Microsoft

AZ-305

Designing Microsoft Azure Infrastructure Solutions

AZ-305 Agenda

Module 01 Design a governance solution

Module 02 Design a compute solution

Module 03 Design a non-relational data storage solution

Module 04 Design a data storage solution for relational data

Module 05 Design a data integration solution

Module 06 Design an application architecture solution

Module 07 Design Authentication and Authorization Solutions

Module 08 Design a solution to log and monitor Azure resources

Module 09 Design a network infrastructure solution

Module 10 Design a business continuity solution

Module 11 Design a migration solution

Design a data storage solution for relational data

Learning Objectives

- Design for data storage
- Design for Azure SQL databases
- Recommend a solution for database scalability
- Recommend a solution for database availability
- Design security for data at rest, data in transmission, and data in use
- Design for Azure SQL Edge
- Design for Azure Cosmos DB and tables
- Case study
- Learning recap

AZ-305: Design Data Storage Solutions (20-25%):

Design data storage solution for relational data

- Recommend a solution for storing relational data
- Recommend a database service tier and compute tier
- Recommend a solution for database scalability
- Recommend a solution for data protection

AZ-305: Design Business Continuity Solutions (15-20%):

Design for High Availability

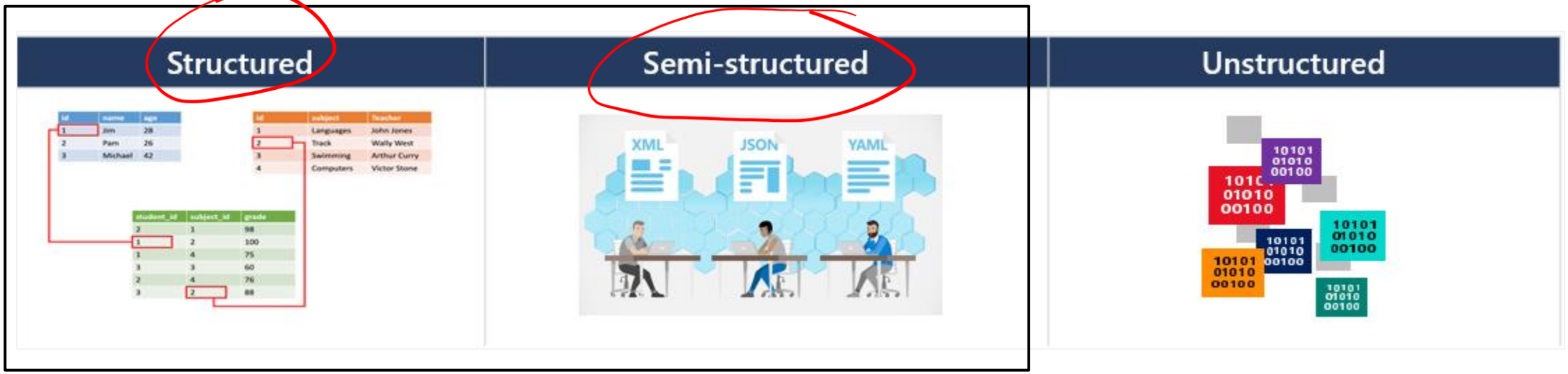
- Recommend a high availability solution for relational data

Design for data storage

Cosmos DB

Design for structured and semi-structured data

SQL → No SQL



To design Azure storage, you first must determine what type of data you have.

- **Structured data** includes relational data and has a shared schema
- **Semi-structured** is less organized than structured data and isn't stored in a relational format
- **Unstructured data** is the least organized type of data

© Copyright Microsoft Corporation. All rights reserved.

Design for Azure SQL databases

When to use Azure SQL databases

SQL virtual machines

Best for migrations and applications requiring OS-level access

SQL virtual machine

- SQL Server and OS server access
- Expansive SQL and OS version support
- Automated manageability features

Managed instances

Best for most lift-and-shift migrations to the cloud

Single instance

- SQL Server surface area (vast majority)
- Native virtual network support
- Fully managed service

Instance pool

- Resource sharing between multiple instances to price optimize
- Simplified performance management for multiple databases
- Fully managed service

Databases

Best for modern cloud applications

Single database

- Hyperscale storage (up to 100TB)
- Serverless compute
- Fully managed service

Elastic pool

- Resource sharing between multiple databases to price optimize
- Simplified performance management for multiple databases
- Fully managed service

© Copyright Microsoft Corporation. All rights reserved.

Recommend a solution for database scalability

Database scaling strategy

The following table identifies scenarios that require different scaling solutions

Requirement	Solution
Do you have to manage and scale multiple Azure SQL databases that have varying and predictable resource requirements?	SQL elastic pools.
Are you developing a new application with a single database that you want to test before launching it to thousands of users?	Azure SQL Database or SQL Managed Instance
Do you need to optimize the price performance for a group of databases within a prescribed budget while delivering performance elasticity for each database?	SQL elastic pools.

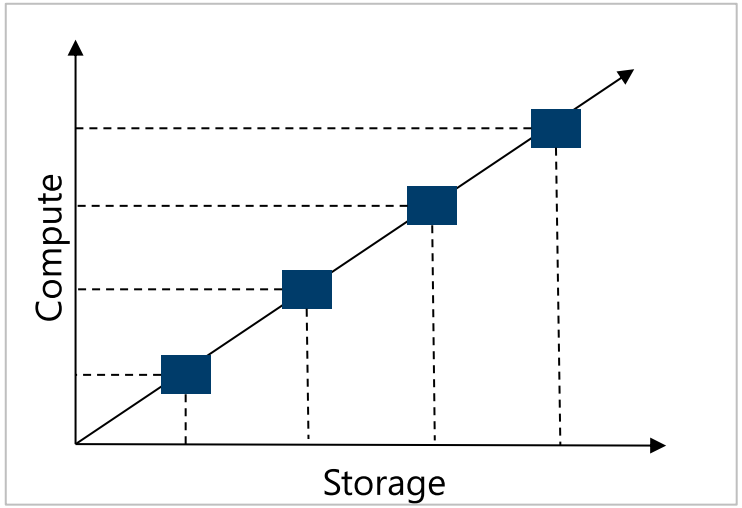
Consider cost together with your scaling strategy to find an optimal solution

© Copyright Microsoft Corporation. All rights reserved.

