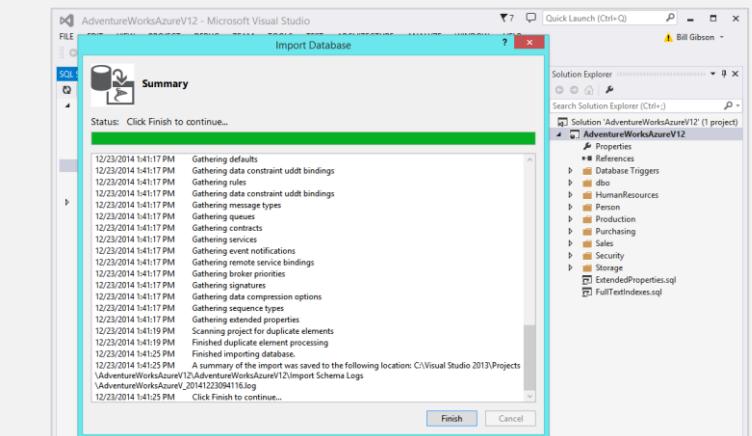
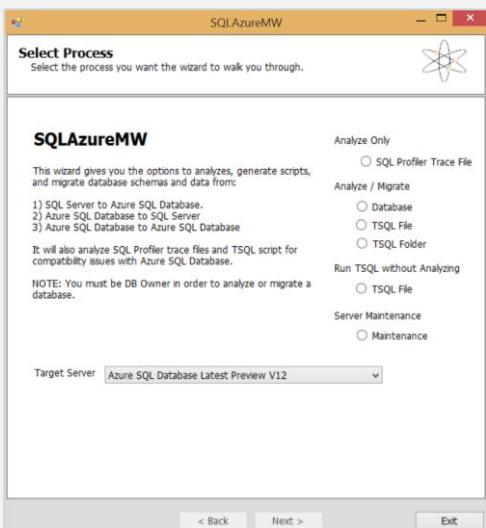
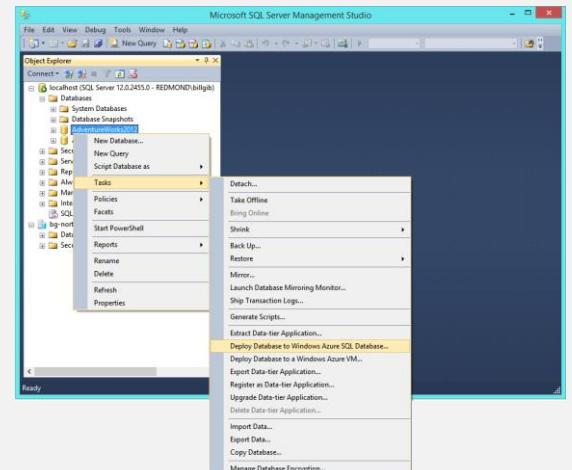
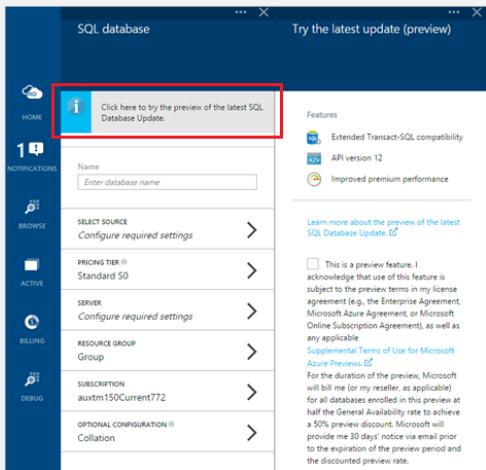
Migration Tools

Azure Management Portal

SQL Server Management Studio (SSMS)

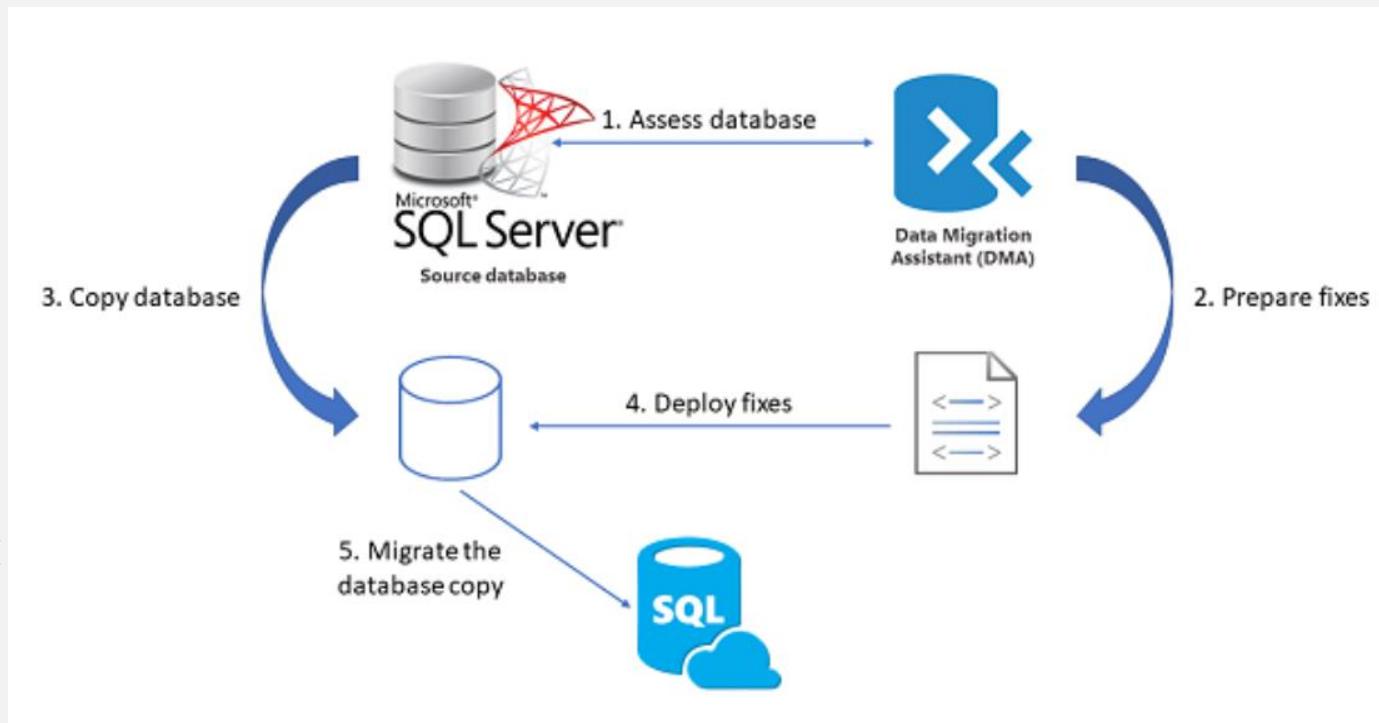
Azure Database Migration Service

Data Migration Assistant



Migration Process

1. Assess database for compatibility issues with DMA
2. Prepare TSQL Script for fixes
3. Create a local copy of the database
4. Deploy fixes to a copy of the database
5. Migrate the database using various options:
 - DMA
 - SSMS
 - DMS

