AZ-305T00A Designing Microsoft Azure Infrastructure Architect

Design a non-relational data storage solution

https://learn.microsoft.com/training/modules/design-data-storage-solution-for-non-relational-data/

Learning Objectives

- Design for data storage
- Design for Azure storage accounts
- Design for data redundancy
- Design for Azure blob storage
- Design for Azure files
- Design an Azure disk solutions
- Design for storage security
- Case study
- Learning recap

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

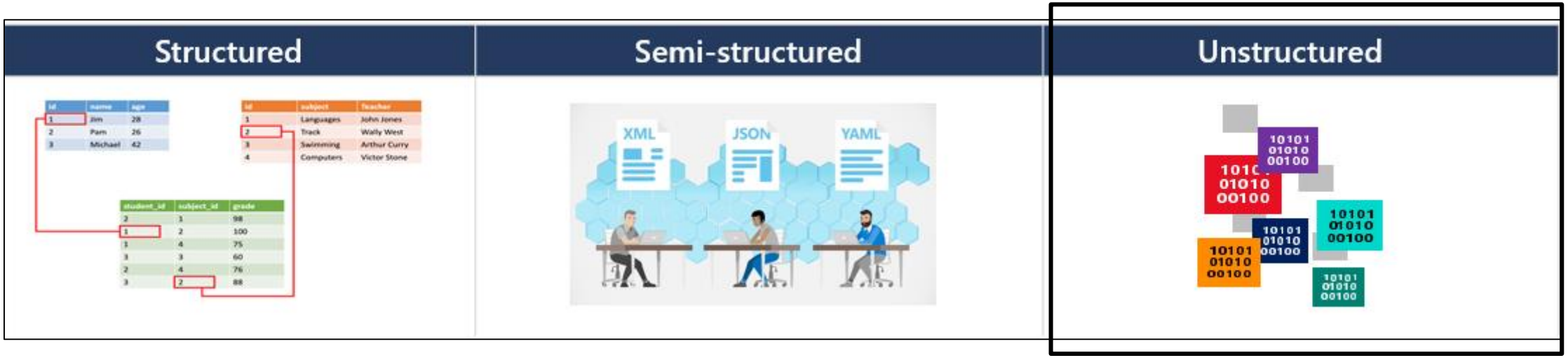
Design data storage solutions for semi-structured and unstructured data

- Recommend a solution for storing semi-structured data
- Recommend a solution for storing unstructured data
- Recommend a data storage solution to balance features, performance, and costs
- Recommend a data solution for protection and durability

Design for data storage

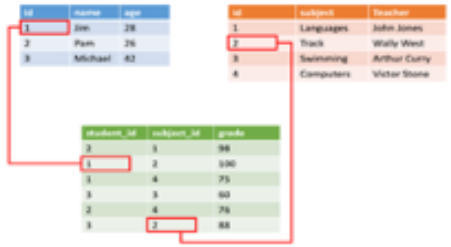
Classify your data storage

The first step in your design for Azure storage is to determine what types of data are required to support the Tailwind Traders organization. In general, data can be classified in three ways: structured, semi-structured, and unstructured.



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

Structured	Semi-structured	Unstructured
		
Structured data is stored in a relational format that has a shared schema. Structured data is often contained in a database table with rows, columns, and keys.	Semi-structured data is less organized. The data fields don't fit neatly into tables, rows, and columns. Semi-structured data contains tags that clarify how the data is organized. The data is defined by using a serialization language.	Unstructured data is the least organized. This data is a mix of information that's stored together, but the data doesn't have a clear relationship. The format of unstructured data is referred to as *non-relational*.
- Relational databases, such as medical records, phone books, and financial accounts - Application data for an e-commerce website	- Hypertext Markup Language (HTML) files - JavaScript Object Notation (JSON) files - Extensible Markup Language (XML) files	- Media files like photos, videos, and audio - Office files, such as Word documents and PowerPoint slides - Text files like PDF, TXT, and RTF

Things to consider when choosing data storage

Non-relational data in Azure can be stored in several different data objects. We'll look at scenarios that implement four storage objects. As you review these options for Tailwind Traders, think about what types of non-relational data are of most interest to your organization. Consider the storage objects that you might need to implement.

- **Consider Azure Blob Storage**. Store vast amounts of unstructured data by using Azure Blob Storage. Blob stands for Binary Large Object. Blob Storage is often used for images and multimedia files.
- **Consider Azure Files**. Provide fully managed file shares in the cloud with Azure Files. This storage data is accessible via the industry standard Server Message Block (SMB) protocol, Network File System (NFS) protocol, and the Azure Files REST API.
- **Consider Azure managed disks**. Support Azure Virtual Machines by using Azure managed disks. These disks are block-level storage volumes that are managed by Azure. Managed disks perform like physical disks in an on-premises server, but in a virtual environment.
- **Consider Azure Queue Storage**. Use Azure Queue Storage to store large numbers of messages. Queue Storage is commonly used to create a backlog of work to process asynchronously.

Design for Azure storage accounts

Determine the best storage account type

Select an account type based on supported services, usage cases, and SLA.

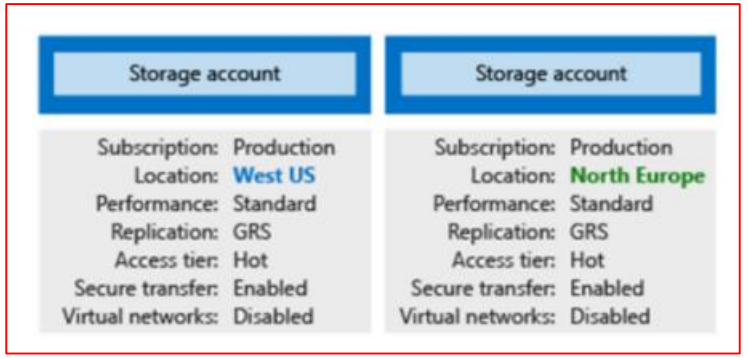
Account Type	Supported services	Usage
Standard general-purpose v2 (default)	Blobs / Data Lake, Queues, Tables, Azure Files	Recommended for most scenarios
Premium block blobs	Blob storage, Data Lake	High transactions rates, single digit storage latency, or large numbers of small transactions
Premium file shares	Azure Files	Enterprise or high-performance scale applications - supports both SMB and NFS file shares
Premium page blobs	Page blobs only	High performance and low latency storage scenarios

After you determine the data storage requirements for your organization, you need to create storage accounts for Tailwind Traders.