Recommend a solution for database availability

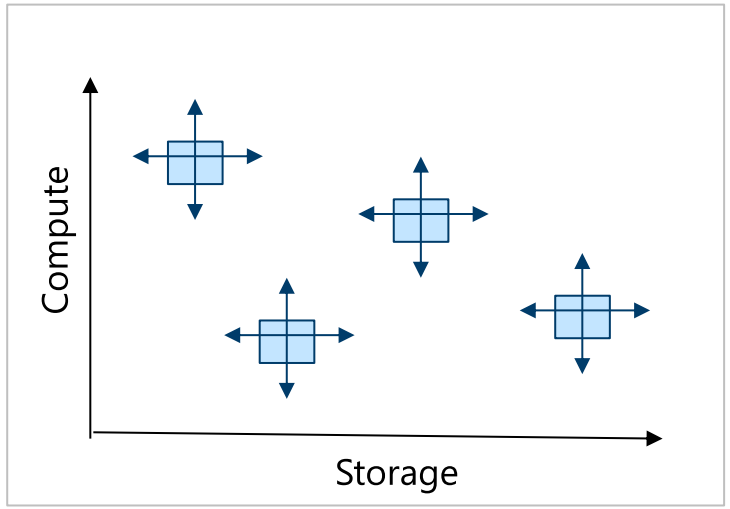
Select an Azure SQL Database pricing model

DTU



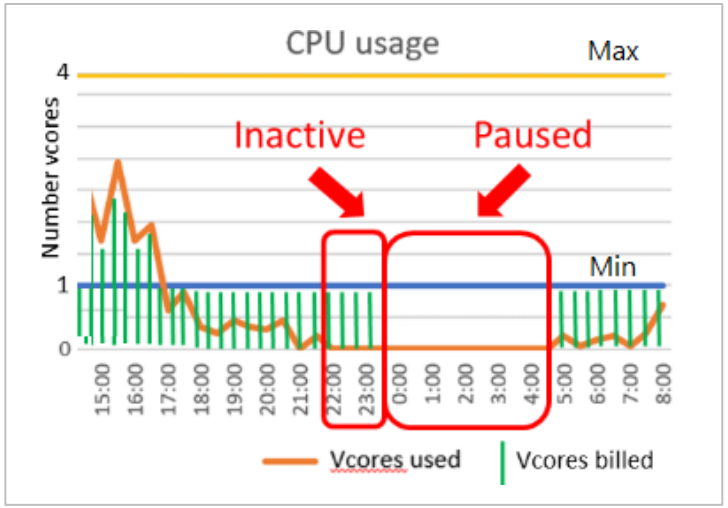
- A simple, preconfigured purchase option.
- A blended measure of CPU, memory, reads, and writes.

vCore



- Flexibility, control and transparency
- Independent scaling of compute, storage, and I/O resources

Serverless



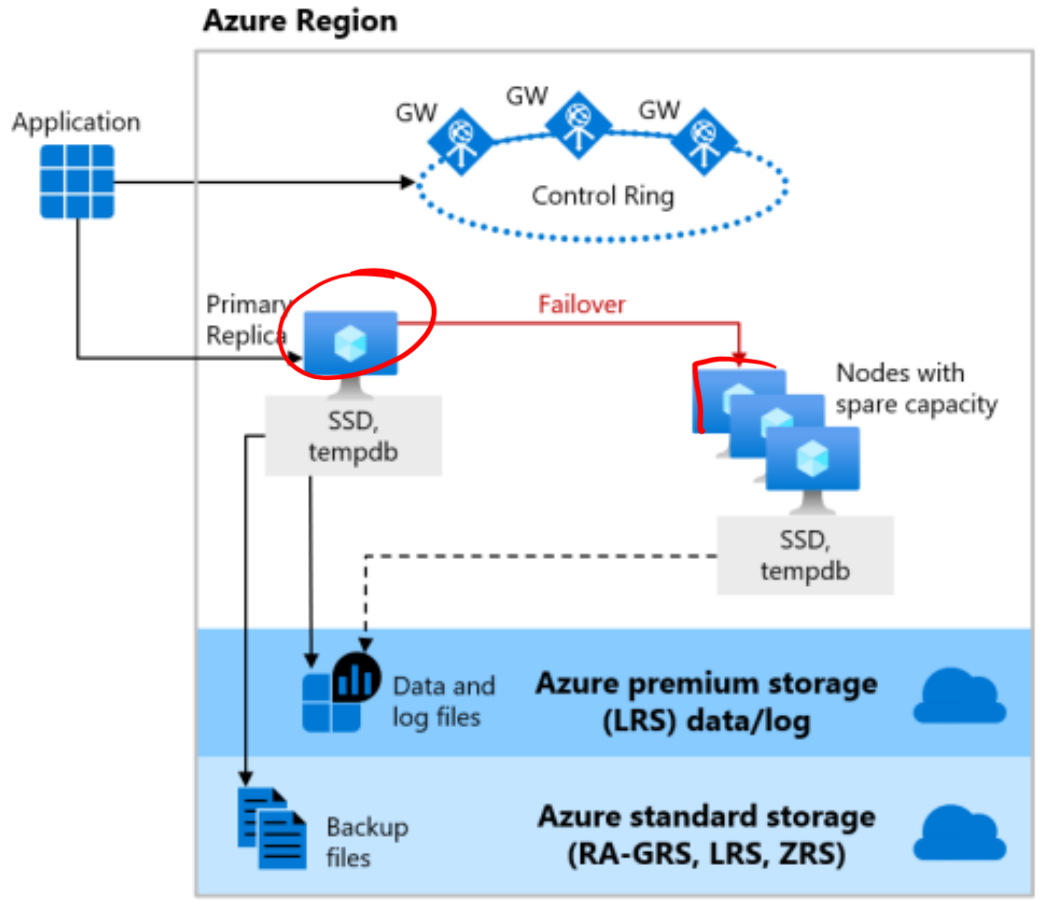
- Intermittent, unpredictable usage
- Automatically scales compute, based on workload demand

© Copyright Microsoft Corporation. All rights reserved.

High availability with the General Purpose/Standard tier

Azure SQL Database offers three service tiers that are designed for different types of applications:

- Designed for common workloads
- Budget oriented balanced compute and storage
- Uses nodes with spare capacity to spin up a new SQL Server instances
- Uses LRS and RA-GRS (backup files)