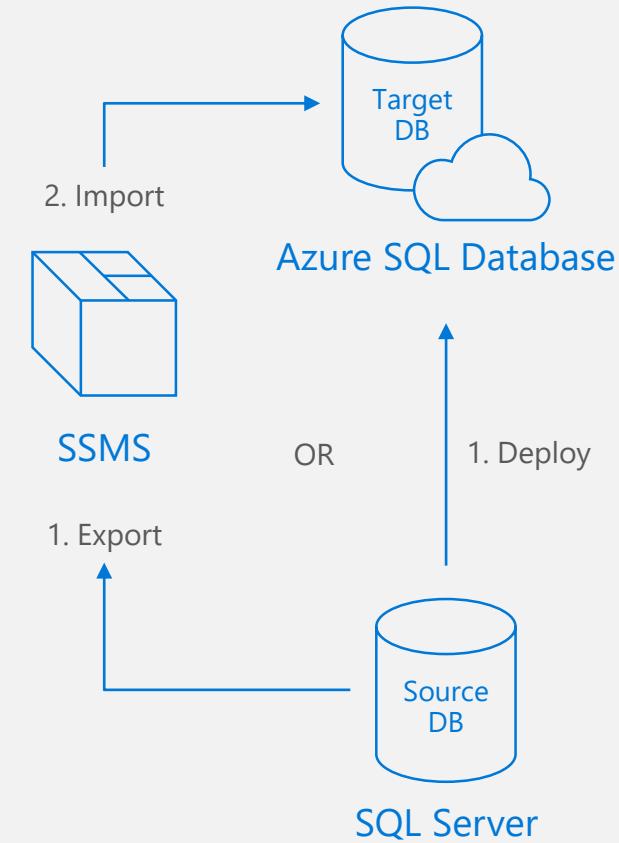


SQL SERVER MANAGEMENT STUDIO

Migrate a compatible database using SQL Server Management Studio

Use SSMS to deploy to Azure SQL Database

Use SSMS to export a BACPAC and then import it to Azure SQL Database



AZURE DATABASE MIGRATION SERVICE – MIGRATION AT SCALE

Two modes of migration, continuous sync and one time migration

Create a new migration project in DMS

Define your source database

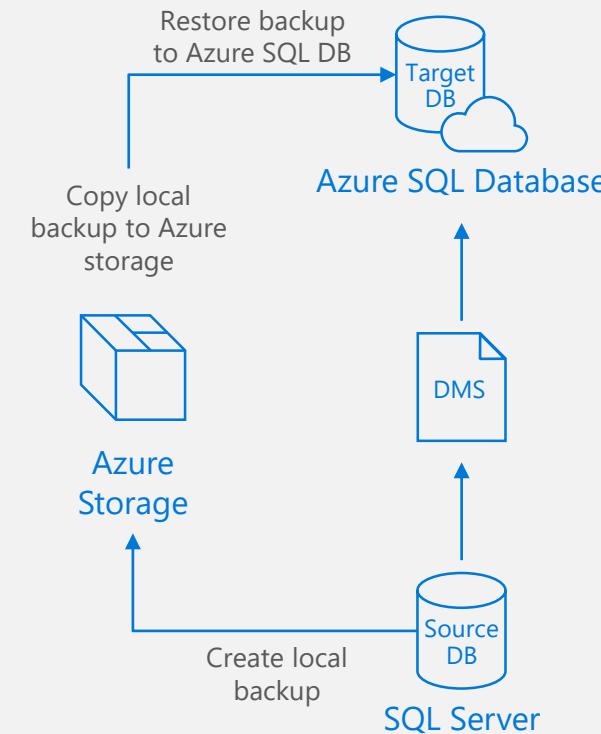
Define your target database

Deploy DMS in either continuous or one time mode

Database is backed up locally

Local backup is copied to Azure Storage

Backup is restored, including all the database objects, to the target Azure SQL DB

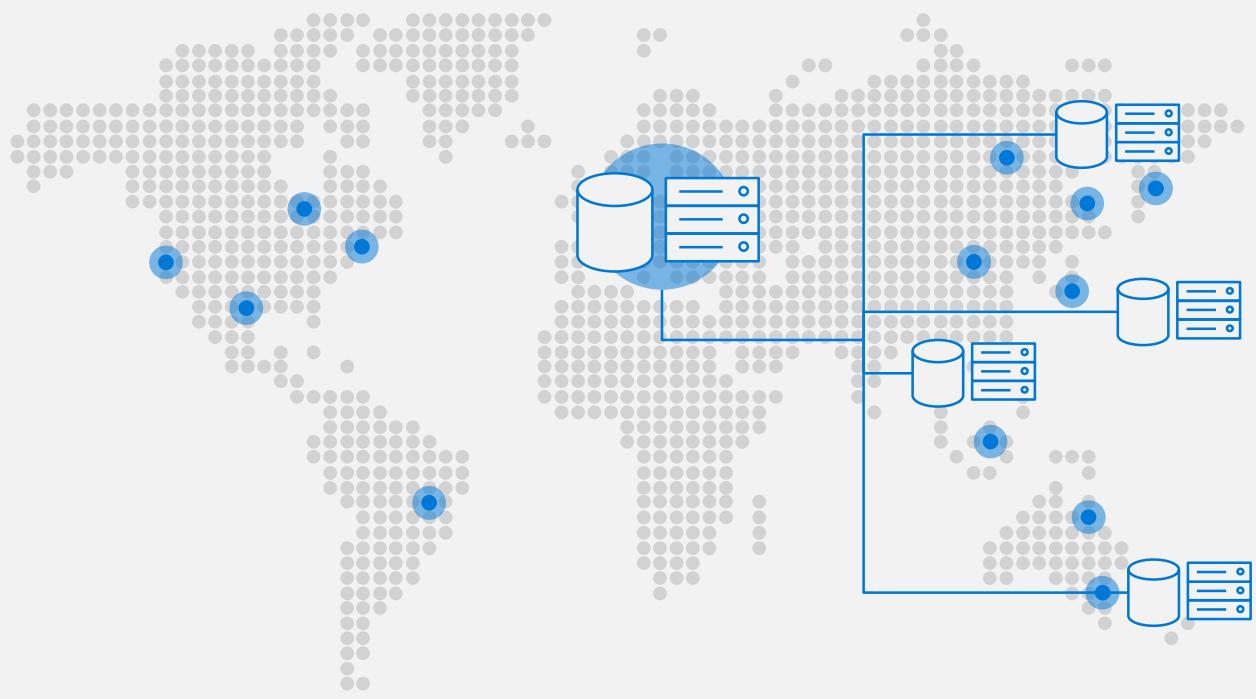


Migrate an entire database and the database objects using Database Migration Service

ACTIVE GEO-REPLICATION

Mission-critical business continuity on your terms, via programmatic APIs

Service levels	Basic, standard, premium Self service
Readable secondaries	Up to 4
Regions available	Any Azure region
Replication	Automatic, asynchronous
Manageability tools	REST API, PowerShell, or Azure Portal
Recovery time objective (RTO)	<1 hour
Recovery point objective	<5 minutes
Failover	On demand



