

## Introduction

In this module, you'll be introduced to the Azure storage services. You'll learn about the Azure Storage Account and how that relates to the different storage services that are available. You'll also learn about blob storage tiers, data redundancy options, and ways to move data or even entire infrastructures to Azure.

## Learning objectives

After completing this module, you'll be able to:

- Compare Azure storage services.
- Describe storage tiers.
- Describe redundancy options.
- Describe storage account options and storage types.
- Identify options for moving files, including AzCopy, Azure Storage Explorer, and Azure File Sync.
- Describe migration options, including Azure Migrate and Azure Data Box.

## Describe Azure storage accounts

A storage account provides a unique namespace for your Azure Storage data that's accessible from anywhere in the world over HTTP or HTTPS. Data in this account is secure, highly available, durable, and massively scalable.

When you create your storage account, you'll start by picking the storage account type. The type of account determines the storage services and redundancy options and has an impact on the use cases. Below is a list of redundancy options that will be covered later in this module:

- Locally redundant storage (LRS)
- Geo-redundant storage (GRS)
- Read-access geo-redundant storage (RA-GRS)
- Zone-redundant storage (ZRS)
- Geo-zone-redundant storage (GZRS)
- Read-access geo-zone-redundant storage (RA-GZRS)

Type	Supported services	Redundancy Options	Usage
Standard general-purpose v2	Blob Storage (including Data Lake Storage), Queue Storage, Table Storage, and Azure Files	LRS, GRS, RA-GRS, ZRS, GZRS, RA-GZRS	Standard storage account type for blobs, file shares, queues, and tables. Recommended for most scenarios using Azure Storage. If you want support for network file system (NFS) in Azure Files, use the premium file shares account type.
Premium block blobs	Blob Storage (including Data Lake Storage)	LRS, ZRS	Premium storage account type for block blobs and append blobs. Recommended for scenarios with high transaction rates or that use smaller objects or require consistently low storage latency.
Premium file shares	Azure Files	LRS, ZRS	Premium storage account type for file shares only. Recommended for enterprise or high-performance scale applications. Use this account type if you want a storage account that supports both Server Message Block (SMB) and NFS file shares.

Type	Supported services	Redundancy Options	Usage
Premium page blobs	Page blobs only	LRS	Premium storage account type for page blobs only.

## Storage account endpoints

One of the benefits of using an Azure Storage Account is having a unique namespace in Azure for your data. In order to do this, every storage account in Azure must have a unique-in-Azure account name. The combination of the account name and the Azure Storage service endpoint forms the endpoints for your storage account.

When naming your storage account, keep these rules in mind:

- Storage account names must be between 3 and 24 characters in length and may contain numbers and lowercase letters only.
- Your storage account name must be unique within Azure. No two storage accounts can have the same name. This supports the ability to have a unique, accessible namespace in Azure.

The following table shows the endpoint format for Azure Storage services.

Storage service	Endpoint
Blob Storage	<a href="https://.blob.core.windows.net">https://.blob.core.windows.net</a>
Data Lake Storage Gen2	<a href="https://.dfs.core.windows.net">https://.dfs.core.windows.net</a>
Azure Files	<a href="https://.file.core.windows.net">https://.file.core.windows.net</a>
Queue Storage	<a href="https://.queue.core.windows.net">https://.queue.core.windows.net</a>
Table Storage	<a href="https://.table.core.windows.net">https://.table.core.windows.net</a>

## Describe Azure storage redundancy

Azure Storage always stores multiple copies of your data so that it's protected from planned and unplanned events such as transient hardware failures, network or power outages, and natural disasters. Redundancy ensures that your storage account meets its availability and durability targets even in the face of failures.

When deciding which redundancy option is best for your scenario, consider the tradeoffs between lower costs and higher availability. The factors that help determine which redundancy option you should choose include:

- How your data is replicated in the primary region.
- Whether your data is replicated to a second region that is geographically distant to the primary region, to protect against regional disasters.

- Whether your application requires read access to the replicated data in the secondary region if the primary region becomes unavailable.

## Redundancy in the primary region

Data in an Azure Storage account is always replicated three times in the primary region. Azure Storage offers two options for how your data is replicated in the primary region, locally redundant storage (LRS) and zone-redundant storage (ZRS).