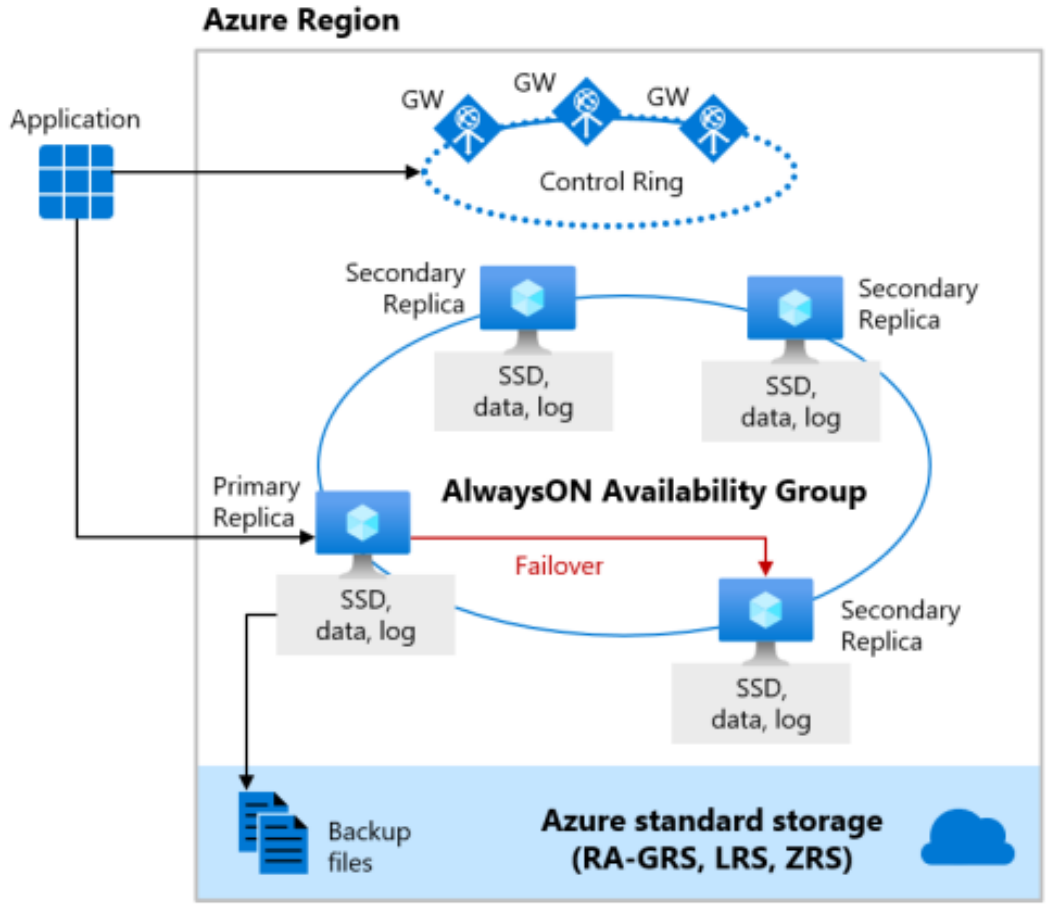


© Copyright Microsoft Corporation. All rights reserved.

High availability with the Business Critical/Premium tier

Azure SQL Database offers three service tiers designed for different types of applications:

- Designed for OLTP applications
- High transaction rate and low I/O latency
- Offers the highest resilience to failures by using several isolated replicas
- Deploys an Always On availability group using multiple synchronously updated replicas
- Uses local SSD storage and RA-GRS (backup files)

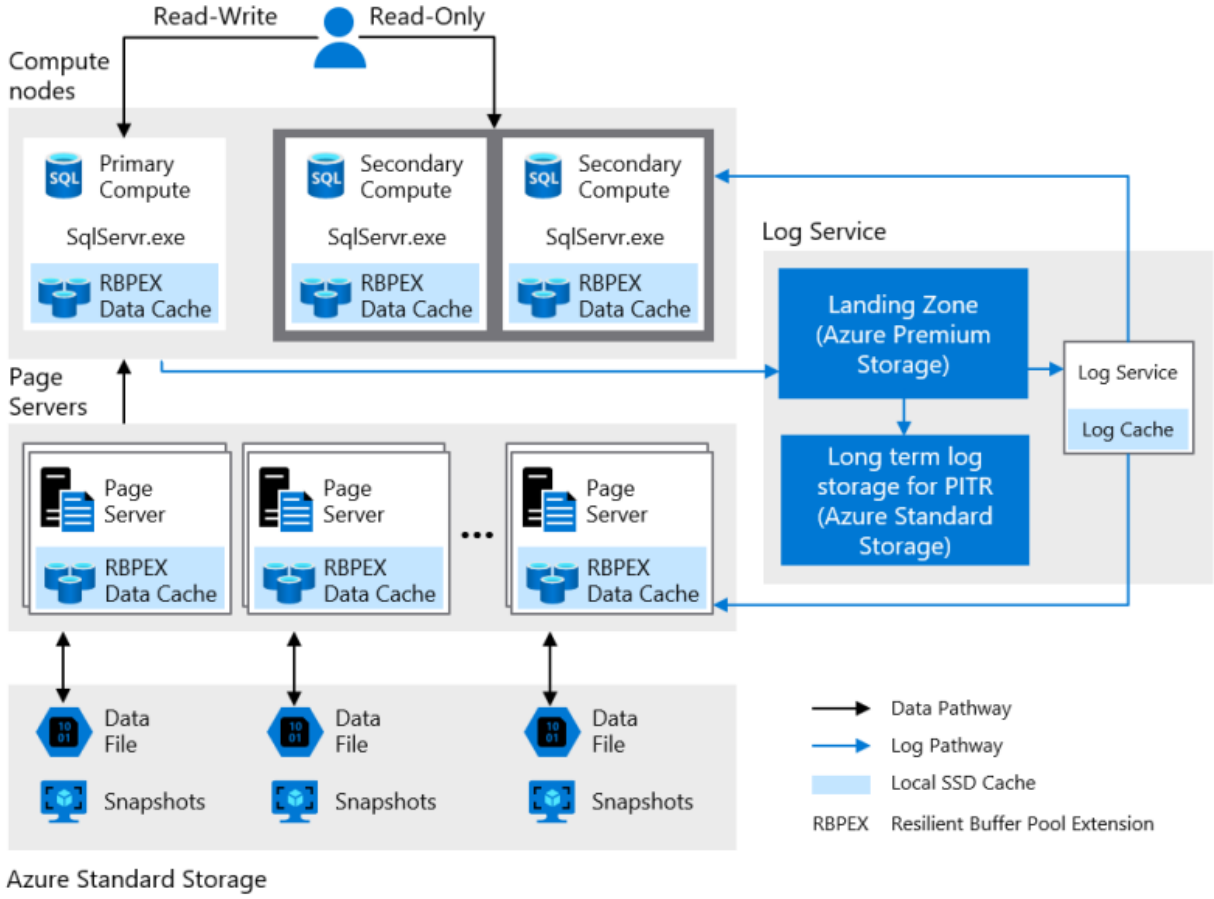


© Copyright Microsoft Corporation. All rights reserved.