Up to 4 secondaries

Query non-relational data with Polybase

Capability

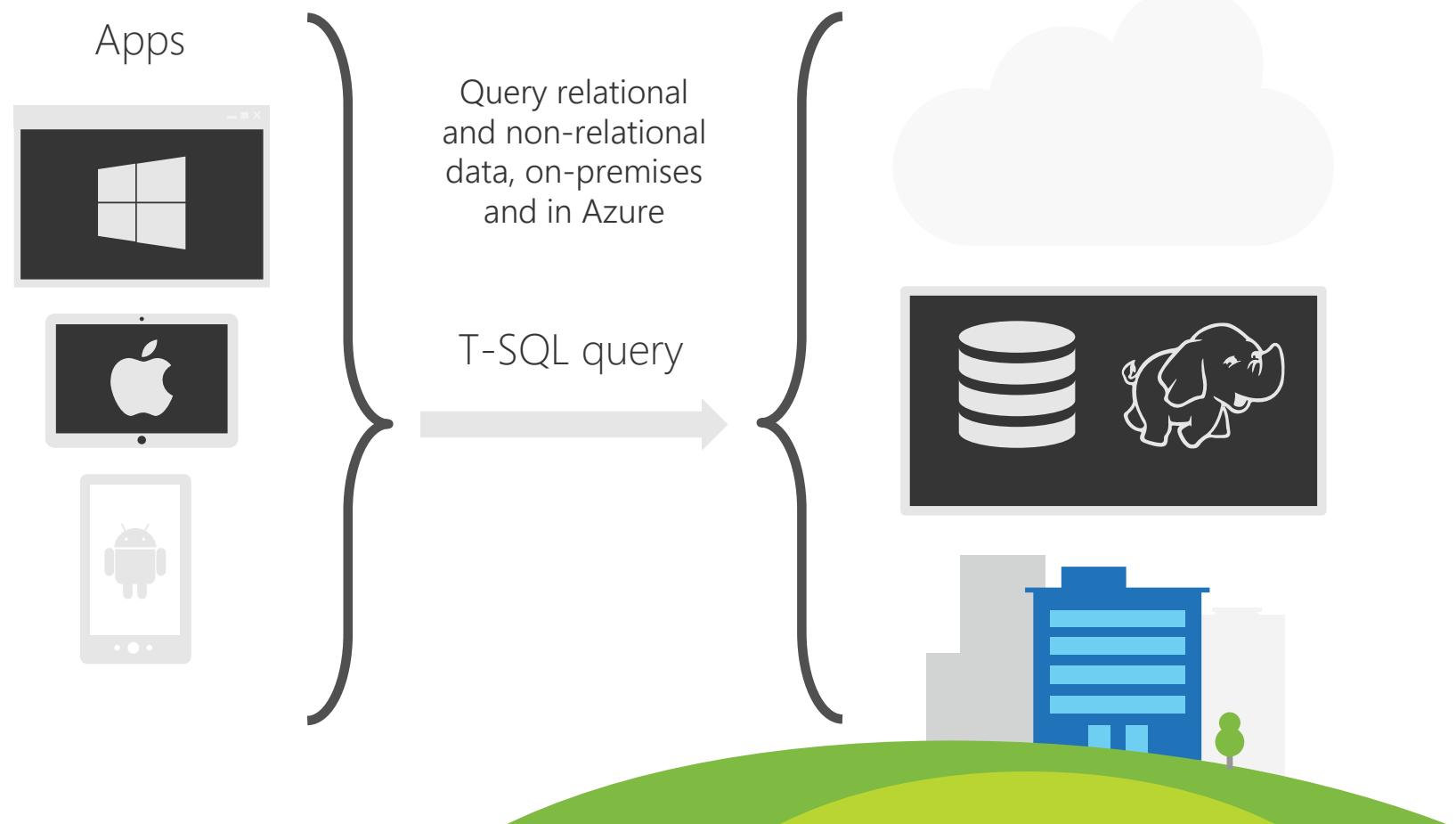
T-SQL for querying relational
and non-relational data across SQL
Server and Hadoop

Benefits

New business insights across your data
lake

Leverage existing skill sets
and BI tools

Faster time to insights and simplified
ETL process



PolyBase use cases

Load data

Use Hadoop as an ETL tool to cleanse data before loading to data warehouse with PolyBase

Interactively query

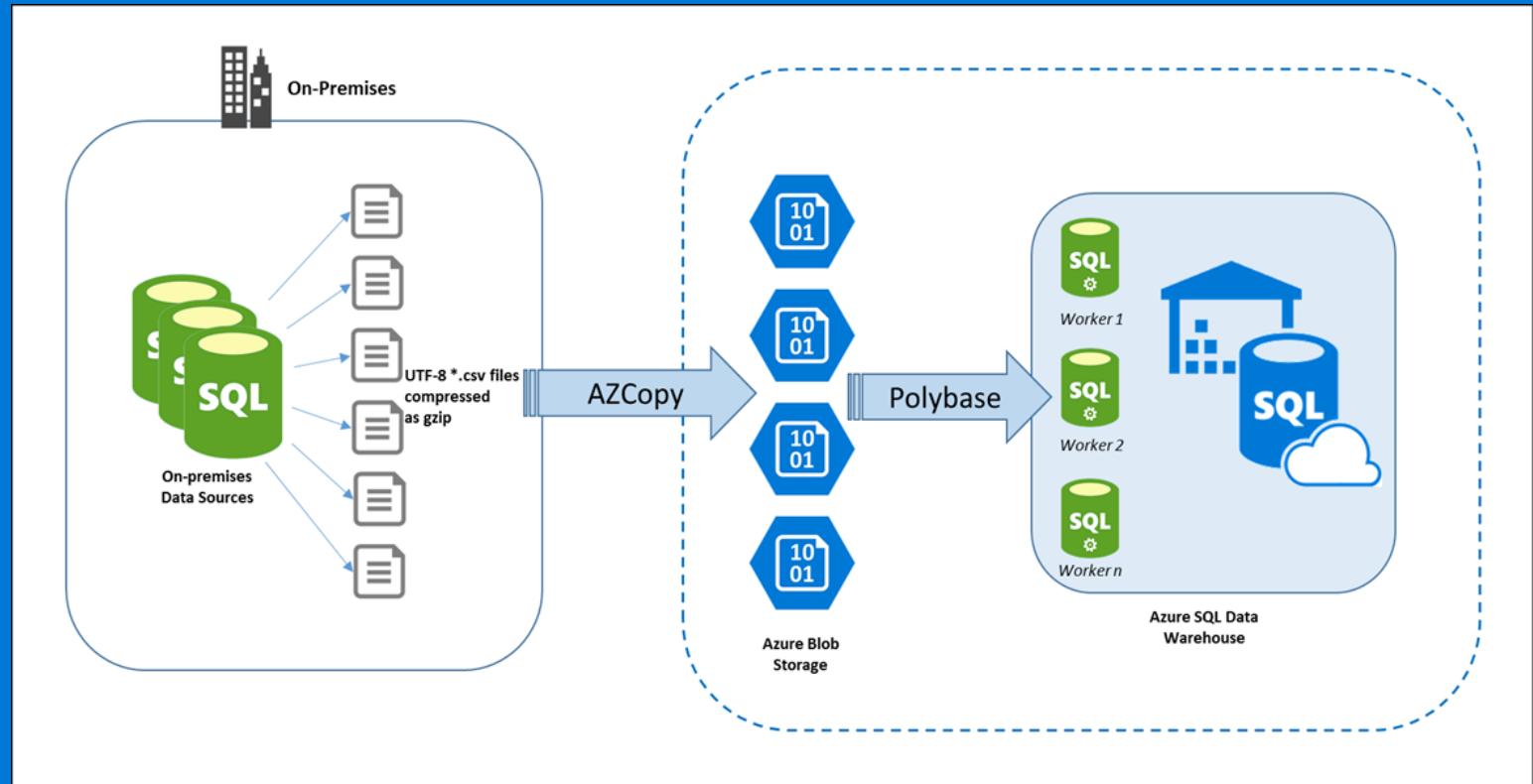
Analyze relational data with semi-structured data using split-based query processing

Age out data

Age out data to HDFS and use it as “cold” but queryable storage

Polybase & AZcopy

- Export data to csv files in UTF-8
- Transfer files to blob storage (e.g., using AZCopy)
- Load data using Polybase (external tables pointing to blob storage)