### Locally redundant storage

Locally redundant storage (LRS) replicates your data three times within a single data center in the primary region. LRS provides at least 11 nines of durability (99.999999999%) of objects over a given year.

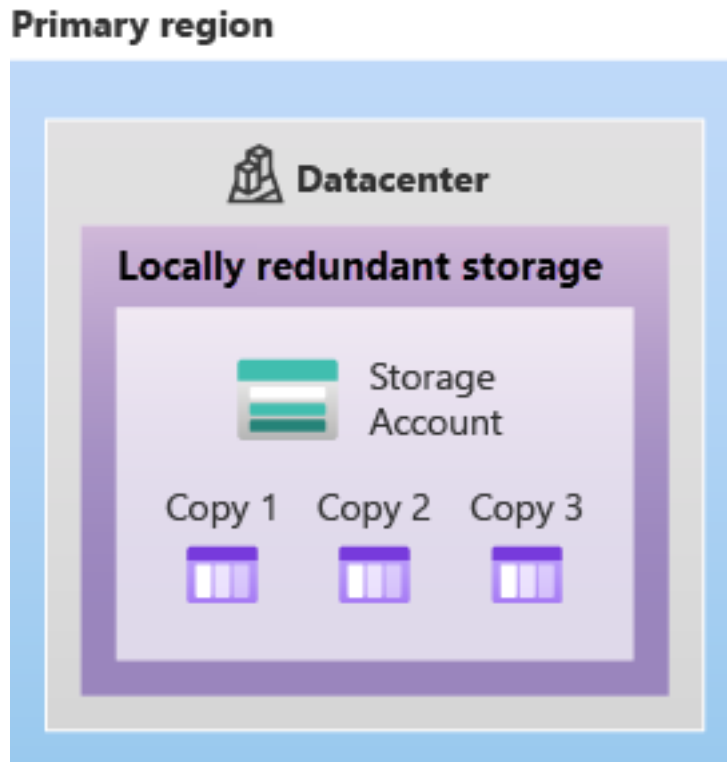


Figure 1: Diagram showing the structure used for locally redundant storage.

LRS is the lowest-cost redundancy option and offers the least durability compared to other options. LRS protects your data against server rack and drive failures. However, if a disaster such as fire or flooding occurs within the data center,

all replicas of a storage account using LRS may be lost or unrecoverable. To mitigate this risk, Microsoft recommends using zone-redundant storage (ZRS), geo-redundant storage (GRS), or geo-zone-redundant storage (GZRS).

### Zone-redundant storage

For Availability Zone-enabled Regions, zone-redundant storage (ZRS) replicates your Azure Storage data synchronously across three Azure availability zones in the primary region. ZRS offers durability for Azure Storage data objects of at least 12 nines (99.999999999%) over a given year.