Storage account	Supported services	Recommended usage
Standard general-purpose v2	Blob Storage (including Data Lake Storage), Queue Storage, Table Storage, and Azure Files	Standard storage account for most scenarios, including blobs, file shares, queues, tables, and disks (page blobs).
Premium block blobs	Blob Storage (including Data Lake Storage)	Premium storage account for block blobs and append blobs. Recommended for applications with high transaction rates. Use Premium block blobs if you work with smaller objects or require consistently low storage latency. This storage is designed to scale with your applications.
Premium file shares	Azure Files	Premium storage account for file shares only. Recommended for enterprise or high-performance scale applications. Use Premium file shares if you require support for both Server Message Block (SMB) and NFS file shares.
Premium page blobs	Page blobs only	Premium high-performance storage account for page blobs only. Page blobs are ideal for storing index-based and sparse data structures, such as operating systems, data disks for virtual machines, and databases.

Considerations for storage accounts

It is important to plan your storage accounts.



Location

For performance reasons locate the data close to users. One storage account for each location.

Replication

Data storage could have different replication strategies.

Compliance

Regulatory guidelines for keeping data in a specific location / Internal requirements for auditing or storing data.

Administrative overhead

Each storage account requires some time and attention from an administrator to create and maintain.

Cost

The settings for the account do influence the cost of services in the account.

Security - Data sensitivity

Data plane security and data storage security.

Things to consider when choosing storage accounts

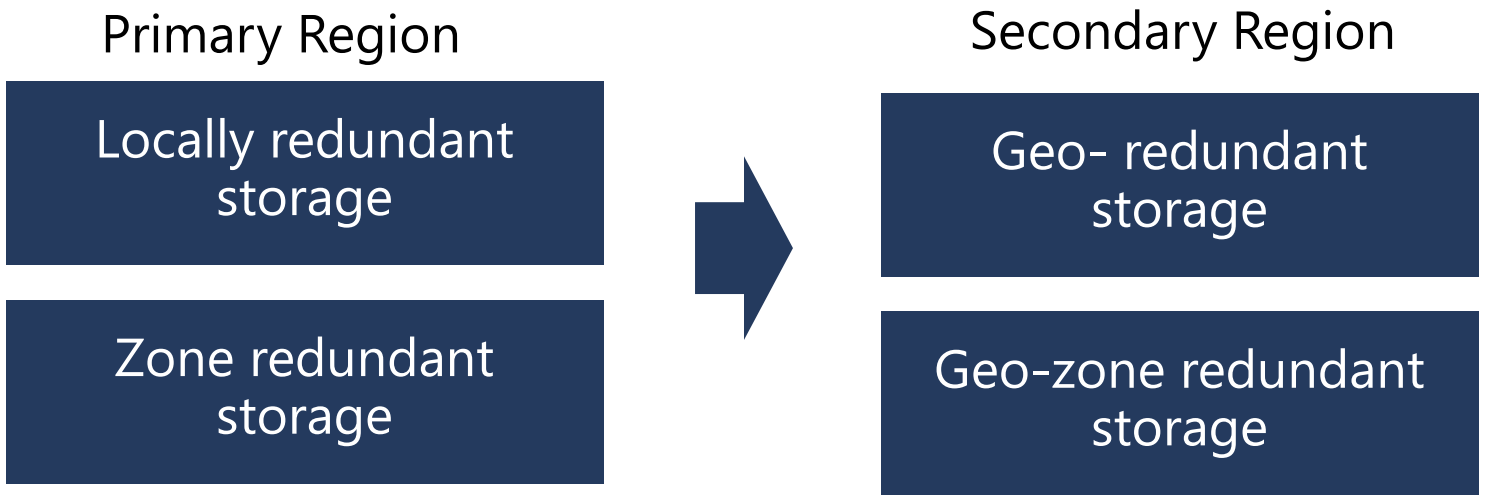
You've reviewed Azure storage account options and some scenarios for when to use different types of storage accounts. Take a few minutes to think about the storage accounts in the Tailwind Traders organization. If you're already using storage accounts, explore how well the configuration meets the business scenarios.

- **Consider your storage locations**. Locate data storage close to where it's most frequently used to increase performance. Does Tailwind Traders have data that's specific to a country or region? You might need a storage account to best support each location.
- **Consider compliance requirements**. Examine regulatory guidelines for Tailwind Traders business scenarios. Are there guidelines for keeping data in a specific location? Does your company have internal requirements for auditing or storing data? You might require different storage accounts to meet the different requirements.
- **Consider data storage costs**. Factor in data storage costs into your plan for Tailwind Traders. A storage account by itself has no financial cost. But, the settings you choose for the account do influence the cost of services in the account. Geo-redundant storage costs more than locally redundant storage. Premium performance and the hot access tier increase the cost of blobs. Do you need to keep track of expenses or billing by department or project? Are you working with partners where storage costs need to be separated? By creating multiple storage accounts, you can better control the overall costs.
- **Consider replication scenarios**. Configure data storage to support different replication strategies. You could partition your data into critical and non-critical categories. You could place Tailwind Traders critical data into a storage account with geo-redundant storage. You could put Tailwind Traders non-critical data in a different storage account with locally redundant storage.
- **Consider administrative overhead**. Plan for administrative overhead in your Tailwind Traders storage design. Each storage account requires some time and attention from an administrator to create and maintain. Using multiple storage accounts increases the complexity for users who add data to your cloud storage. Users in this role need to understand the purpose of each storage account to ensure they add new data to the correct account.
- **Consider data sensitivity**. Protect sensitive and proprietary Tailwind Traders data in your data storage. You can enable virtual networks for proprietary data and not for public data. This scenario might require separate storage accounts.
- **Consider data isolation**. Segregate regulatory and compliance data, or local policies by using multiple storage accounts for Tailwind Traders. You can separate data in one application from data in another application to ensure data isolation.