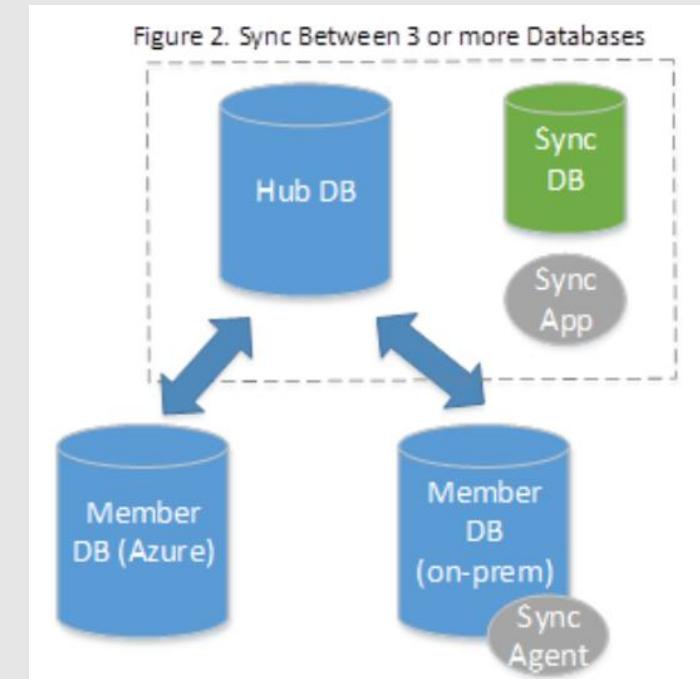
Azure SQL Data Sync

Synchronize data bi-directionally across multiple SQL databases and SQL Server instances.

Uses Hub & Scope architecture:

- The Hub Database must be an Azure SQL Database.
- The member databases can be either SQL Databases, on-premises SQL Server databases, or SQL Server instances on Azure virtual machines.
- The Sync Database contains the metadata and log for Data Sync.

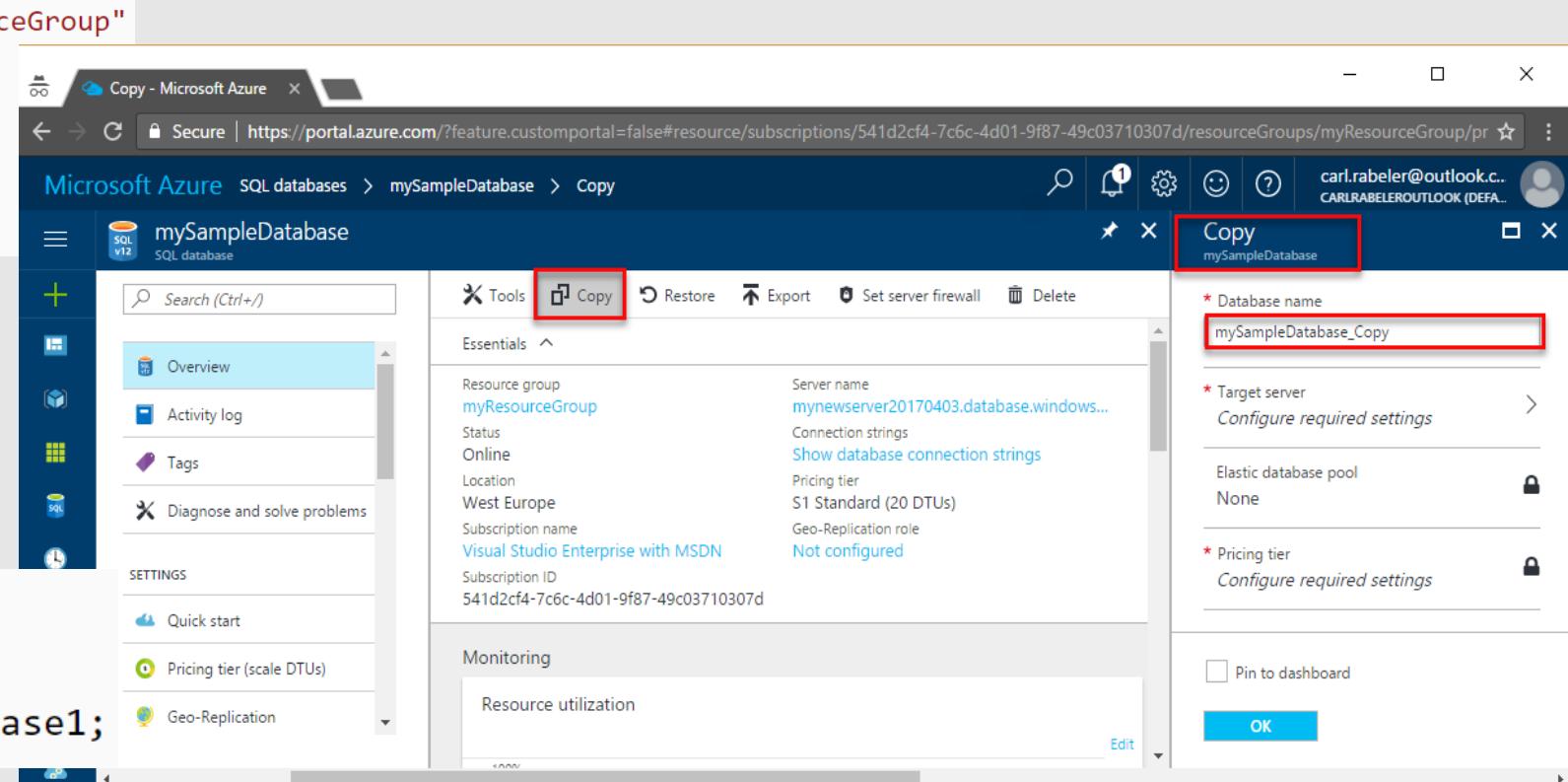
Use Cases: Hybrid Data Sync & Distributed Applications