#### Primary region

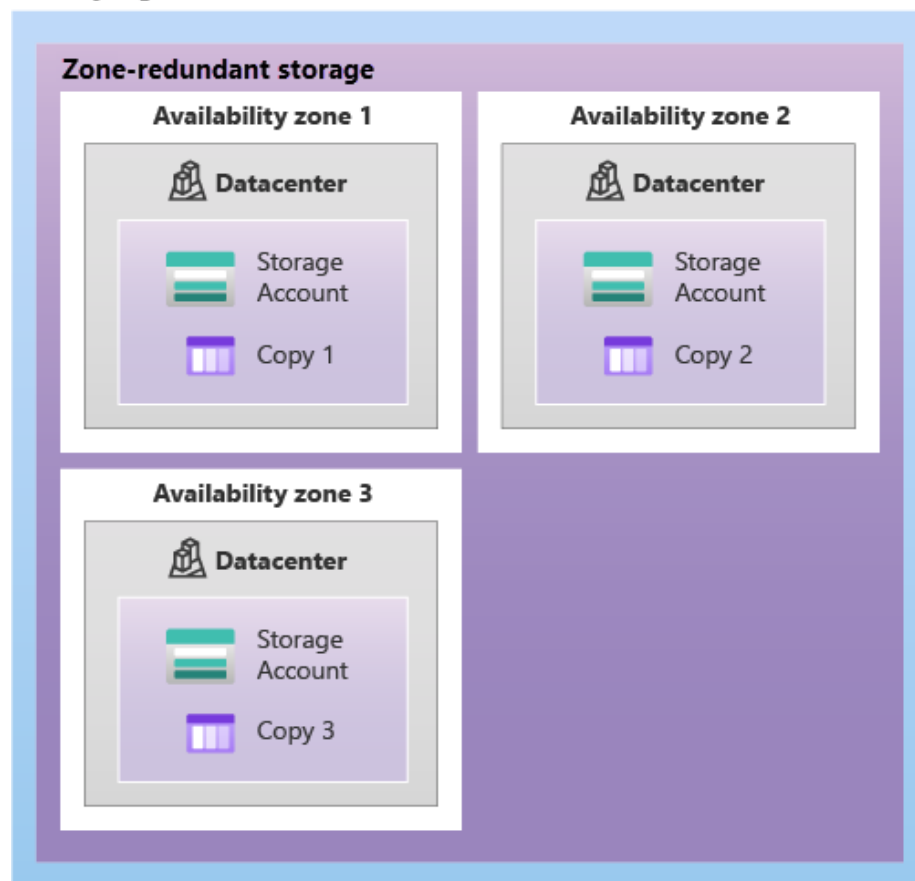


Figure 2: Diagram showing ZRS, with a copy of data stored in each of three availability zones.

With ZRS, your data is still accessible for both read and write operations even if a zone becomes unavailable. No remounting of Azure file shares from the connected clients is required. If a zone becomes unavailable, Azure undertakes

networking updates, such as DNS repointing. These updates may affect your application if you access data before the updates have completed.

Microsoft recommends using ZRS in the primary region for scenarios that require high availability. ZRS is also recommended for restricting replication of data within a country or region to meet data governance requirements.

## **Redundancy in a secondary region**

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 the data in your storage account is copied to a secondary region, then your data is durable even in the event of a catastrophic failure that prevents the data in the primary region from being recovered.

When you create a storage account, you select the primary region for the account. The paired secondary region is based on Azure Region Pairs, and can't be changed.

Azure Storage offers two options for copying your data to a secondary region: geo-redundant storage (GRS) and geo-zone-redundant storage (GZRS). GRS is similar to running LRS in two regions, and GZRS is similar to running ZRS in the primary region and LRS in the secondary region.

By default, data in the secondary region isn't available for read or write access unless there's a failover to the secondary region. If the primary region becomes unavailable, you can choose to fail over to the secondary region. After the failover has completed, the secondary region becomes the primary region, and you can again read and write data.

[!info] Important Because data is replicated to the secondary region asynchronously, a failure that affects the primary region may result in data loss if the primary region can't be recovered. The interval between the most recent writes to the primary region and the last write to the secondary region is known as the recovery point objective (RPO). The RPO indicates the point in time to which data can be recovered. Azure Storage typically has an RPO of less than 15 minutes, although there's currently no SLA on how long it takes to replicate data to the secondary region.

## **Geo-redundant storage**

GRS copies your data synchronously three times within a single physical location in the primary region using LRS. It then copies your data asynchronously to a single physical location in the secondary region (the region pair) using LRS. GRS offers durability for Azure Storage data objects of at least 16 nines (99.99999999999999%) over a given year.

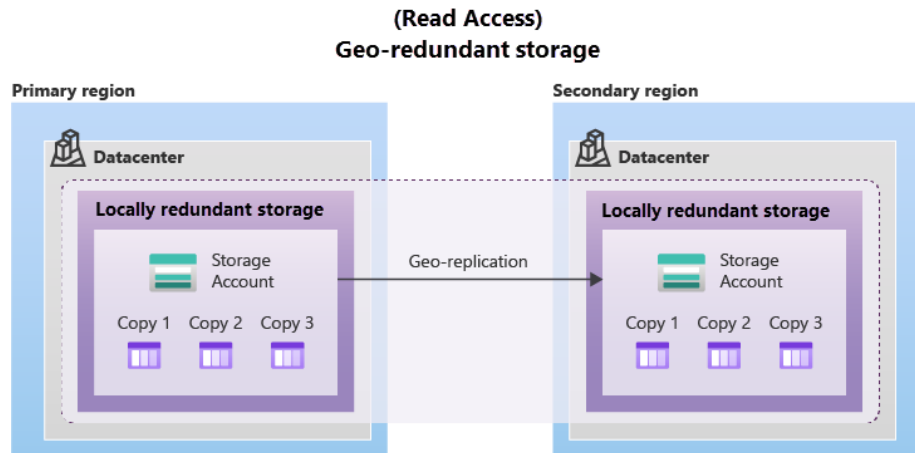


Figure 3: Diagram showing GRS, with primary region LRS replicating data to LRS in a second region.