High availability with the Hyperscale tier

Azure SQL Database offers three service tiers that are designed for different types of applications:

- Designed for very large OLTP databases – as large as 100 TB
- Able to autoscale storage and scale compute
- Captures instantaneous backups (using snapshots)
- Restores in minutes rather than hours and days
- Scale up or down in real time to accommodate workload changes

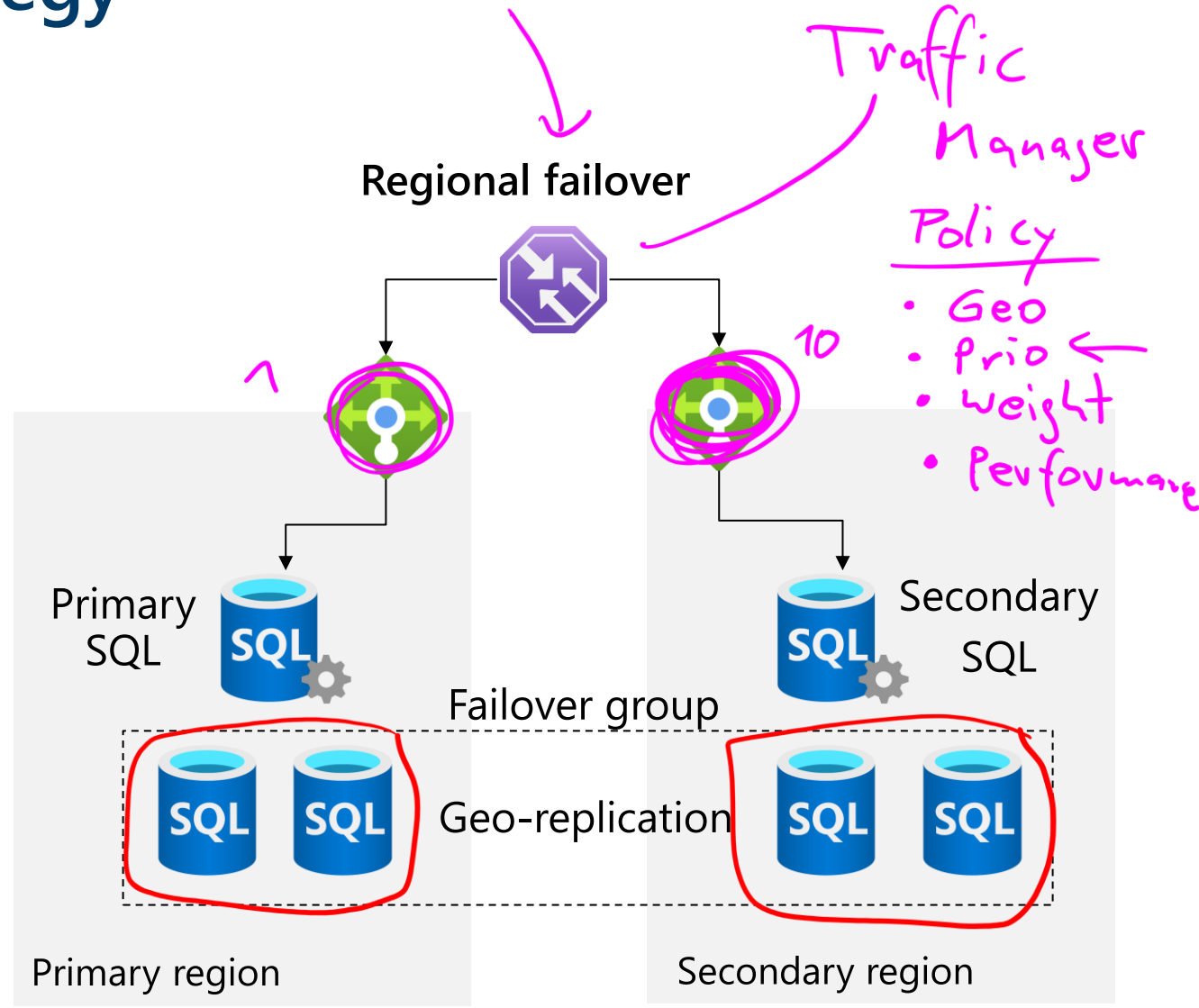


© Copyright Microsoft Corporation. All rights reserved.

Select a database failover strategy

Consider datacenter and regional failover.

- In the same region -use AlwaysOn availability group with failover to secondary replicas
- Across regions – use geo-replication and failover groups



© Copyright Microsoft Corporation. All rights reserved.

Select a database strategy (activity)

- Highly available Azure SQL database that is over 16 TB
- On-premises SQL migration to Azure
- Known database usage at minimal cost
- Replicates across regions
- A cloud platform that tracks inventory for multiple car dealerships.

Auto-failover group	Hyperscale
Business critical	Managed instances
SQL elastic pools	SQL Server on a VM
SQL Database	vCore pricing
Always On	DTU pricing

© Copyright Microsoft Corporation. All rights reserved.

Design security for data at rest, data in transit, and data in use

Protect your database

Use a layered (defense in depth) approach to data protection.

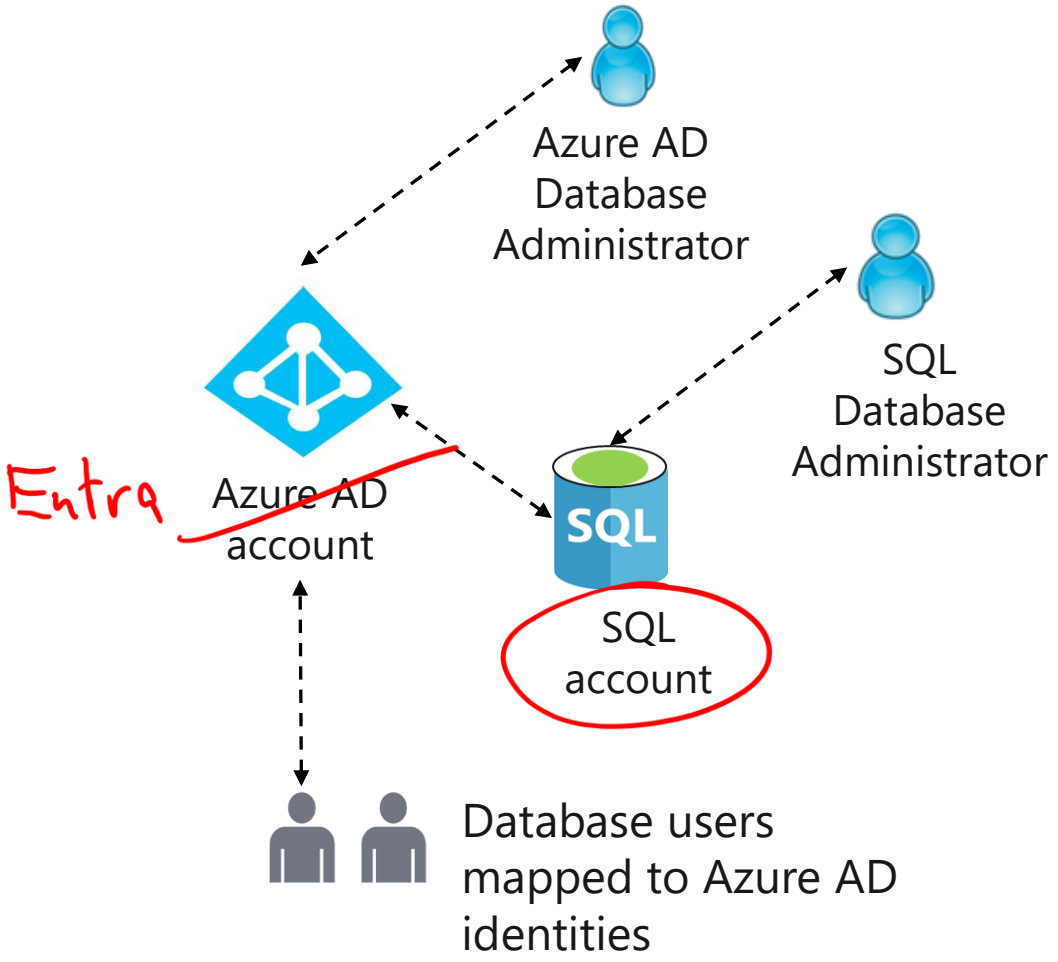
Network security	Identity and access	Data protection	Security management
• VNet • Firewall rules, NSG • Private link	• Authentication options: Azure AD, SQL Auth, Windows Auth • Azure RBAC • Roles and permissions • Row level security	• Encryption-in-use (Always encrypted) • Encryption-at-rest (TDE) • Encryption-in-flight (TLS) • Customer-managed keys • Dynamic data masking	• Advanced threat protection • SQL audit • Audit integration with log analytics and event hubs • Vulnerability assessment • Data discovery and classification • Microsoft Defender for Cloud