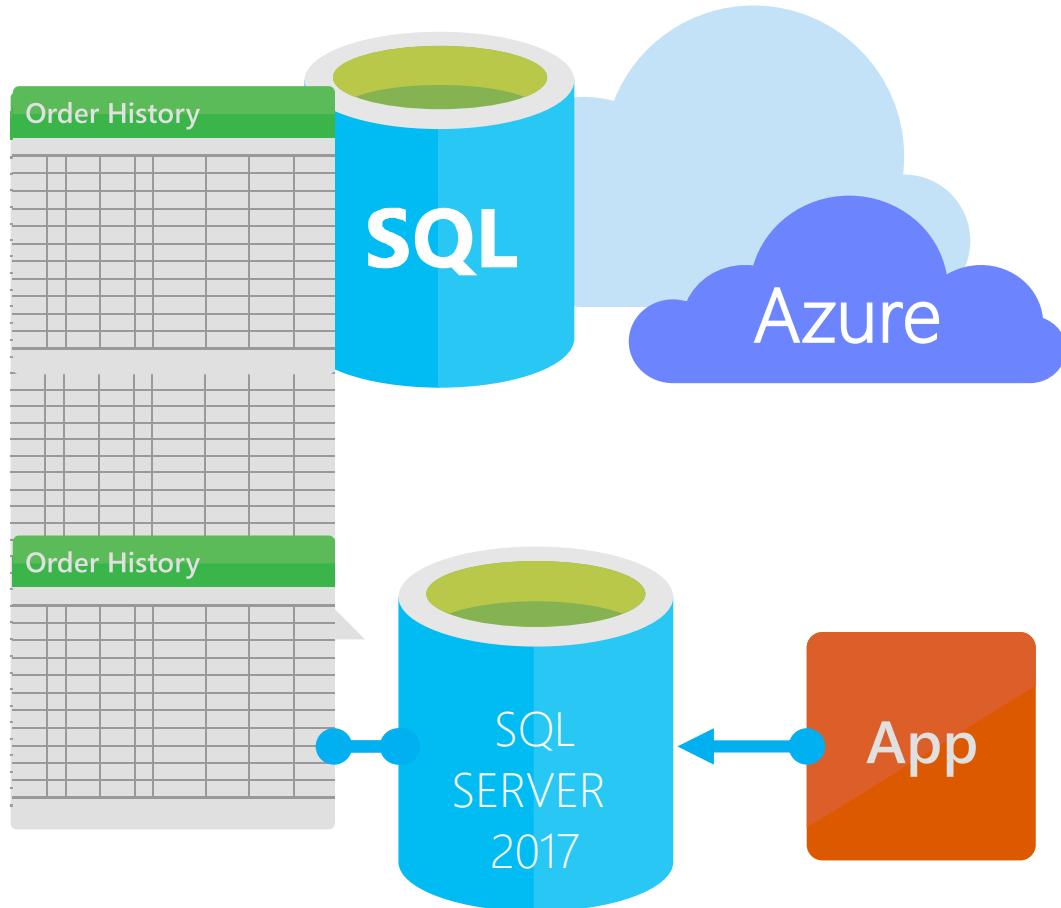
Database Snapshot – Azure SQL DB

- Copy a transactionally consistent copy of an Azure SQL database
- Multiple options: T-SQL, PowerShell, Azure Portal

```
New-AzureRmSqlDatabaseCopy -ResourceGroupName "myResourceGroup"  
-ServerName $sourceserver  
-DatabaseName "MySampleDatabase"  
-CopyResourceGroupName "myResourceGroup"  
-CopyServerName $targetserver  
-CopyDatabaseName "CopyOfMySampleDatabase"  
  
-- Execute on the master database.  
-- Start copying.  
CREATE DATABASE Database2 AS COPY OF Database1;
```



Stretch SQL Server into Azure



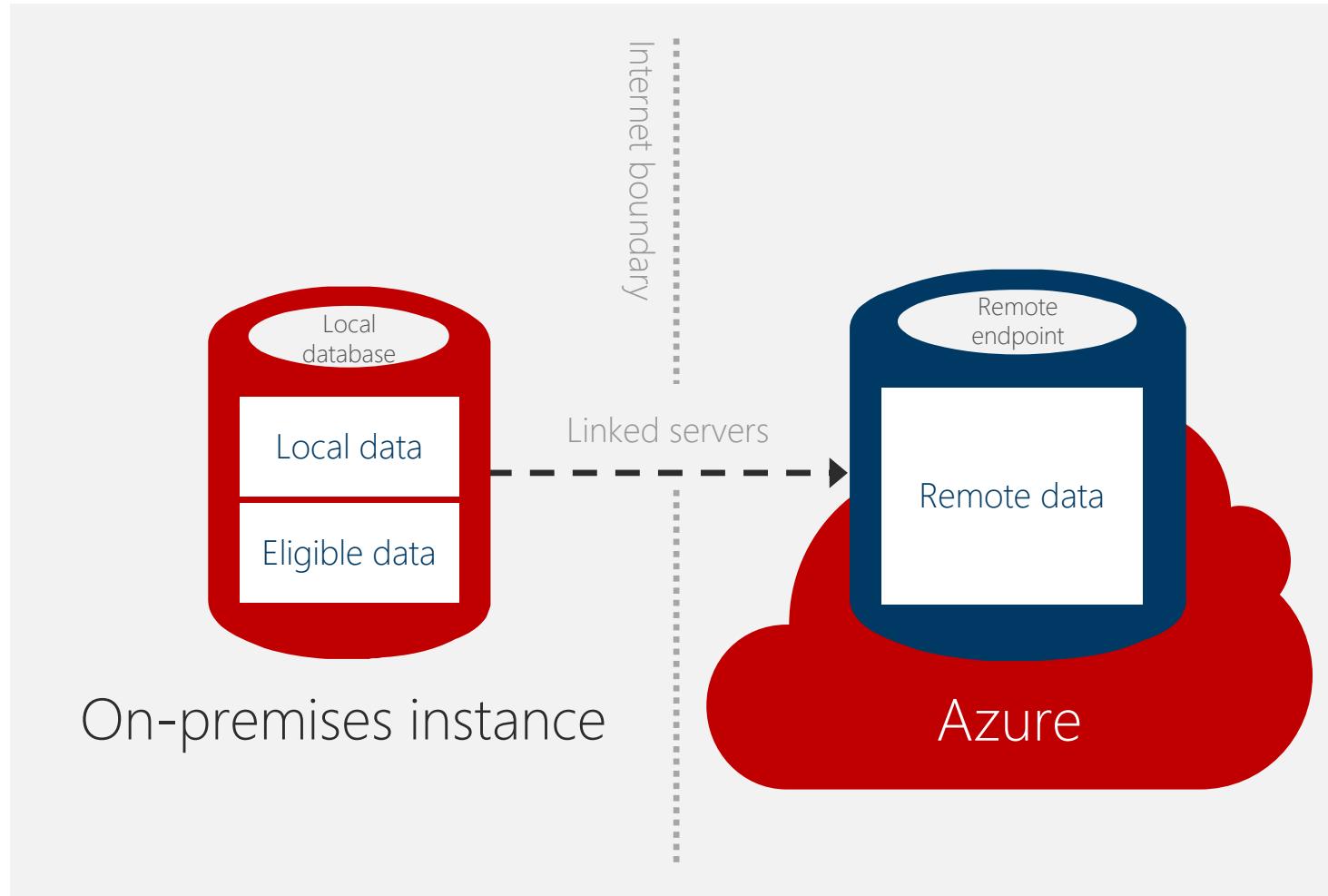
Capability

- Stretch large operational tables from on-premises to Azure with the ability to query

Benefits

- Cost-effective online cold data
- Entire table is online and remains queryable from on-premises apps
- No application changes
- Support for Always Encrypted and Row-Level Security
- Stretching the history tables of temporal tables is a great scenario for Stretch database to allow time-travel for a longer period of time.