### Geo-zone-redundant storage

GZRS combines the high availability provided by redundancy across availability zones with protection from regional outages provided by geo-replication. Data in a GZRS storage account is copied across three Azure availability zones in the primary region (similar to ZRS) and is also replicated to a secondary geographic region, using LRS, for protection from regional disasters. Microsoft recommends using GZRS for applications requiring maximum consistency, durability, and availability, excellent performance, and resilience for disaster recovery.

GZRS is designed to provide at least 16 nines (99.99999999999999%) of durability of objects over a given year.

### Read access to data in the secondary region

Geo-redundant storage (with GRS or GZRS) replicates your data to another physical location in the secondary region to protect against regional outages. However, that data is available to be read only if the customer or Microsoft initiates a failover from the primary to secondary region. However, if you enable read access to the secondary region, your data is always available, even when the primary region is running optimally. For read access to the secondary region, enable read-access geo-redundant storage (RA-GRS) or read-access geo-zone-redundant storage (RA-GZRS).

[!info] Important Remember that the data in your secondary region may not be up-to-date due to RPO.

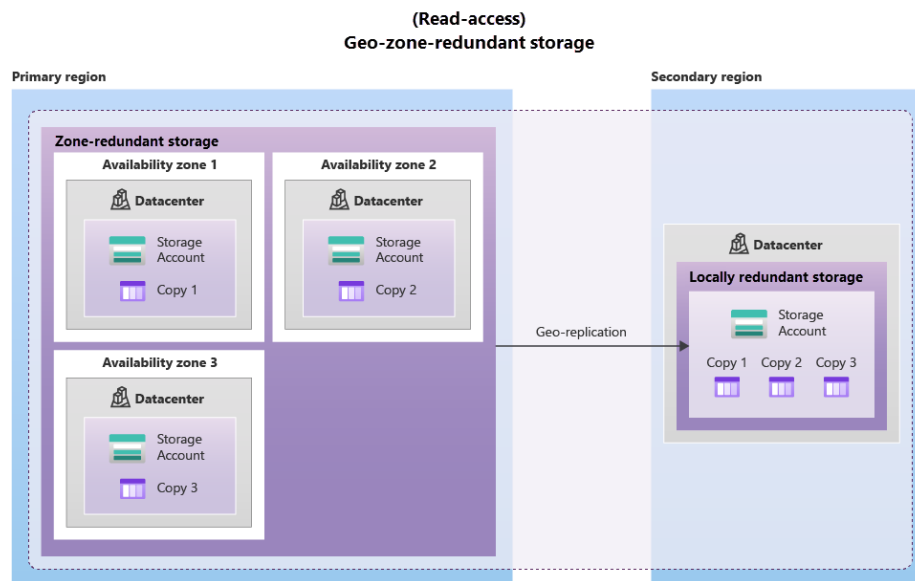


Figure 4: Diagram showing GZRS, with primary region ZRS replicating data to LRS in a second region.

## Describe Azure storage services

The Azure Storage platform includes the following data services:

- **Azure Blobs:** A massively scalable object store for text and binary data. Also includes support for big data analytics through Data Lake Storage Gen2.
- **Azure Files:** Managed file shares for cloud or on-premises deployments.
- **Azure Queues:** A messaging store for reliable messaging between application components.
- **Azure Disks:** Block-level storage volumes for Azure VMs.
- **Azure Tables:** NoSQL table option for structured, non-relational data.

## Benefits of Azure Storage

Azure Storage services offer the following benefits for application developers and IT professionals:

- **Durable and highly available.** Redundancy ensures that your data is safe if transient hardware failures occur. You can also opt to replicate data across data centers or geographical regions for additional protection from local catastrophes or natural disasters. Data replicated in this way remains highly available if an unexpected outage occurs.
- **Secure.** All data written to an Azure storage account is encrypted by the



service. Azure Storage provides you with fine-grained control over who has access to your data.