ASC

© Copyright Microsoft Corporation. All rights reserved.

Authenticate to an Azure SQL database

- SQL database supports two types of authentication - SQL server authentication and Azure AD authentication
- SQL server authentication credentials are stored in the database
- Azure AD authentication uses managed identities



Select the appropriate features (activity)

Cross Database Query	Linked Servers
OS Level Access	SQL Server Integration Services
Active Geo-Replication	SQL Server Reporting Services
Older SQL Server Version Support	SQL Server Analysis Services
Native VNet Support	Automated Backup

?

- Azure SQL Database
- Azure SQL Managed Instance
- SQL Server on Virtual Machine

© Copyright Microsoft Corporation. All rights reserved.

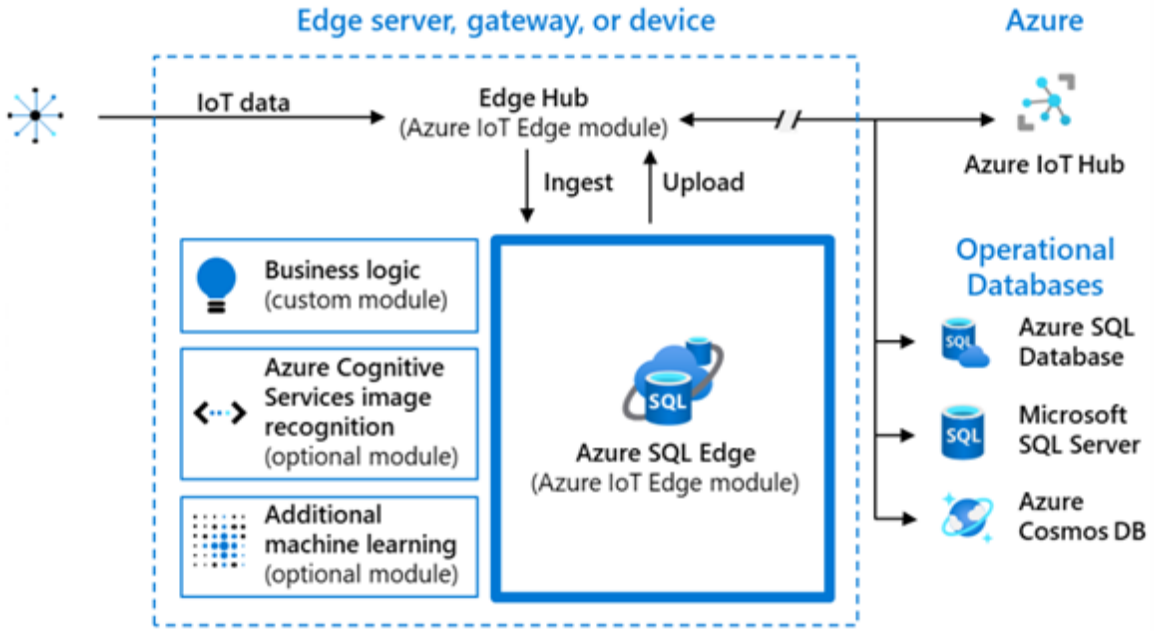
Design for Azure SQL Edge

When to use Azure SQL Edge

An optimized relational database engine geared for IoT and IoT Edge deployments. It is a containerized Linux application that runs on a process that's based on ARM64 or x64.

Use SQL Edge when you need to:

- Capture continuous data streams in real time
- Integrate the data in a comprehensive organizational data solution
- Synchronization and connectivity to back-end systems
- Overcome slow or intermittent broadband connections



© Copyright Microsoft Corporation. All rights reserved.

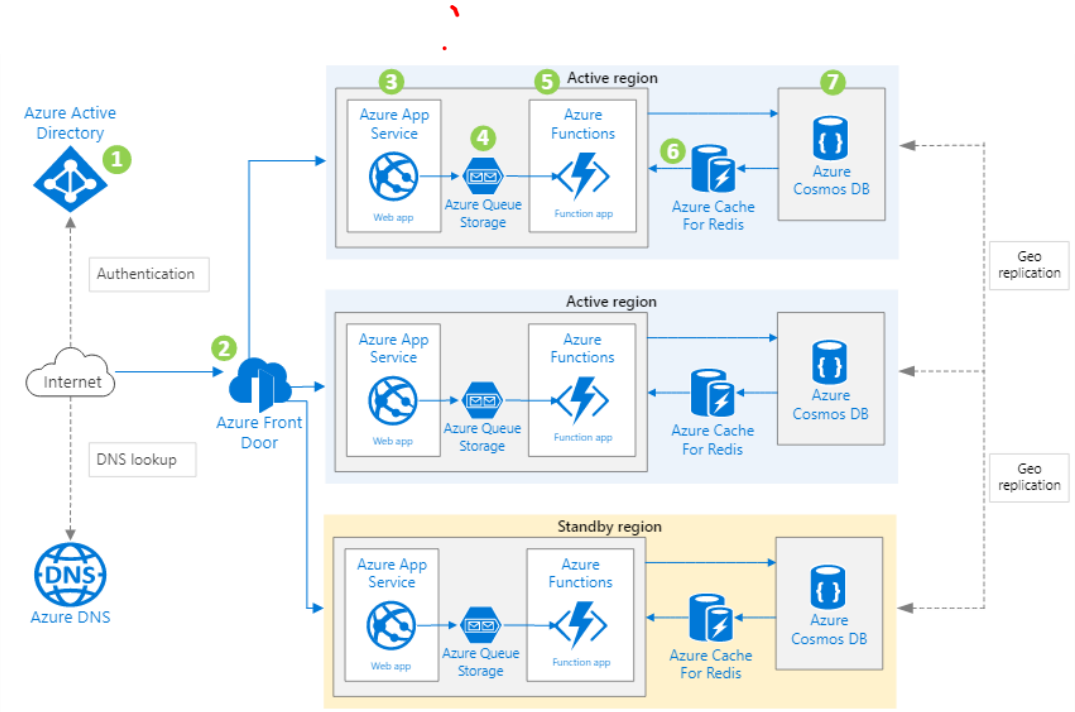
Design for Azure Cosmos DB

When to use Azure Cosmos DB

SQL
Mongo
Gremlin
Table
→ Cosmos DB

A fully managed NoSQL database service for modern app development. It has single-digit millisecond response times and guaranteed speed at any scale.