Design for data redundancy

Select a storage replication strategy

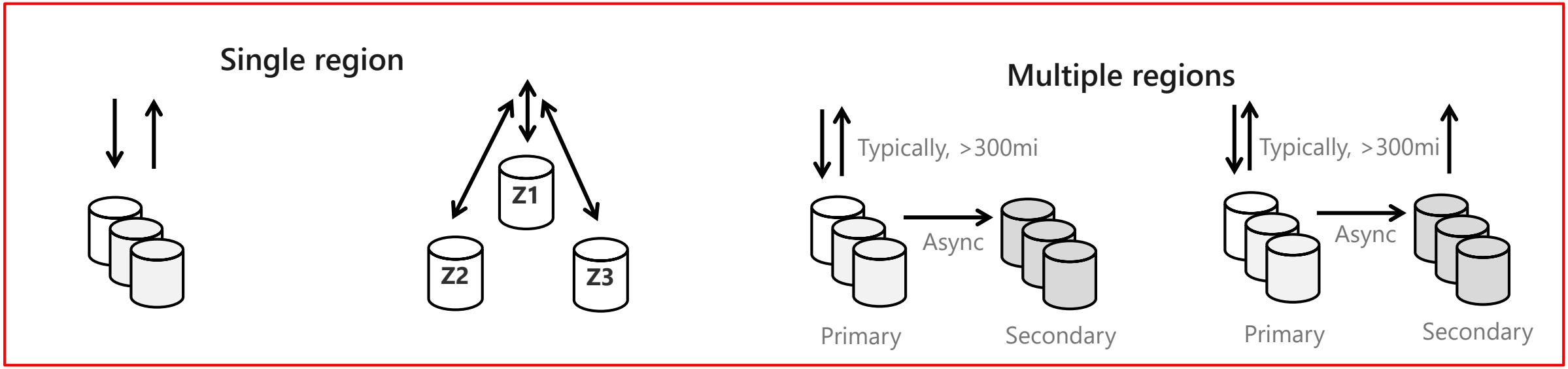
What level of redundancy do you need?



Read and write access to the secondary region are available if the primary region becomes unavailable

- A node within a data center becomes unavailable
- An entire data center (zonal or non-zonal) becomes unavailable
- A region-wide outage occurs in the primary region

Determine Replication Strategies (1 of 2)



LRS

- Three replicas, one region
- Protects against disk, node, rack failures
- Write is acknowledged when all replicas are committed
- Superior to dual-parity RAID

ZRS

- Three replicas, three zones, one region
- Protects against disk, node, rack, and zone failures
- Synchronous writes to all three zones

GRS

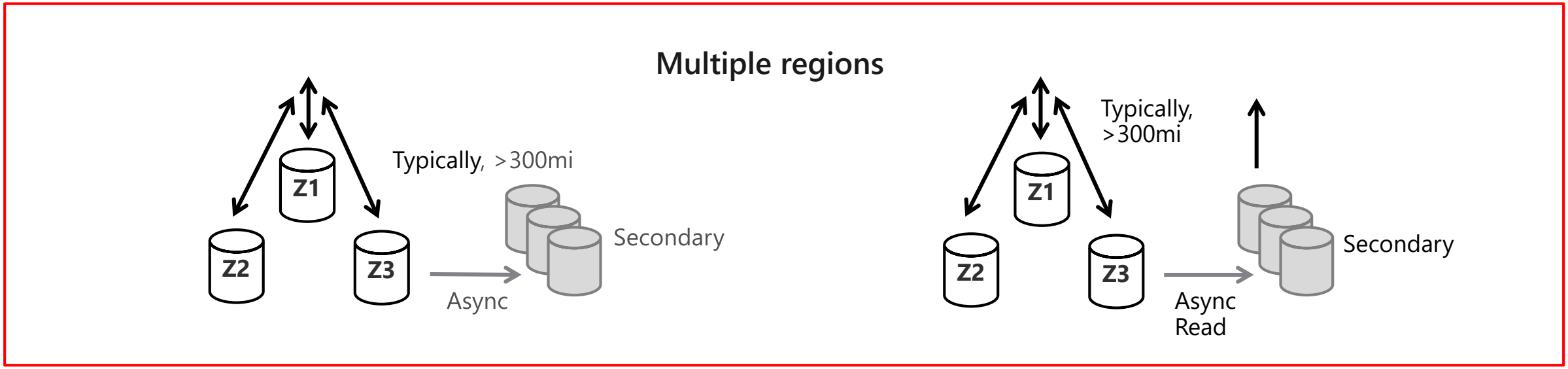
- Six replicas, two regions (three per region)
- Protects against major regional disasters
- Asynchronous copy to secondary

RA-GRS

- GRS + read access to secondary
- Separate secondary endpoint
- Recovery point objective (RPO) delay to secondary can be queried

Continued next slide

Determine Replication Strategies (2 of 2)



GZRS

- Six replicas, 3+1 zones, two regions
- Protects against disk, node, rack, zone, and region failures
- Synchronous writes to all three zones and asynchronous copy to secondary

RA-GZRS

- GZRS + read access to secondary
- Separate secondary endpoint
- RPO delay to secondary can be queried

Access Storage

Every object has a unique URL address – based on account name and storage type

Container service: https://*mystorageaccount*.blob.core.windows.net
Table service: https://*mystorageaccount*.table.core.windows.net
Queue service: https://*mystorageaccount*.queue.core.windows.net
File service: https://*mystorageaccount*.file.core.windows.net

If you prefer you can configure a custom domain name

CNAME record	Target
blobs.contoso.com	contosoblobs.blob.core.windows.net

Things to consider when using data redundancy

You've reviewed the different options for implementing replication. Data redundancy is accomplished through a primary region and paired secondary region. As you plan the storage accounts and redundancy settings for Tailwind Traders, consider the following factor.

- **Consider primary replication options**. Explore different scenarios for how Tailwind Traders data can be replicated in the primary region. The redundancy options present tradeoffs between lower costs and higher availability. Some business centers can require more data redundancy. Specific departments or regions might work with data that's not sensitive or which doesn't require high durability. You can implement multiple storage accounts with different redundancy to control the overall costs across the organization.

- **Consider locally redundant storage**. Implement LRS for a low cost redundancy solution, but with limited durability. LRS is suited for Tailwind Traders apps that store data that can be easily reconstructed if data loss occurs. LRS is also a good choice for apps that are restricted to replicating data only within a country or region due to data governance requirements.

- **Consider zone-redundant storage**. Choose ZRS for excellent performance, low latency, and resiliency for your data if it becomes temporarily unavailable. Keep in mind that ZRS by itself might not protect your data against a regional disaster where multiple zones are permanently affected.

- **Consider secondary regions**. For applications requiring high durability, you can choose to additionally copy the data in your storage account to a secondary region that is hundreds of miles away from the primary region. If your storage account is copied to a secondary region, then your data is durable even in the case of a complete regional outage or a disaster in which the primary region isn't recoverable.

- **Consider read access requirements**. Identify Tailwind Traders applications that require read access to the replicated data in the secondary region, if the primary region becomes unavailable for any reason. Configure your storage account with read access to the secondary region. Your applications can seamlessly shift to reading data from the secondary region if the primary region becomes unavailable.