- **Scalable.** Azure Storage is designed to be massively scalable to meet the data storage and performance needs of today's applications.
- **Managed.** Azure handles hardware maintenance, updates, and critical issues for you.
- **Accessible.** Data in Azure Storage is accessible from anywhere in the world over HTTP or HTTPS. Microsoft provides client libraries for Azure Storage in a variety of languages, including .NET, Java, Node.js, Python, PHP, Ruby, Go, and others, as well as a mature REST API. Azure Storage supports scripting in Azure PowerShell or Azure CLI. And the Azure portal and Azure Storage Explorer offer easy visual solutions for working with your data.

## Azure Blobs

Azure Blob storage is an object storage solution for the cloud. It can store massive amounts of data, such as text or binary data. Azure Blob storage is unstructured, meaning that there are no restrictions on the kinds of data it can hold. Blob storage can manage thousands of simultaneous uploads, massive amounts of video data, constantly growing log files, and can be reached from anywhere with an internet connection.

Blobs aren't limited to common file formats. A blob could contain gigabytes of binary data streamed from a scientific instrument, an encrypted message for another application, or data in a custom format for an app you're developing. One advantage of blob storage over disk storage is that it doesn't require developers to think about or manage disks. Data is uploaded as blobs, and Azure takes care of the physical storage needs.

Blob storage is ideal for:

- Serving images or documents directly to a browser.
- Storing files for distributed access.
- Streaming video and audio.
- Storing data for backup and restore, disaster recovery, and archiving.
- Storing data for analysis by an on-premises or Azure-hosted service.

## Accessing blob storage

Objects in blob storage can be accessed from anywhere in the world via HTTP or HTTPS. Users or client applications can access blobs via URLs, the Azure Storage REST API, Azure PowerShell, Azure CLI, or an Azure Storage client library. The storage client libraries are available for multiple languages, including .NET, Java, Node.js, Python, PHP, and Ruby.

## Blob storage tiers

Data stored in the cloud can grow at an exponential pace. To manage costs for your expanding storage needs, it's helpful to organize your data based on attributes like frequency of access and planned retention period. Data stored in the cloud can be handled differently based on how it's generated, processed, and accessed over its lifetime. Some data is actively accessed and modified throughout its lifetime. Some data is accessed frequently early in its lifetime, with access dropping drastically as the data ages. Some data remains idle in the cloud and is rarely, if ever, accessed after it's stored. To accommodate these different access needs, Azure provides several access tiers, which you can use to balance your storage costs with your access needs.

Azure Storage offers different access tiers for your blob storage, helping you store object data in the most cost-effective manner. The available access tiers include:

- **Hot access tier:** Optimized for storing data that is accessed frequently (for example, images for your website).
- **Cool access tier:** Optimized for data that is infrequently accessed and stored for at least 30 days (for example, invoices for your customers).
- **Cold access tier:** Optimized for storing data that is infrequently accessed and stored for at least 90 days.
- **Archive access tier:** Appropriate for data that is rarely accessed and stored for at least 180 days, with flexible latency requirements (for example, long-term backups).

The following considerations apply to the different access tiers:

- Hot, cool, and cold access tiers can be set at the account level. The archive access tier isn't available at the account level.
- Hot, cool, cold, and archive tiers can be set at the blob level, during or after upload.
- Data in the cool and cold access tiers can tolerate slightly lower availability, but still requires high durability, retrieval latency, and throughput characteristics similar to hot data. For cool and cold data, a lower availability service-level agreement (SLA) and higher access costs compared to hot data are acceptable trade-offs for lower storage costs.
- Archive storage stores data offline and offers the lowest storage costs, but also the highest costs to rehydrate and access data.

## Azure Files

Azure File storage offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) or Network File System (NFS) protocols. Azure Files file shares can be mounted concurrently by cloud or on-premises deployments. SMB Azure file shares are accessible from Windows, Linux, and macOS clients. NFS Azure Files shares are accessible from Linux or macOS clients. Additionally, SMB Azure file shares can be cached on Windows

Servers with Azure File Sync for fast access near where the data is being used.

