Azure Blob storage is Microsoft's object storage solution for the cloud. Blob storage is optimized for storing massive amounts of unstructured data. Unstructured data is data that doesn't adhere to a particular data model or definition, such as text or binary data.

**About Blob storage**

Blob storage is designed for:

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

Azure Data Lake Storage Gen2 offers a hierarchical file system as well as the advantages of Blob storage, including:

* Low-cost, tiered storage
* High availability
* Strong consistency
* Disaster recovery capabilities

The Azure Storage platform includes the following data services:

* [Azure Blobs](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blobs-introduction): A massively scalable object store for text and binary data. Also includes support for big data analytics through Data Lake Storage Gen2.
* [Azure Files](https://docs.microsoft.com/en-us/azure/storage/files/storage-files-introduction): Managed file shares for cloud or on-premises deployments.
* [Azure Queues](https://docs.microsoft.com/en-us/azure/storage/queues/storage-queues-introduction): A messaging store for reliable messaging between application components.
* [Azure Tables](https://docs.microsoft.com/en-us/azure/storage/tables/table-storage-overview): A NoSQL store for schemaless storage of structured data.
* [Azure Disks](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/managed-disks-overview): Block-level storage volumes for Azure VMs.

**Blob storage**

Azure Blob storage is Microsoft's object storage solution for the cloud. Blob storage is optimized for storing massive amounts of unstructured data, such as text or binary data.

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.

Objects in Blob storage can be accessed from anywhere in the world via HTTP or HTTPS.

# Introduction to Azure Blob storage

* 06/24/2020
* 4 minutes to read
  + [[https://github.com/tamram.png?size=32](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md)](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md" \o "12 Contributors)

* + [[https://github.com/CarlRabeler.png?size=32](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md)](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md" \o "12 Contributors)

* + [[https://github.com/bandersmsft.png?size=32](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md)](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md" \o "12 Contributors)

* + [[https://github.com/SyntaxC4.png?size=32](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md)](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md" \o "12 Contributors)

* + [[https://github.com/alkohli.png?size=32](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md)](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md" \o "12 Contributors)

* + [+7](https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blobs-introduction.md" \o "12 Contributors)

Azure Blob storage is Microsoft's object storage solution for the cloud. Blob storage is optimized for storing massive amounts of unstructured data. Unstructured data is data that doesn't adhere to a particular data model or definition, such as text or binary data.

## About Blob storage

Blob storage is designed for:

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

**Move data to Blob storage**

A number of solutions exist for migrating existing data to Blob storage:

* **AzCopy** is an easy-to-use command-line tool for Windows and Linux that copies data to and from Blob storage, across containers, or across storage accounts. For more information about AzCopy, see [Transfer data with the AzCopy v10](https://docs.microsoft.com/en-us/azure/storage/common/storage-use-azcopy-v10).
* The **Azure Storage Data Movement library** is a .NET library for moving data between Azure Storage services. The AzCopy utility is built with the Data Movement library. For more information, see the [reference documentation](https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.storage.datamovement) for the Data Movement library.
* **Azure Data Factory** supports copying data to and from Blob storage by using the account key, a shared access signature, a service principal, or managed identities for Azure resources. For more information, see [Copy data to or from Azure Blob storage by using Azure Data Factory](https://docs.microsoft.com/en-us/azure/data-factory/connector-azure-blob-storage?toc=/azure/storage/blobs/toc.json).
* **Blobfuse** is a virtual file system driver for Azure Blob storage. You can use blobfuse to access your existing block blob data in your Storage account through the Linux file system. For more information, see [How to mount Blob storage as a file system with blobfuse](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-how-to-mount-container-linux).
* **Azure Data Box** service is available to transfer on-premises data to Blob storage when large datasets or network constraints make uploading data over the wire unrealistic. Depending on your data size, you can request [Azure Data Box Disk](https://docs.microsoft.com/en-us/azure/databox/data-box-disk-overview), [Azure Data Box](https://docs.microsoft.com/en-us/azure/databox/data-box-overview), or [Azure Data Box Heavy](https://docs.microsoft.com/en-us/azure/databox/data-box-heavy-overview) devices from Microsoft. You can then copy your data to those devices and ship them back to Microsoft to be uploaded into Blob storage.
* The **Azure Import/Export service** provides a way to import or export large amounts of data to and from your storage account using hard drives that you provide. For more information, see [Use the Microsoft Azure Import/Export service to transfer data to Blob storage](https://docs.microsoft.com/en-us/azure/storage/common/storage-import-export-service).