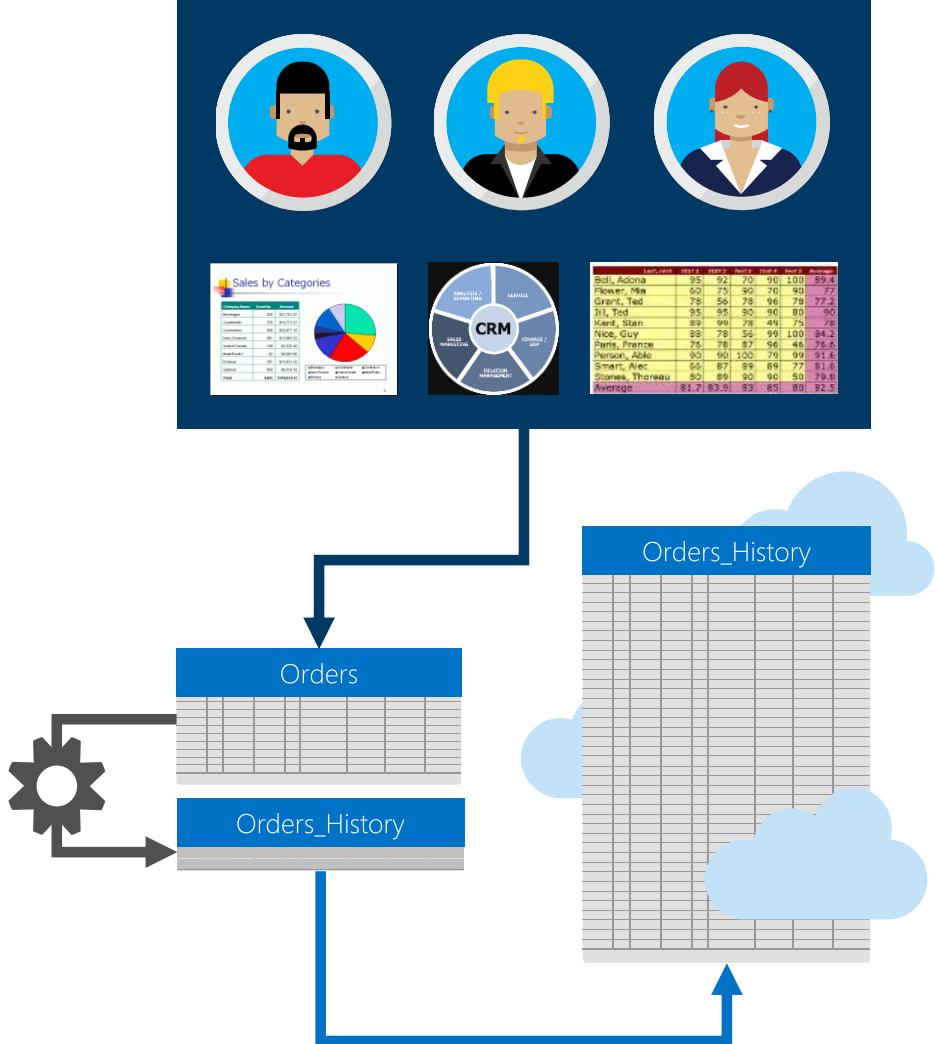
Stretch Database architecture



How it works

- Creates a secure linked server definition in the on-premises SQL Server
- Targets remote endpoint with linked server definition
- Provisions remote resources and begins to migrate eligible data, if migration is enabled
- Queries against tables run against both local database and remote endpoint

Queries continue working



- Business applications continue working without disruption
- DBA scripts and tools work as before (all controls still held in local SQL Server)
- Developers continue building or enhancing applications with existing tools and methods

Practice Question 2

You have an Azure SQL Database named DB1. You have an Azure App Service web application named WebApp1.

WebApp1 is not a mission-critical application. A prolonged outage will not result in massive data loss.

You are designing a disaster recovery solution for WebApp1. You have the following requirements:

- * Meet an Estimated Recovery Time (ERT) of 12 hours.
- * Provide a Recovery Point Objective (RPO) of 1 hour.
- * Restore is only needed during prolonged regional outages.
- * Cannot require additional development or setup.

Your solution must minimize additional or unnecessary costs.

You need to choose the recovery option for WebApp1.

Which recovery option should you use?

Practice Question 2 (contd.)

Which recovery option should you use?

- A. Transactional replication
- B. Geo-restore
- C. Standard Geo-Replication
- D. Active Geo-Replication

*Courtesy of MeasureUp

Practice Question 2 (contd.)

Which recovery option should you use?

- A. Transactional replication
- B. Geo-restore
- C. Standard Geo-Replication
- D. Active Geo-Replication

*Courtesy of MeasureUp

<https://azure.microsoft.com/en-us/blog/azure-sql-database-geo-restore/>

Lab 03: Migrating a database to Azure SQL DB

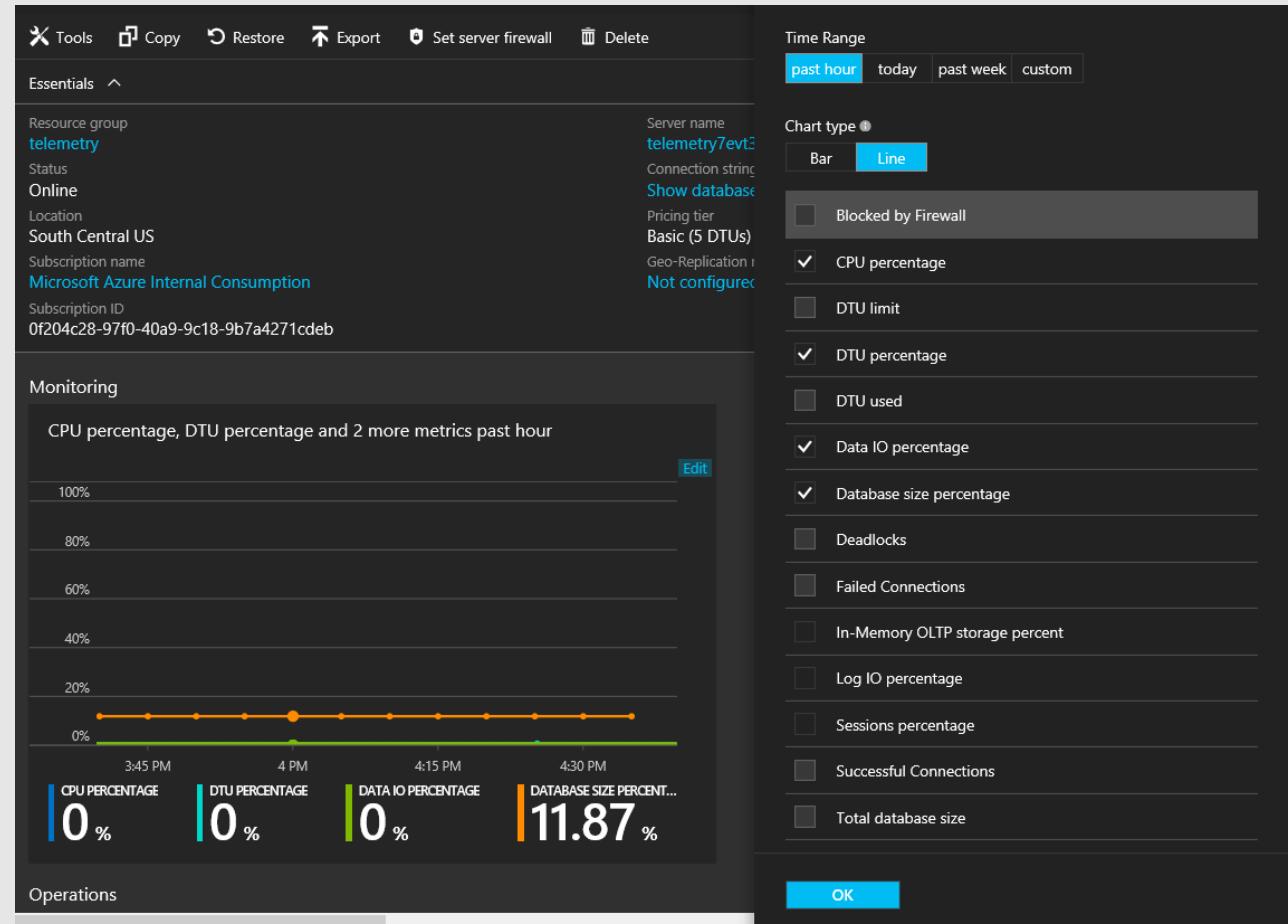
4. Monitor and manage database implementations
in Azure (25–30%)

Objectives

- Monitor and troubleshoot SQL Server VMs on Azure
 - Monitor database and instance activity
 - Monitor by using DMVs and DMFs
 - Monitor performance and scalability
- Monitor and troubleshoot SQL Database
 - Monitor database activity
 - Monitor using DMVs and DMFs
 - Monitor performance and scalability
- Automate and manage database implementations in Azure
 - Automate and manage SQL Server in Azure VMs
 - Automate and manage Azure SQL Database
 - Configure automation and runbooks

SQL Database resource monitoring

- master.sys.resource_stats
 - Based on 5 minute averages
 - Maintained for approximately 35 days
 - CPU usage, storage size change, and database SKU modification per database
- userdb.sys.dm_db_resource_stats
 - Based on 15 second averages
 - Maintained for one hour
- Accessible through Azure Portal
 - Enables configuration of alerts



Database Auditing

- Tracks database events and writes them to an audit log in your Azure storage account, OMS workspace, Event Hubs or Windows Security Log/event logs/files for SQL on IaaS
- Understand database activity, and gain insight into discrepancies and anomalies that could indicate business concerns or suspected security violations
- Server-level vs. database-level auditing policy:
 - Server policy applies to all existing and newly created databases on the server
 - Individual options are grouped into "action groups."