#### Azure Files key benefits:

- **Shared access:** Azure file shares support the industry standard SMB and NFS protocols, meaning you can seamlessly replace your on-premises file shares with Azure file shares without worrying about application compatibility.
- **Fully managed:** Azure file shares can be created without the need to manage hardware or an OS. This means you don't have to deal with patching the server OS with critical security upgrades or replacing faulty hard disks.
- **Scripting and tooling:** PowerShell cmdlets and Azure CLI can be used to create, mount, and manage Azure file shares as part of the administration of Azure applications. You can create and manage Azure file shares using Azure portal and Azure Storage Explorer.
- **Resiliency:** Azure Files has been built from the ground up to always be available. Replacing on-premises file shares with Azure Files means you don't have to wake up in the middle of the night to deal with local power outages or network issues.
- **Familiar programmability:** Applications running in Azure can access data in the share via file system I/O APIs. Developers can therefore use their existing code and skills to migrate existing applications. In addition to System IO APIs, you can use Azure Storage Client Libraries or the Azure Storage REST API.

#### Azure Queues

Azure Queue storage is a service for storing large numbers of messages. Once stored, you can access the messages from anywhere in the world via authenticated calls using HTTP or HTTPS. A queue can contain as many messages as your storage account has room for (potentially millions). Each individual message can be up to 64 KB in size. Queues are commonly used to create a backlog of work to process asynchronously.

Queue storage can be combined with compute functions like Azure Functions to take an action when a message is received. For example, you want to perform an action after a customer uploads a form to your website. You could have the submit button on the website trigger a message to the Queue storage. Then, you could use Azure Functions to trigger an action once the message was received.

#### Azure Disks

Azure Disk storage, or Azure managed disks, are block-level storage volumes managed by Azure for use with Azure VMs. Conceptually, they're the same as a physical disk, but they're virtualized – offering greater resiliency and availability

than a physical disk. With managed disks, all you have to do is provision the disk, and Azure will take care of the rest.

## Azure Tables

Azure Table storage stores large amounts of structured data. Azure tables are a NoSQL datastore that accepts authenticated calls from inside and outside the Azure cloud. This enables you to use Azure tables to build your hybrid or multicloud solution and have your data always available. Azure tables are ideal for storing structured, non-relational data.

## Identify Azure data migration options

Now that you understand the different storage options within Azure, it's important to also understand how to get your data and information into Azure. Azure supports both real-time migration of infrastructure, applications, and data using Azure Migrate as well as asynchronous migration of data using Azure Data Box.

## Azure Migrate

Azure Migrate is a service that helps you migrate from an on-premises environment to the cloud. Azure Migrate functions as a hub to help you manage the assessment and migration of your on-premises datacenter to Azure. It provides the following:

- **Unified migration platform:** A single portal to start, run, and track your migration to Azure.
- **Range of tools:** A range of tools for assessment and migration. Azure Migrate tools include Azure Migrate: Discovery and assessment and Azure Migrate: Server Migration. Azure Migrate also integrates with other Azure services and tools, and with independent software vendor (ISV) offerings.
- **Assessment and migration:** In the Azure Migrate hub, you can assess and migrate your on-premises infrastructure to Azure.

## Integrated tools

In addition to working with tools from ISVs, the Azure Migrate hub also includes the following tools to help with migration:

- **Azure Migrate: Discovery and assessment.** Discover and assess on-premises servers running on VMware, Hyper-V, and physical servers in preparation for migration to Azure.
- **Azure Migrate: Server Migration.** Migrate VMware VMs, Hyper-V VMs, physical servers, other virtualized servers, and public cloud VMs to Azure.
- **Data Migration Assistant.** Data Migration Assistant is a stand-alone tool to assess SQL Servers. It helps pinpoint potential problems blocking

migration. It identifies unsupported features, new features that can benefit you after migration, and the right path for database migration.

- **Azure Database Migration Service.** Migrate on-premises databases to Azure VMs running SQL Server, Azure SQL Database, or SQL Managed Instances.
- **Azure App Service migration assistant.** Azure App Service migration assistant is a standalone tool to assess on-premises websites for migration to Azure App Service. Use Migration Assistant to migrate .NET and PHP web apps to Azure.
- **Azure Data Box.** Use Azure Data Box products to move large amounts of offline data to Azure.