Design for Azure blob storage

Determine the storage tier

Blob storage is an object store used for storing vast amounts of unstructured data.

Tier	Storage Duration	Usage cases
Premium	N/A	• High throughout and large numbers of I/O operations per second
Standard Hot	N/A	• Active and frequent use • Data staged for processing
Standard Cool	> 30 days	• Short-term backup • Older media infrequently viewed • Large data sets
Standard Cold	> 90 days	
Standard Archive	> 180 days	• Long-term backup • Original (raw) data • Compliance or archival data

- Use lifecycle rules to manage the storage tiers

Comparison	Hot access tier	Cool access tier	Cold access tier	Archive access tier
Availability	99.9%	99%	99%	99%
Availability (RA-GRS reads)	99.99%	99.9%	99.9%	99.9%
Latency (time to first byte)	milliseconds	milliseconds	milliseconds	hours
Minimum storage duration	N/A	30 days	90 days	180 days

Things to know about Azure Blob immutable storage

Immutable storage for Azure Blob Storage enables users to store business-critical data in a WORM (Write Once, Read Many) state. While in a WORM state, data can't be modified or deleted for a user-specified interval. By configuring immutability policies for blob data, you can protect your data from overwrites and deletes. Policies are applied at the container level and audit logs are available.

Azure Blob Storage supports two forms of immutability policies for implementing immutable storage:

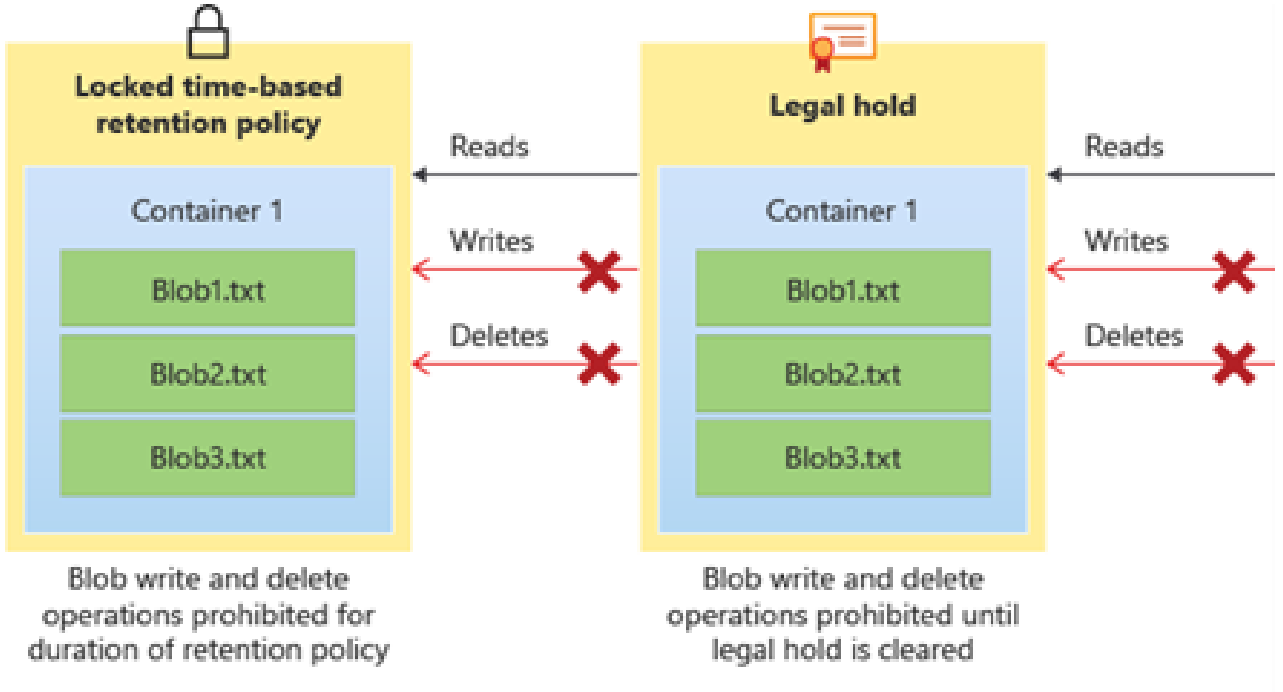
- Time-based retention policies let users set policies to store data for a specified interval. When a time-based retention policy is in place, objects can be created and read, but not modified or deleted. After the retention period has expired, objects can be deleted, but not overwritten. The Hot, Cool, and Archive access tiers support immutable storage by using time-retention policies.

- Legal hold policies store immutable data until the legal hold is explicitly cleared. When a legal hold is set, objects can be created and read, but not modified or deleted. Premium Blob Storage uses legal holds to support immutable storage.

Consider immutable storage policies

Determine regulatory compliance, secure document retention, and legal hold policies.

- Apply immutable storage policies at the container level
- Use **time-based retention policies** for business-critical data
- Use **legal-hold policies** for sensitive information to ensure a tamper proof state
- Policies apply to all objects within the container
- Audit logs are available



Things to consider when implementing Azure Blob Storage

You've reviewed the different access options for Azure Blob Storage, and how to use immutable storage. Take a few minutes to determine how you can configure Azure Blob Storage for Tailwind Traders.

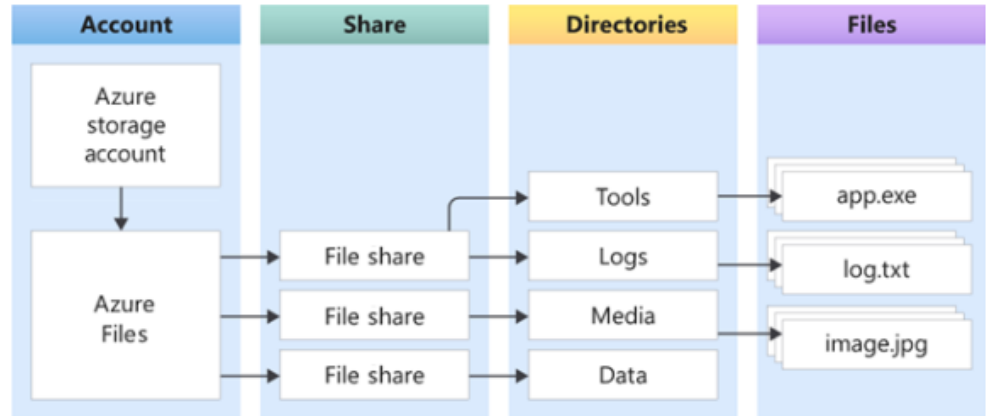
- **Consider Blob Storage availability**. Determine the level of availability required for your data. Are there scenarios where offline data is sufficient? The Archive access tier is optimized for data that can remain offline for hours.
- **Consider Blob Storage latency**. Plan for the required time to access the first byte of data in different scenarios. Some work tasks require instant access to data, while others can accommodate some delay. Premium Blob Storage supports single-digit millisecond latency for data, while the Hot and Cool access tiers support latency in milliseconds.
- **Consider Blob Storage costs**. Weigh your options for total cost. Factor in data storage minimum durations, and potential charges for transactions and access. Premium Blob Storage and the Hot access tier have higher overall storage costs, but lower charges for access and transactions. The Cool and Archive access tiers offer lower storage costs, but tend to have higher charges for access and transactions.
- **Consider immutable storage**. Review your business scenarios to identify where you might need immutable storage. Consider the different types immutability policies and which form satisfies your organization's requirements.