Save Discard View audit logs Feedback

Action group name	Description
LOGIN_CHANGE_PASSWORD_GROUP	This event is raised whenever a login password is changed by way of ALTER LOGIN statement or sp_password stored procedure. Equivalent to the Audit Login Change Password Event Class .
BACKUP_RESTORE_GROUP	This event is raised whenever a backup or restore command is issued. Equivalent to the Audit Backup and Restore Event Class .
DATABASE_CHANGE_GROUP	This event is raised when a database is created, altered, or dropped. This event is raised whenever any database is created, altered or dropped. Equivalent to the Audit Database Management Event Class .

If Blob Auditing is enabled on the server, it will affect database settings.

[View server settings](#)

Server-level Auditing: Disabled

Auditing

ON OFF

Audit log destination (choose at least one):

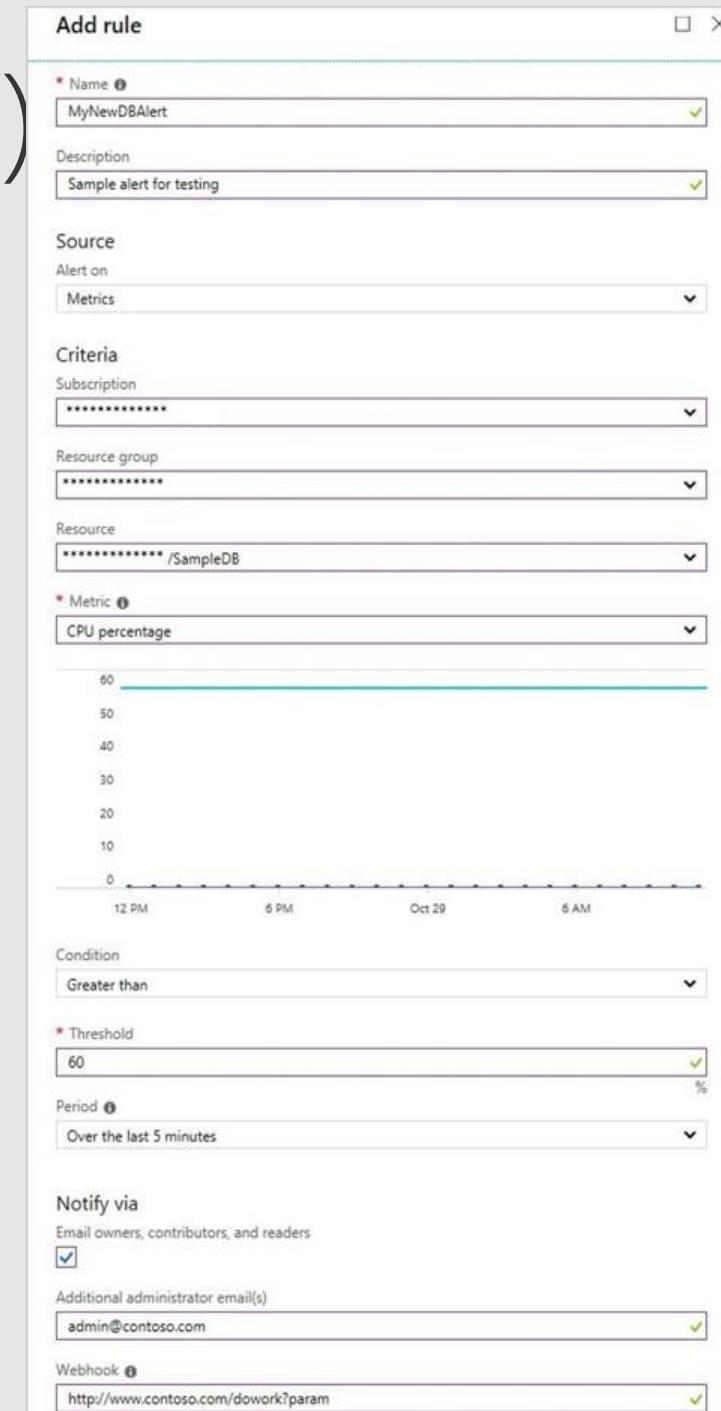
Azure Automation Runbook

- Using Azure Automation Runbooks, long-running, manual, error-prone, and frequently repeated tasks can be automated
- Azure SQL Database can be managed in Azure Automation by using the Azure SQL Database PowerShell cmdlets
- Common Tasks:
 - Scale Azure SQL Database
 - <https://gallery.technet.microsoft.com/scriptcenter/Azure-SQL-Database-e957354f>
 - Run SQL Queries/Jobs
 - <https://gallery.technet.microsoft.com/scriptcenter/How-to-use-a-SQL-Command-be77f9d2>

Azure Automation Runbook(contd.)

- Auto-scale Azure SQL DB
 - Start Azure Automation Runbook with Webhook
 - A *webhook* allows you to start a particular runbook in Azure Automation through a single HTTP request

```
# Establish credentials for Azure SQL Database server
$Servercredential = new-  
object System.Management.Automation.PSCredential($Using:Credential.UserName, ($Using:Credential).GetNe  
tworkCredential().Password | ConvertTo-SecureString -asPlainText -Force)  
  
# Create connection context for Azure SQL Database server
$CTX = New-AzureSqlDatabaseServerContext -  
ManageUrl "https://$Using:SqlServerName.database.windows.net" -Credential $ServerCredential  
  
# Get Azure SQL Database context
$Db = Get-AzureSqlDatabase $CTX -DatabaseName $Using:DatabaseName  
  
# Specify the specific performance level for the target $DatabaseName
$ServiceObjective = Get-AzureSqlDatabaseServiceObjective $CTX -  
ServiceObjectiveName "$Using:PerfLevel"  
  
# Set the new edition/performance level
Set-AzureSqlDatabase $CTX -Database $Db -ServiceObjective $ServiceObjective -  
Edition $Using:Edition -Force  
  
# Output final status message
Write-  
Output "Scaled the performance level of $Using:DatabaseName to $Using:Edition - $Using:PerfLevel"
Write-Output "Completed vertical scale"
}
```



Protecting deletions of Azure SQL Resources

- Azure Resource Management Locks feature prevents accidental deletion of resources such as a SQL Server
- E.g. typo in an API call or accidentally deleting the server that was in a different region.

The screenshot shows the Azure portal interface for managing locks on a SQL server named "myservereast1".

Left Sidebar:

- Search bar: "Search (Ctrl+ /)"
- Navigation menu:
 - Overview
 - Activity log
 - Access control (IAM)
 - Tags
 - Diagnose and solve problems
- SETTINGS section:
 - Quick start
 - Firewall
 - Long-term backup retention
 - Auditing & Threat Detection
 - Active Directory admin
 - Deleted databases
- Properties section:
 - Locks (selected)
 - Automation script
- SUPPORT + TROUBLESHOOTING section:
 - Automatic tuning
 - New support request

Locks Blade:

LOCK NAME	LOCK TYPE	SCOPE	NOTES
This resource has no locks.			