## Azure Data Box

Azure Data Box is a physical migration service that helps transfer large amounts of data in a quick, inexpensive, and reliable way. The secure data transfer is accelerated by shipping you a proprietary Data Box storage device that has a maximum usable storage capacity of 80 terabytes. The Data Box is transported to and from your datacenter via a regional carrier. A rugged case protects and secures the Data Box from damage during transit.

You can order the Data Box device via the Azure portal to import or export data from Azure. Once the device is received, you can quickly set it up using the local web UI and connect it to your network. Once you're finished transferring the data (either into or out of Azure), simply return the Data Box. If you're transferring data into Azure, the data is automatically uploaded once Microsoft receives the Data Box back. The entire process is tracked end-to-end by the Data Box service in the Azure portal.

### Use cases

Data Box is ideally suited to transfer data sizes larger than 40 TBs in scenarios with no to limited network connectivity. The data movement can be one-time, periodic, or an initial bulk data transfer followed by periodic transfers.

Here are the various scenarios where Data Box can be used to import data to Azure.

- Onetime migration - when a large amount of on-premises data is moved to Azure.
- Moving a media library from offline tapes into Azure to create an online media library.
- Migrating your VM farm, SQL server, and applications to Azure.
- Moving historical data to Azure for in-depth analysis and reporting using HDInsight.
- Initial bulk transfer - when an initial bulk transfer is done using Data Box (seed) followed by incremental transfers over the network.

- Periodic uploads - when large amount of data is generated periodically and needs to be moved to Azure.

Here are the various scenarios where Data Box can be used to export data from Azure.

- Disaster recovery - when a copy of the data from Azure is restored to an on-premises network. In a typical disaster recovery scenario, a large amount of Azure data is exported to a Data Box. Microsoft then ships this Data Box, and the data is restored on your premises in a short time.
- Security requirements - when you need to be able to export data out of Azure due to government or security requirements.
- Migrate back to on-premises or to another cloud service provider - when you want to move all the data back to on-premises, or to another cloud service provider, export data via Data Box to migrate the workloads.

Once the data from your import order is uploaded to Azure, the disks on the device are wiped clean in accordance with NIST 800-88r1 standards. For an export order, the disks are erased once the device reaches the Azure datacenter.

## Identify Azure file movement options

In addition to large scale migration using services like Azure Migrate and Azure Data Box, Azure also has tools designed to help you move or interact with individual files or small file groups. Among those tools are AzCopy, Azure Storage Explorer, and Azure File Sync.

### AzCopy

AzCopy is a command-line utility that you can use to copy blobs or files to or from your storage account. With AzCopy, you can upload files, download files, copy files between storage accounts, and even synchronize files. AzCopy can even be configured to work with other cloud providers to help move files back and forth between clouds.

[!info] Important Synchronizing blobs or files with AzCopy is one-direction synchronization. When you synchronize, you designate the source and destination, and AzCopy will copy files or blobs in that direction. It doesn't synchronize bi-directionally based on timestamps or other metadata.

### Azure Storage Explorer

Azure Storage Explorer is a standalone app that provides a graphical interface to manage files and blobs in your Azure Storage Account. It works on Windows, macOS, and Linux operating systems and uses AzCopy on the backend to

perform all of the file and blob management tasks. With Storage Explorer, you can upload to Azure, download from Azure, or move between storage accounts.

## **Azure File Sync**

Azure File Sync is a tool that lets you centralize your file shares in Azure Files and keep the flexibility, performance, and compatibility of a Windows file server. It's almost like turning your Windows file server into a miniature content delivery network. Once you install Azure File Sync on your local Windows server, it will automatically stay bi-directionally synced with your files in Azure.

With Azure File Sync, you can:

- Use any protocol that's available on Windows Server to access your data locally, including SMB, NFS, and FTPS.
- Have as many caches as you need across the world.
- Replace a failed local server by installing Azure File Sync on a new server in the same datacenter.
- Configure cloud tiering so the most frequently accessed files are replicated locally, while infrequently accessed files are kept in the cloud until requested.