Design for Azure files

Design for Azure Files

5 minutes

Azure Files provides fully managed cloud-based file shares that are hosted on Azure. Shared files are accessible by using the industry standard Server Message Block (SMB) protocol, Network File System (NFS) protocol, and the Azure Files REST API. You can mount or connect to an Azure file share at the same time on all the main operating systems.



Things to know about Azure Files

Azure Files can be used to add to or replace a company's existing on-premises network attached storage (NAS) devices or file servers. Here are some reasons why your organization might want to use Azure Files:

- Developers can store apps and configuration files in a file share and connect new VMs to the shared files. This action reduces the time to get new machines into production.
- With file shares on Azure, a company doesn't need to buy and deploy expensive redundant hardware and manage software updates. The shares are cross-platform, and you can connect to them from Windows, Linux, or macOS.
- All the resilience of the Azure platform is inherited by your file share, which makes files globally redundant. You also gain options to use the integrated snapshots feature, and set up automatic backups by using Recovery Services vaults.
- All the data is encrypted in transit by using HTTPS and is stored encrypted when at rest.

Compare Azure files to Azure blobs

The technology you choose depends on the use case, protocol, and performance.

Category	Azure Files	Azure Blob Storage
Use cases	• Replace or supplement traditional on-premises file servers or NAS devices • Access files shares from anywhere • Lift and shift content to the cloud • Replicate and cache with Azure File Sync • Share stored application settings	• Large scale analytical data • Throughput sensitive high-performance computing • Backup and archive • Autonomous driving, media rendering, or genomic sequencing data
Available protocols	• SMB • REST • NFS 4.1	• NFS 3.0 • REST • Data Lake Storage Gen2
Performance (Per volume)	• Better IOPS	• Better throughput

Choose your data access method

To move your company's shared files into Azure Files, you need to analyze your options and make an important decision. How are you going to access and update the files? You could replace your existing Server Message Block (SMB) file shares with their equivalent in Azure Files. Another option is to set up an instance of Azure File Sync. If you choose to use Azure File Sync, there's more flexibility on how files are secured and accessed.

Azure file shares can be used in two ways. You can directly mount serverless Azure file shares (SMB) or cache Azure file shares on-premises by using Azure File Sync.

- **Direct mount of Azure file shares**: Because Azure Files provides SMB access, you can mount Azure file shares on-premises or in the cloud. Mounting uses the standard SMB client available in Windows, macOS, and Linux. Because Azure file shares are serverless, deploying for production scenarios doesn't require managing a file server or NAS device. Direct mounting means you don't have to apply software patches or swap out physical disks.

- **Cache Azure file shares on-premises with Azure File Sync**: Azure File Sync lets you centralize your organization's file shares. Azure Files provides the flexibility, performance, and compatibility of an on-premises file server. Azure File Sync transforms an on-premises (or cloud) Windows Server into a quick cache of your Azure file share.

Select a file storage tier (activity)

Tailor your file tiers to the performance and price you need

File storage tiers	
Premium	You have highly I/O-intensive workloads, with high throughput and low latency
Transaction optimized	You need storage optimized for general purpose file sharing scenarios such as team shares and Azure File Sync
Hot	You need cost-efficient storage optimized for online archive storage scenarios
Cool	You have transaction heavy workloads and applications that require file storage and backend storage

Determine your storage tier

Azure Files offers four tiers of storage. These tiers allow you to tailor your file shares to meet the performance and price requirements for your scenarios.

- **Premium**: File shares are backed by solid-state drives (SSDs) and provide consistent high performance and low latency. Used for the most intensive IO workloads. Suitable workloads include databases, web site hosting, and development environments. Can be used with both Server Message Block (SMB) and Network File System (NFS) protocols.
- **Transaction optimized**: Used for transaction heavy workloads that don't need the latency offered by premium file shares. File shares are offered on the standard storage hardware backed by hard disk drives (HDDs).
- **Hot access tier**: Storage optimized for general purpose file sharing scenarios such as team shares. Offered on standard storage hardware backed by HDDs.
- **Cool access tier**: Cost-efficient storage optimized for online archive storage scenarios. Offered on storage hardware backed by HDDs.