- Web and mobile applications that store and query user generated content like Tweets or blog posts
- Retail and marketing industry that store catalog data and event sourcing in order proccing pipelines
- Gaming that requires single-millisecond latencies for reads and writes and can handle massive spikes in request rates during new game launches or feature updates.
- IoT use cases can load data into Azure Cosmos DB for adhoc querying. New data and changes to existing data can be read on change feed. Then all data or just changes to data in Azure Cosmos DB can be used as reference data as part of real-time analytics.



© Copyright Microsoft Corporation. All rights reserved.

Azure Storage tables and Azure Cosmos DB tables

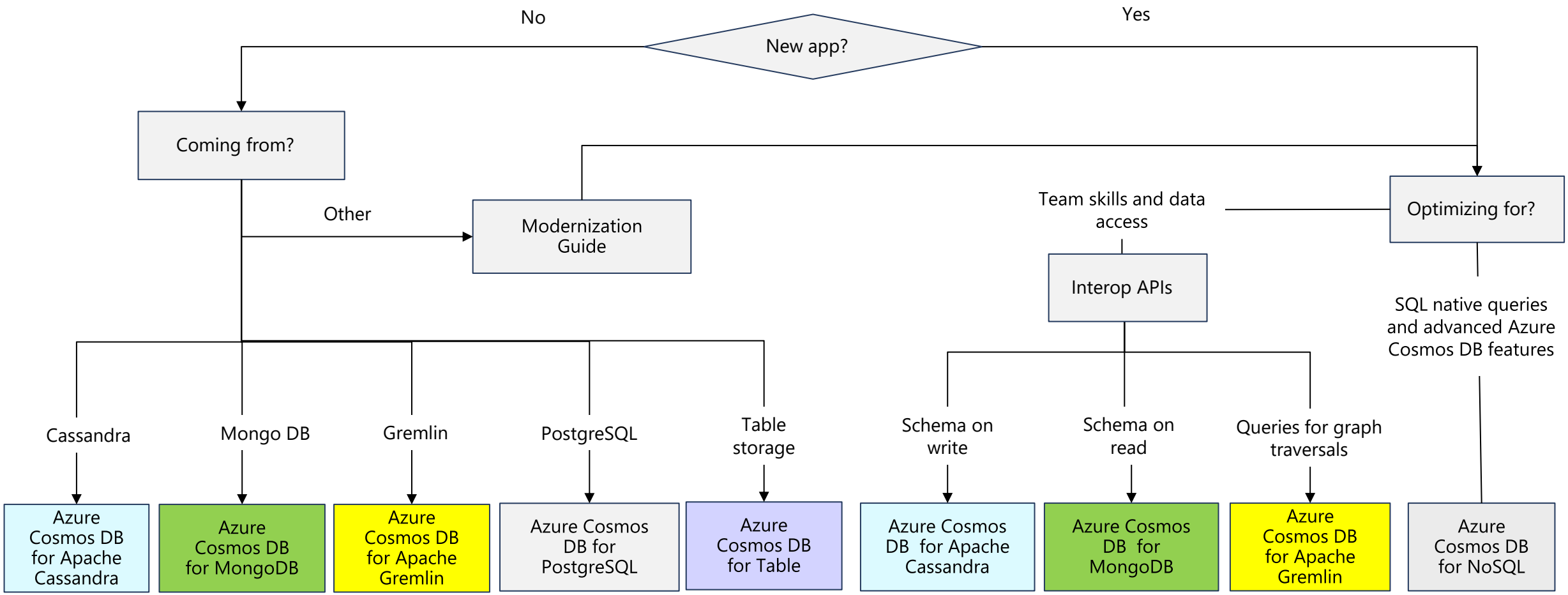
- **Azure Table storage** is a service that stores non-relational structured data (also known as structured NoSQL data) in the cloud, providing a key/attribute store with a schemeless design.
- **Azure Cosmos DB for Table** provides the Table API for applications that are written for Azure Table storage and that need premium capabilities like high availability, scalability, and dedicated throughput.

Differences in behavior

- You are charged for the capacity of an Azure Cosmos DB table as soon as it is created, even if that capacity isn't used.
- Query results from Azure Cosmos DB are not sorted in order of partition key and row key as they are from Storage tables.
- Row keys in Azure Cosmos DB are limited to 255 bytes.
- Table names are case-sensitive in Azure Cosmos DB. They are not case-sensitive in Storage tables.

© Copyright Microsoft Corporation. All rights reserved.

What Azure Cosmos DB APIs are supported?



© Copyright Microsoft Corporation. All rights reserved.