**Types of storage accounts**

Azure Storage offers several types of storage accounts. Each type supports different features and has its own pricing model. Consider these differences before you create a storage account to determine the type of account that is best for your applications. The types of storage accounts are:

* **General-purpose v2 accounts**: Basic storage account type for blobs, files, queues, and tables. Recommended for most scenarios using Azure Storage.
* **General-purpose v1 accounts**: Legacy account type for blobs, files, queues, and tables. Use general-purpose v2 accounts instead when possible.
* **BlockBlobStorage accounts**: Storage accounts with premium performance characteristics for block blobs and append blobs. Recommended for scenarios with high transactions rates, or scenarios that use smaller objects or require consistently low storage latency.
* **FileStorage accounts**: Files-only storage accounts with premium performance characteristics. Recommended for enterprise or high performance scale applications.
* **BlobStorage accounts**: Legacy Blob-only storage accounts. Use general-purpose v2 accounts instead when possible.

Azure Key Vault helps solve the following problems:

* **Secrets Management** - Azure Key Vault can be used to Securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets
* **Key Management** - Azure Key Vault can also be used as a Key Management solution. Azure Key Vault makes it easy to create and control the encryption keys used to encrypt your data.
* **Certificate Management** - Azure Key Vault is also a service that lets you easily provision, manage, and deploy public and private Transport Layer Security/Secure Sockets Layer (TLS/SSL) certificates for use with Azure and your internal connected resources.
* **Store secrets backed by Hardware Security Modules** - The secrets and keys can be protected either by software or FIPS 140-2 Level 2 validated HSMs

### Purge protection

Purge protection is an optional Key Vault behavior and is **not enabled by default**. Purge protection can only be enabled once soft-delete is enabled. It can be turned on via [CLI](https://docs.microsoft.com/en-us/azure/key-vault/general/soft-delete-cli#enabling-purge-protection) or [PowerShell](https://docs.microsoft.com/en-us/azure/key-vault/general/soft-delete-powershell#enabling-purge-protection).

When purge protection is on, a vault or an object in the deleted state cannot be purged until the retention period has passed. Soft-deleted vaults and objects can still be recovered, ensuring that the retention policy will be followed.

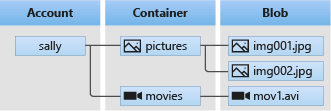
The default retention period is 90 days, but it is possible to set the retention policy interval to a value from 7 to 90 days through the Azure portal. Once the retention policy interval is set and saved it cannot be changed for that vault.

**Blob storage resources**

Blob storage offers three types of resources:

* The **storage account**
* A **container** in the storage account
* A **blob** in a container

The following diagram shows the relationship between these resources.



### Blobs

Azure Storage supports three types of blobs:

* **Block blobs** store text and binary data. Block blobs are made up of blocks of data that can be managed individually. Block blobs store up to about 4.75 TiB of data. Larger block blobs are available in preview, up to about 190.7 TiB
* **Append blobs** are made up of blocks like block blobs, but are optimized for append operations. Append blobs are ideal for scenarios such as logging data from virtual machines.
* **Page blobs** store random access files up to 8 TB in size. Page blobs store virtual hard drive (VHD) files and serve as disks for Azure virtual machines. For more information about page blobs, see [Overview of Azure page blobs](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-pageblob-overview)