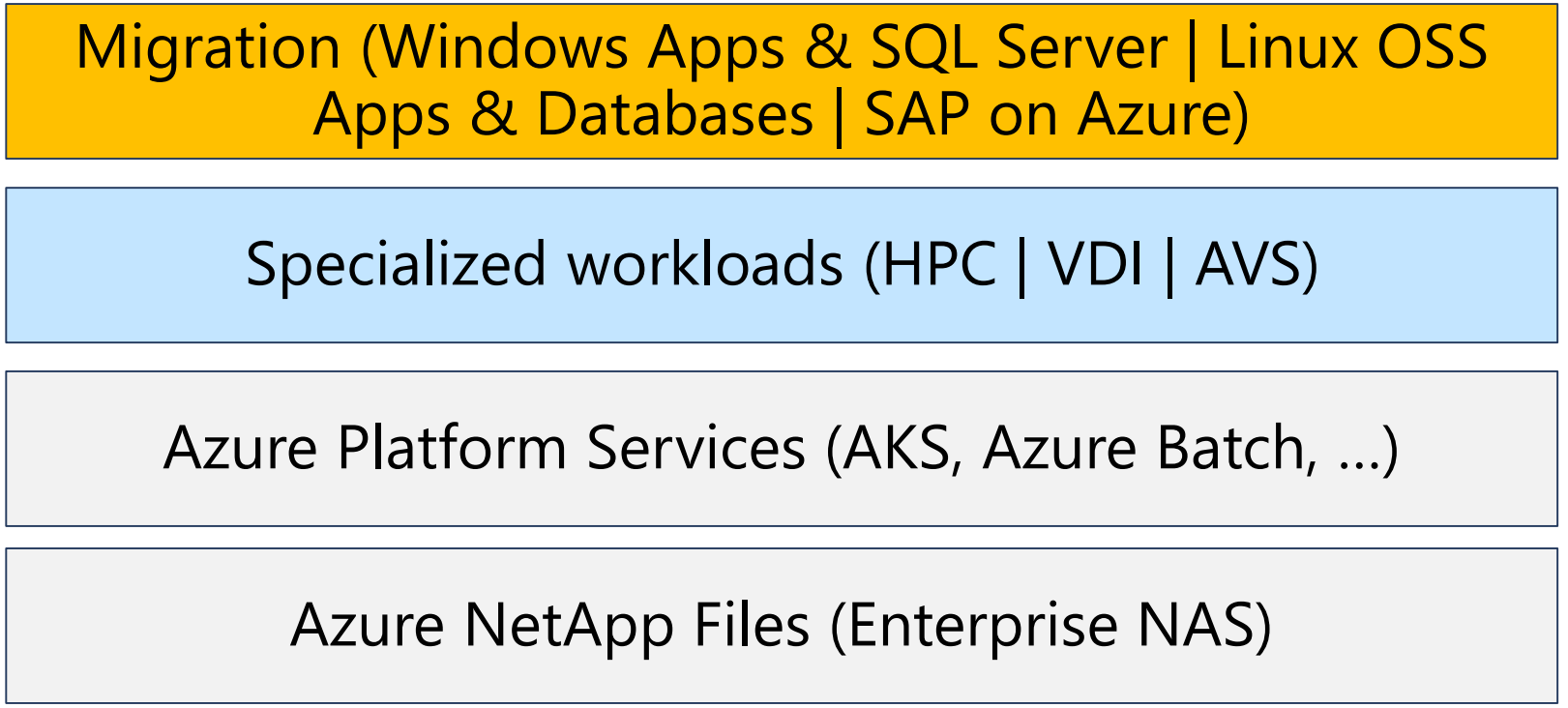
Things to consider when choosing your implementation

Comparison	Azure Blob Storage	Azure Files	Azure NetApp Files
Description	Azure Blob Storage is best suited for large scale read-heavy sequential access workloads where data is ingested once and modified later. Blob Storage offers the lowest total cost of ownership, if there's little or no maintenance.	Azure Files is a highly available service best suited for random access workloads. For NFS shares, Azure Files provides full POSIX file system support and can easily be used from container platforms like Azure Container Instance (ACI) and Azure Kubernetes Service (AKS) with the built-in CSI driver, in addition to VM-based platforms.	Azure NetApp Files is a fully managed file service in the cloud, powered by NetApp, with advanced management capabilities. Azure NetApp Files is suited for workloads that require random access and provides broad protocol support and data protection capabilities.
Use cases	Large scale analytical data, Throughput sensitive high-performance computing, Backup and archive, Autonomous driving, Media rendering, or Genomic sequencing	Shared files, Databases, Home directories, Traditional applications, ERP, CMS, NAS migrations that don't require advanced management, Custom applications that require scale-out file storage	On-premises enterprise NAS migration that requires rich management capabilities, Latency sensitive workloads like SAP HANA, Latency-sensitive or IOPS intensive high performance compute, Workloads that require simultaneous multi-protocol access
Available protocols	- NFS 3.0 - REST - Data Lake Storage Gen2	- SMB - NFS 4.1 - REST	- NFS 3.0 and 4.1 - SMB
Performance (per volume)	Up to 20,000 IOPS, up to 15 GiB/s throughput	Up to 100,000 IOPS, up to 10 GiB/s throughput	Up to 460,000 IOPS, up to 4.5 GiB/s throughput for regular volumes, up to 10 GiB/s throughput for large volumes

Design for NetApp files

The Azure NetApp Files service is enterprise-class, high-performance, metered file storage.

- Ease of migration
- Workload scale
- Flexibility
- Storage technology



Design an Azure disk solution

Select an Azure disk solution

Azure disks are block-level storage volumes used with Azure virtual machines.

- Consider disk type, scenario, throughput, and IOPS
- Always use managed disks
- Optimize read and write access with disk caching
- Use Azure Disk Encryption
- Enhance performance with multiple disks
- Use the network acceleration feature
- Share disks across multiple VMs

Disk type	Usage cases
Ultra-disk SSD	IO-intensive workloads such as SAP HANA, top tier databases (SQL, Oracle), and other transaction-heavy workloads
Premium SSD v2	Production and performance-sensitive workloads that consistently require low latency and high IOPS and throughput
Premium SSD	Production and performance sensitive workloads
Standard SSD	Web servers, lightly used enterprise applications and dev/test
Standard HDD	Backup, non-critical, infrequent access

Comparison	Ultra-disk	Premium SSD	Standard SSD	Standard HDD
Disk type	SSD	SSD	SSD	HDD
Scenario	IO-intensive workloads, such as SAP HANA, top tier databases like SQL Server and Oracle, and other transaction-heavy workloads	Production and performance sensitive workloads	Web servers, Lightly used enterprise applications, Development and testing	Backup, Non-critical, Infrequent access
Max throughput	2,000 Mbps	900 Mbps	750 Mbps	500 Mbps
Max IOPS	160,000	20,000	6,000	2,000

Choose an encryption option

There are several encryption types available for your managed disks.

- Azure Disk Encryption (ADE) encrypts the VM's virtual hard disks (VHDs). If VHD is protected with ADE, the disk image is accessible only by the VM that owns the disk.
- Server-Side Encryption (SSE) is performed on the physical disks in the data center. If someone directly accesses the physical disk, the data will be encrypted. When the data is accessed from the disk, it's decrypted and loaded into memory. This form of encryption is also referred to as *encryption at rest* or Azure Storage encryption.
- Encryption at host ensures that data stored on the VM host is encrypted at rest and flows encrypted to the Storage service. Disks with encryption at host enabled aren't encrypted with SSE. Instead, the server hosting your VM provides the encryption for your data, and that encrypted data flows into Azure Storage.

Things to consider when using managed disks

Think about what data disk types are needed for Tailwind Traders. Consider your scenarios, throughput, and IOPS.