Case study and review

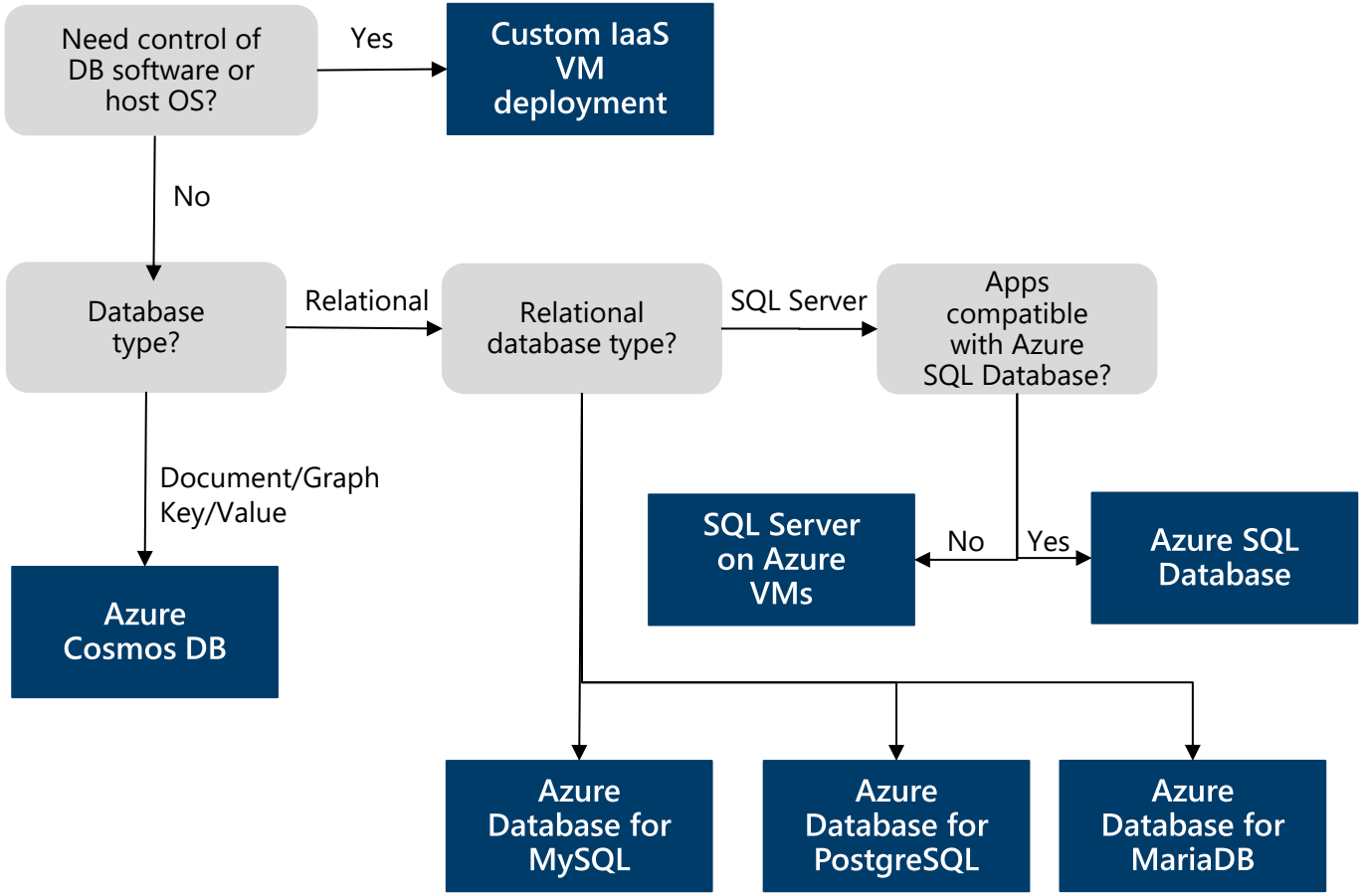
Select a structured data product (activity)

You need a globally distributed, multi-model database with support for NoSQL choices.

You need a fully managed, scalable MySQL relational database that has high availability and security built in at no extra cost.

You need a fully managed relational database that provisions quickly, scales on the fly, and includes built-in intelligence and security.

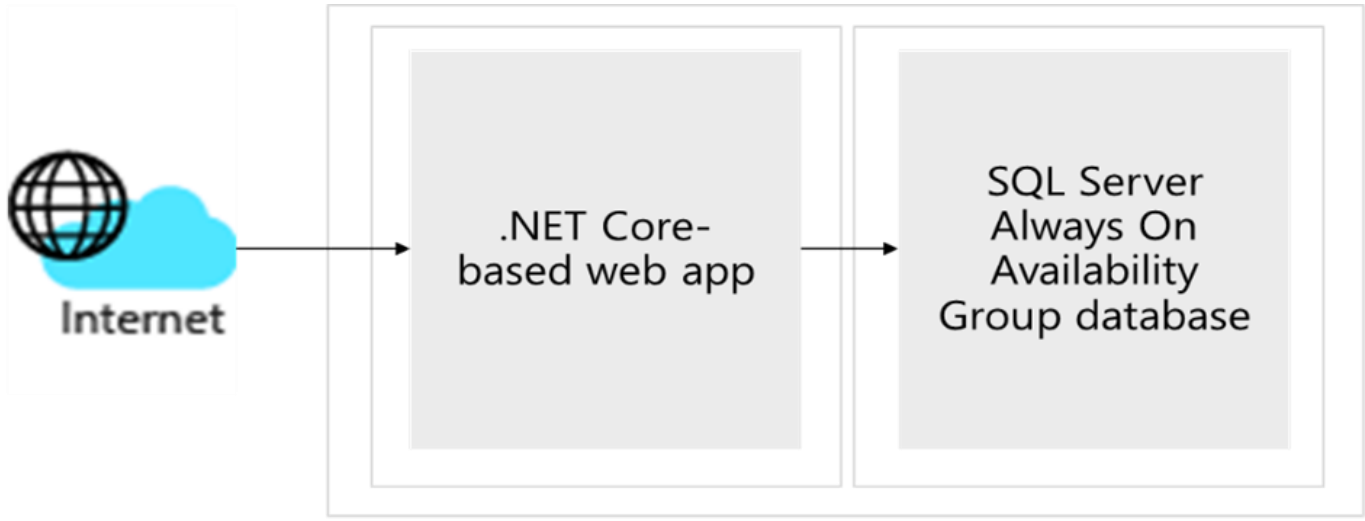
You need to host enterprise SQL Server applications in the cloud and have full control over the server OS.



© Copyright Microsoft Corporation. All rights reserved.