Add lock Dialog:

Add lock

Lock name	Lock type
PreventDelete	Delete

Notes: This server should never be deleted!

OK Cancel

DMVs

- There are two types of dynamic management views and functions:
 - Server-scoped dynamic management views and functions. These require **VIEW SERVER STATE** permission on the server.
 - Database-scoped dynamic management views and functions. These require **VIEW DATABASE STATE** permission on the database

Important DMVs

- **sys.dm_os_wait_stats**: Returns information about all the waits encountered by threads that executed. You can use this aggregated view to diagnose performance issues with SQL Server and also with specific queries and batches
- **sys.dm_db_resources_stats**: Returns CPU, I/O, and memory consumption for an Azure SQL Database database. Historical data for last 1 hour. When a database is a member of an elastic pool, resource statistics presented as percent values, are expressed as the percent of the max limit for the databases as set in the elastic pool configuration.
- **sys.dm_os_performance_counters**: Returns a row per performance counter maintained by the server. For example, the SQL Server Locks object contains counters called Number of Deadlocks/sec and Lock Timeouts/sec.

<https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/system-dynamic-management-views?view=sql-server-2017>

SQL Database monitoring/troubleshooting

- Query tuning
 - Query Performance Insights
 - <https://azure.microsoft.com/en-gb/documentation/articles/sql-database-query-performance/>
 - Deeper insight into your databases resource (DTU) consumption.
 - The top queries by CPU/Duration/Execution count, which can potentially be tuned for improved performance.
 - The ability to drill down into the details of a query, view its text and history of resource utilization.
- Intelligent Insights:
 - <https://docs.microsoft.com/en-gb/azure/sql-database/sql-database-intelligent-insights>
 - using AI to monitor and troubleshoot database performance

Practice Question 3

You are the database administrator for an Azure SQL Database for your company. The Azure SQL Database is used by a web app. The IT manager creates a new slot for this web app.

You need to make a copy of the Azure SQL Database based on a snapshot of the current Azure SQL Database for the newly created web app slot.

Which Azure PowerShell cmdlet best accomplishes this task?

- A. Start-AzureSqlDatabaseCopy -ServerName \$SqlServer -DatabaseName \$DBName -PartnerServer \$TargetSqlServer -ContinuousCopy
- B. Start-AzureSqlDatabaseCopy -ServerName \$SqlServer -DatabaseName \$DBName -PartnerServer \$TargetSqlServer -ContinuousCopy -OfflineSecondary
- C. Invoke-AzureResourceAction -ResourceGroupName \$RGName -ResourceType Microsoft.Web/sites/slots -ResourceName \$WebApp -Action slotsswap -Parameters \$ParametersObj -ApiVersion 2015-07-01
- D. New-AzureRmSqlDatabaseCopy -ResourceGroupName \$RGName -ServerName \$SqlName -DatabaseName \$DBName -CopyDatabaseName \$SqlCopyName

Practice Question 3

You are the database administrator for an Azure SQL Database for your company. The Azure SQL Database is used by a web app. The IT manager creates a new slot for this web app.

You need to make a copy of the Azure SQL Database based on a snapshot of the current Azure SQL Database for the newly created web app slot.

Which Azure PowerShell cmdlet best accomplishes this task?

- A. Start-AzureSqlDatabaseCopy -ServerName \$SqlServer -DatabaseName \$DBName -PartnerServer \$TargetSqlServer -ContinuousCopy
- B. Start-AzureSqlDatabaseCopy -ServerName \$SqlServer -DatabaseName \$DBName -PartnerServer \$TargetSqlServer -ContinuousCopy -OfflineSecondary
- C. Invoke-AzureResourceAction -ResourceGroupName \$RGName -ResourceType Microsoft.Web/sites/slots -ResourceName \$WebApp -Action slotsswap -Parameters \$ParametersObj -ApiVersion 2015-07-01
- D. New-AzureRmSqlDatabaseCopy -ResourceGroupName \$RGName -ServerName \$SqlName -DatabaseName \$DBName -CopyDatabaseName \$SqlCopyName

Exam-Taking Tips

Relax. *"You will not be beat with a stick."*

Look for the differences between choices

Keep it simple

Case studies – bottom up or top down?

When you don't know the answer

Eliminate wrong answers

Mark items for review

Guessing is better than nothing

If you've studied – don't second guess yourself (it's not a test designed to trick you)

Time

Review exam topic areas from guide

- Schedule your study
- Schedule your exam
- Know the GUI, know the code
- Take a practice test, and *learn* the gaps and *learn* the environmental variables of the testing center!

Sleep!!!!

More Resources

Online training

[Provisioning Databases in Azure and SQL Server](#)

[Securing Data in Azure and SQL Server](#)

[Recovering Data in Azure and SQL Server](#)

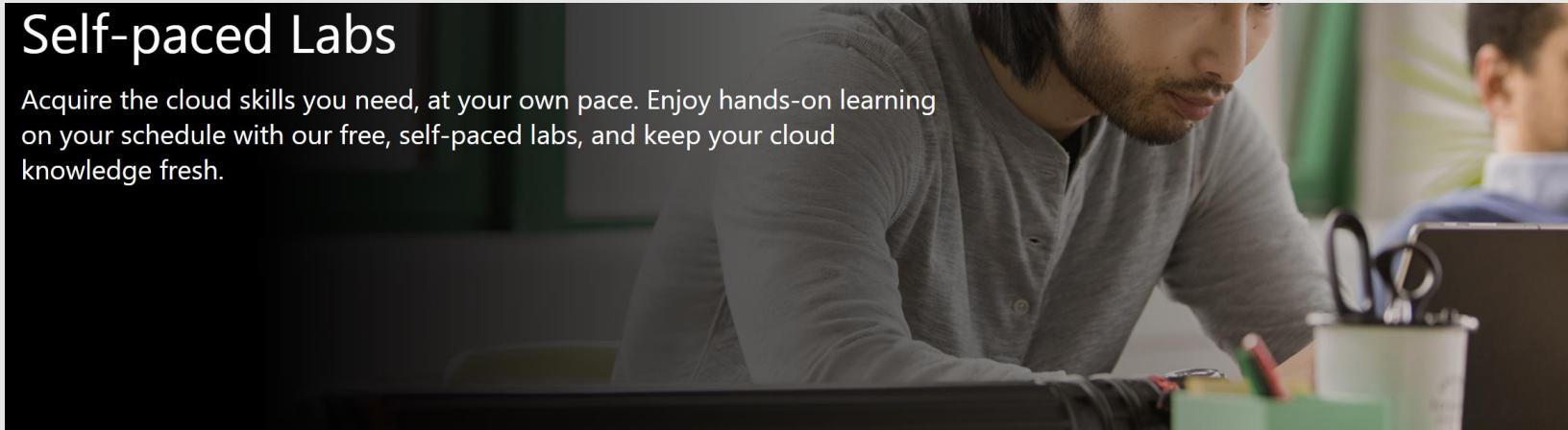
[Managing SQL Server Operations](#)

<https://www.microsoft.com/en-ca/learning/exam-70-473.aspx>

Microsoft.com/handsonlabs

Self-paced Labs

Acquire the cloud skills you need, at your own pace. Enjoy hands-on learning on your schedule with our free, self-paced labs, and keep your cloud knowledge fresh.



< Back to Hands-on Labs

+ Job Roles

+ Audience

48 Self-paced Labs

[Clear Filters](#)

+ Topic

Sort By:

[1](#) [2](#) [3](#)

+ Level

+ Product

SQL Server 2016 - Performance and
Troubleshooting
SQ00115

120 min

The goal of performance and

SQL Server 2016 - PolyBase
SQ00116

120 min

One of the exciting new features in