- **Consider your scenarios, throughput, and IOPS.** Compare disk types and choose the data disks that satisfy your business scenarios, and throughput and IOPS requirements. For more information, see Select a disk type for Azure IaaS VMs - managed disks
  - **Ultra-disk storage**: Azure Ultra Disk storage provides the best performance. Choose this option when you need the fastest storage performance in addition to high throughput, high input/output operations per second (IOPS), and low latency. Ultra-disk storage might not be available in all regions.
  - **Premium SSD storage**: Azure Premium SSD-managed disks provide high throughput and IOPS with low latency. These disks offer a slightly less performance compared to Ultra Disk Storage. Premium SSD storage is available in all regions.
  - **Standard SSD**: Azure Standard SSD-managed disks are a cost-effective storage option for VMs that need consistent performance at lower speeds. Standard SSD disks aren't as fast as Premium SSD disks or Ultra Disk Storage. You can attach Standard SSD disks to any VM.
  - **Standard HDD**: In Azure Standard HDD-managed disks, data is stored on conventional magnetic disk drives that have moving spindles. Disks are slower and the variation in speeds is higher compared to solid-state drives (SSDs). Like Standard SSD disks, you can use Standard HDD disks for any VM.
- **Consider data caching.** Improve performance with disk caching. Azure Virtual Machines disk caching optimizes read and write access to the virtual hard disk (VHD) files. The VHDs are attached to Azure Virtual Machines. For OS disks, the default cache setting is `ReadWrite`, and for data disks, the default is `ReadOnly`.

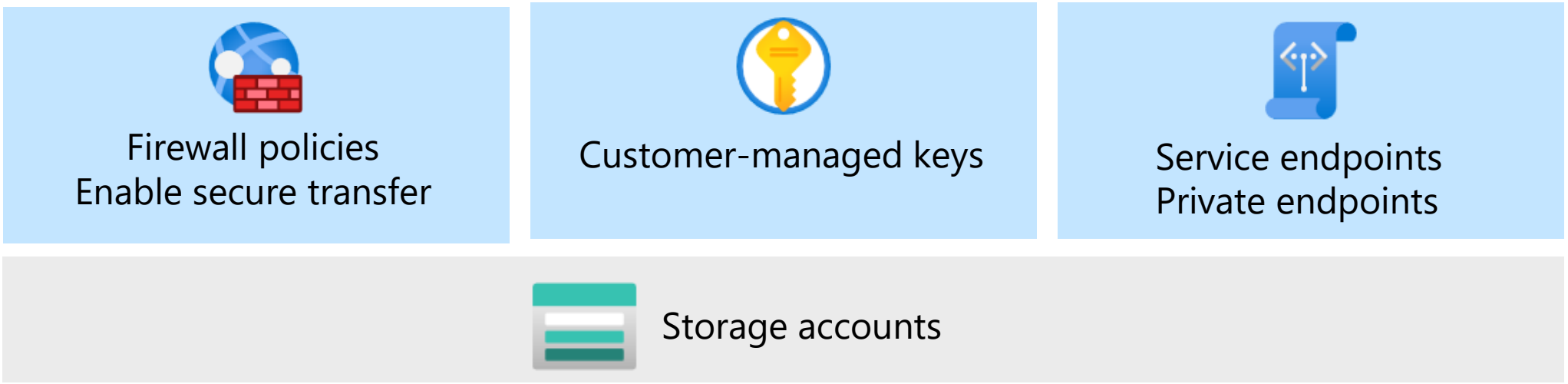
  > ⚠ Warning
  >
  > Disk caching isn't supported for disks 4 TiB and larger. When multiple disks are attached to your Virtual Machine, each disk smaller than 4 TiB supports caching. Changing the cache setting of an Azure disk, detaches and reattaches the target disk. When it's the OS disk, the VM is restarted.

- **Consider using encryption.** Secure your data disks with encryption. To fully protect your data disks, combine encryption services: ADE, SSE, and encryption at rest.