Case Study – Relational data

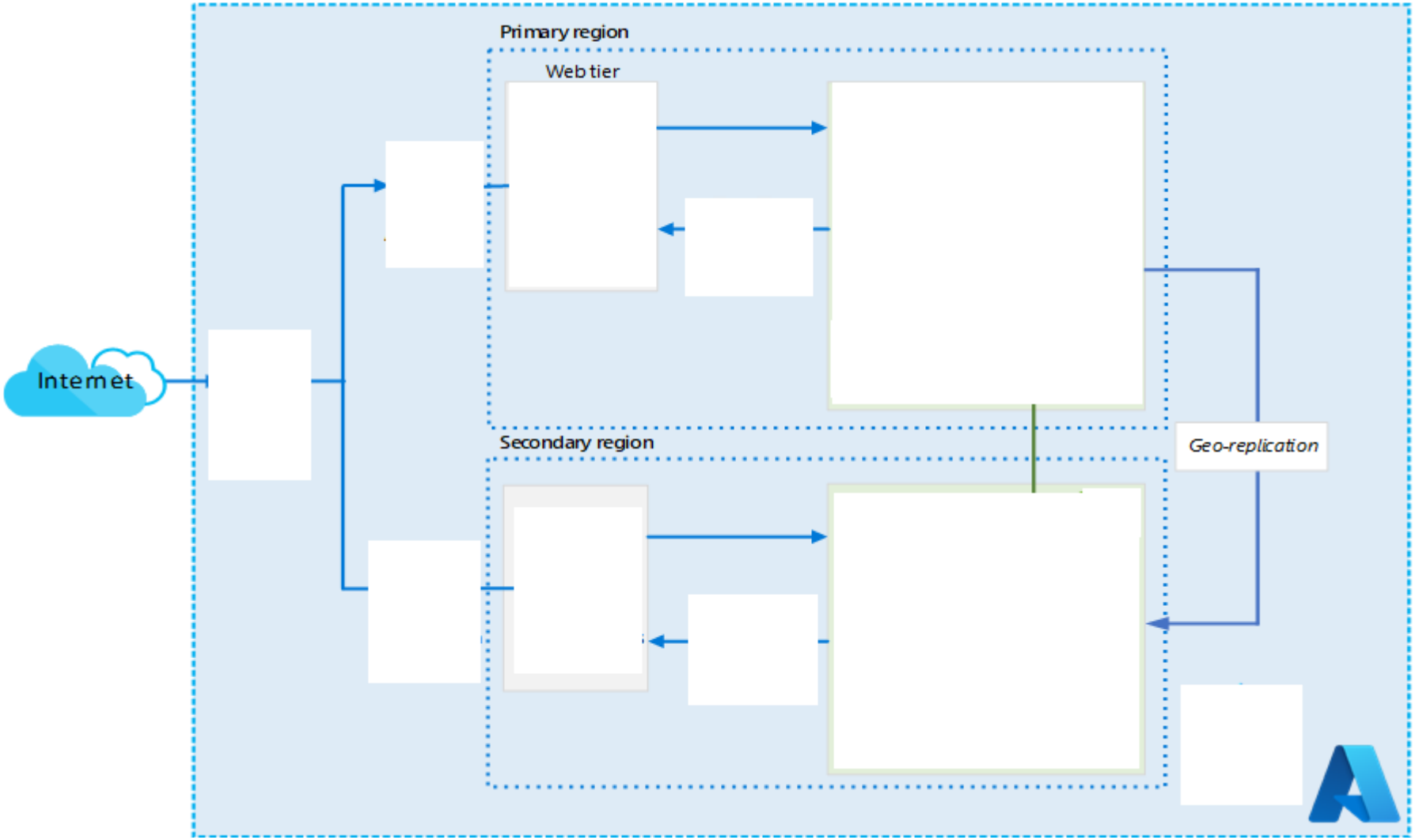
- Design a database solution.
- Your design should include authorization, authentication, pricing, performance, and high availability.
- Diagram what you decide and explain your solution.



- 2-tier Windows based .NET Core-based web app
- Provides access to the product catalog hosted in a SQL Server
- Categorized as mission-critical and requires high availability provisions

© Copyright Microsoft Corporation. All rights reserved.

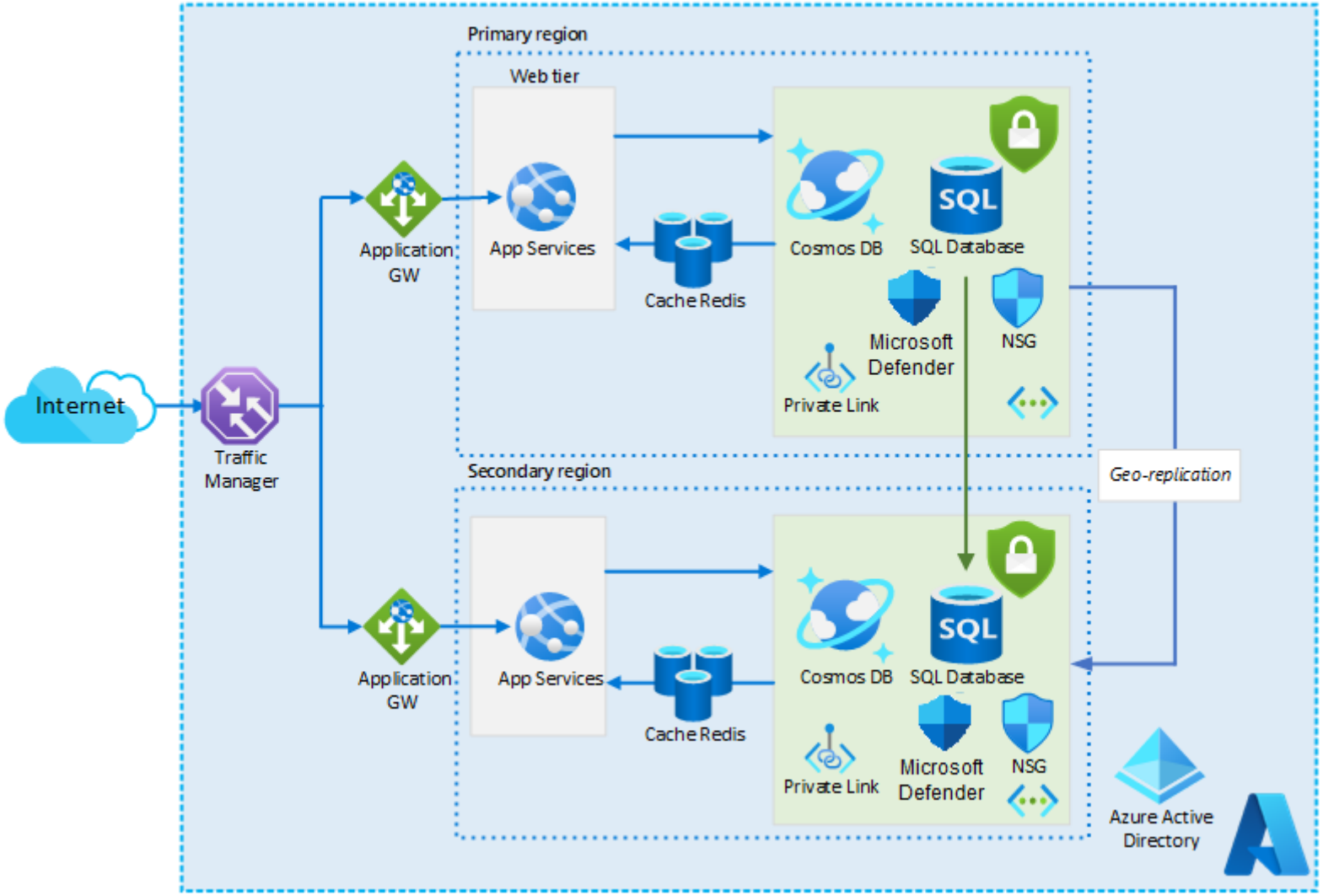
Instructor Solution Diagram



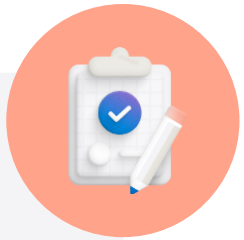
VNet
NSG
App Services
Cosmos DB
Azure AD
Cache Redis
SQL DB
Private Link
Microsoft Defender
App GW
Traffic Manager

© Copyright Microsoft Corporation. All rights reserved.

Instructor Solution Diagram (completed)



Learning recap – relational data solutions



Check your knowledge questions and review

- Choose the appropriate API for Azure Cosmos DB
- Introduction to securing data at rest on Azure
- Secure your Azure SQL database
- Scale multiple Azure SQL Databases with SQL elastic pools
- Configure database authentication and authorization

End of presentation