Design for storage security

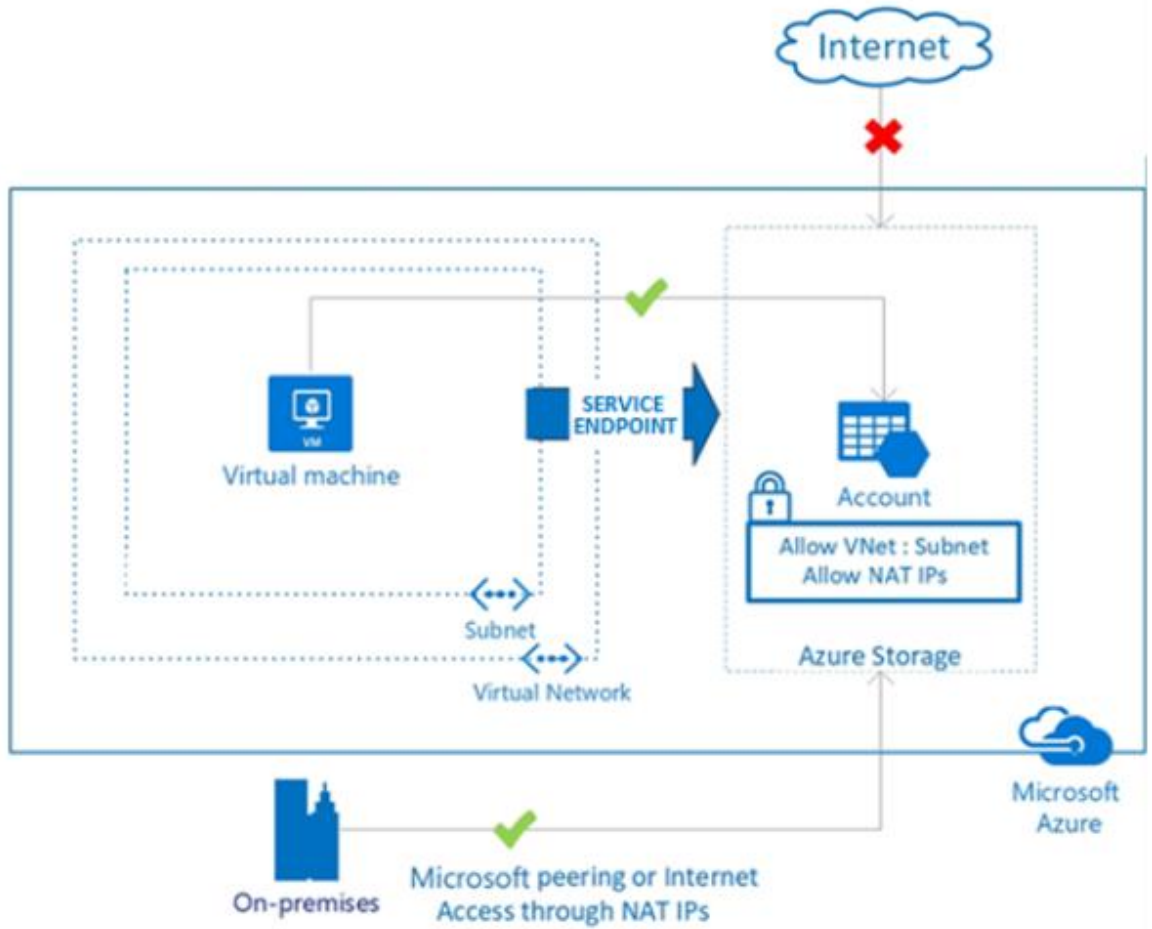
Considerations for storage security

Use a layered security model to secure and control access.

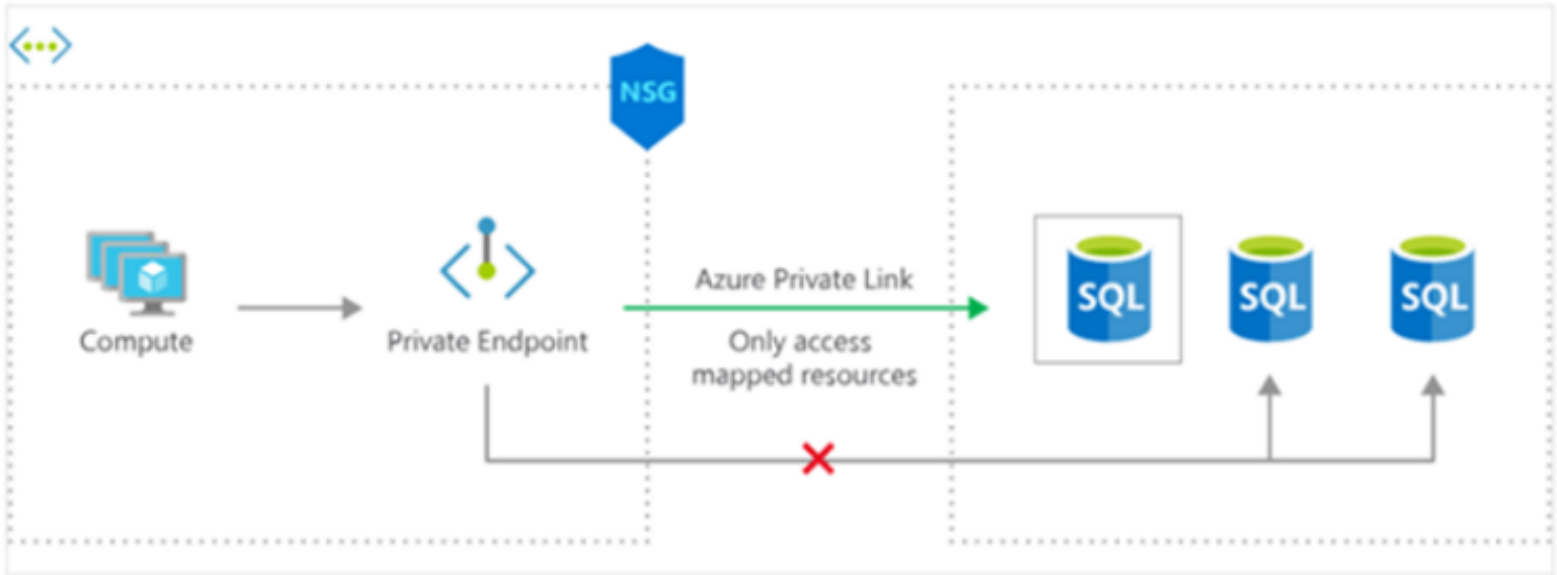


- Grant limited access to Azure Storage resources
- Enable firewall rules to limit access to access - IP addresses or subnets
- Use private endpoints and private links for clients
- Use virtual network service endpoints to provide direct connection
- Use customer managed encryption keys

- **Consider service endpoints**. Secure Azure storage accounts to your virtual networks by using service endpoints. You can provide optimal routing by always keeping traffic destined to Azure Storage on the Azure backbone network. Enable private IP addresses in the virtual network to reach the service endpoint without requiring a public IP address. Allow on-premises networks to access resources by using NAT IP addresses.



- **Consider private endpoints**. Add private endpoints to create a special network interface for an Azure service in your virtual network. When you implement a private endpoint for your storage account, it provides secure connectivity between clients on your virtual network and your storage.

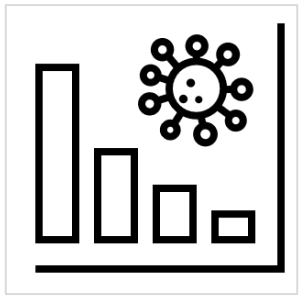


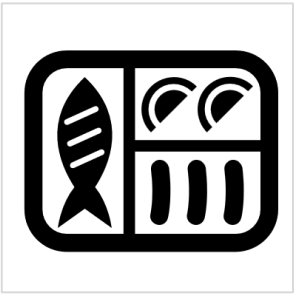
- **Consider secure transfer**. (Microsoft recommended) Always require secure transfer for all your Azure storage accounts. In the Azure portal, choose **Enable secure transfer** for your storage accounts. The `Secure transfer required` property is enabled by default when an Azure storage account is created.

- **Consider customer-managed keys**. Manage encryption keys for your storage account by using customer-managed keys stored in Azure Key Vault. Customer-managed keys give you full control over access to your encryption keys and encrypted data.

Case study and review

Case study – Non-relational data





Media files	Marketing literature	Corporate documents
• Product photos and feature videos • JPEG and MP4 are most common formats	• Customer stories, sales flyers, sizing charts, and eco-friendly manufacturing information • PDF format is the most common	• Internal documents – some sensitive • Mostly Office formats like Word and Excel

Case study discussion

- Design a storage solution for Tailwind Traders.
  - What type of data is represented?
  - What factors will you consider in your design?
  - What type of storage accounts are needed?
  - Will you use blob access tiers?
  - Will you use immutable storage?
  - How will the content be securely accessed?
- Your solution should consider the media, marketing literature, and corporate documents.