Choosing a big data storage technology in Azure

02/12/2018 • 7 minutes to read • Contributors all

In this article

This topic compares options for data storage for big data solutions — specifically, data storage for bulk data ingestion and batch processing, as opposed to analytical data stores or real-time streaming ingestion.

What are your options when choosing data storage in Azure?

There are several options for ingesting data into Azure, depending on your needs.

File storage:

- Azure Storage blobs
- Azure Data Lake Store

NoSQL databases:

- Azure Cosmos DB
- HBase on HDInsight

Azure Storage blobs

Azure Storage is a managed storage service that is highly available, secure, durable, scalable, and redundant. Microsoft takes care of maintenance and handles critical problems for you. Azure Storage is the most ubiquitous storage solution Azure provides, due to the number of services and tools that can be used with it.

There are various Azure Storage services you can use to store data. The most flexible option for storing blobs from a number of data sources is Blob storage. Blobs are basically files. They store pictures, documents, HTML files, virtual hard disks (VHDs), big data such as logs, database backups — pretty much anything. Blobs are stored in containers, which are similar to folders. A container provides a grouping of a set of blobs. A storage account can contain an unlimited number of containers, and a container can store an unlimited number of blobs.

Azure Storage is a good choice for big data and analytics solutions, because of its flexibility, high availability, and low cost. It provides hot, cool, and archive storage tiers for different use cases. For more information, see Azure Blob Storage: Hot, cool, and archive storage tiers.

Azure Blob storage can be accessed from Hadoop (available through HDInsight). HDInsight can use a blob container in Azure Storage as the default file system for the cluster. Through a Hadoop distributed file system (HDFS) interface provided by a WASB driver, the full set of components in HDInsight can operate directly on structured or unstructured data stored as blobs. Azure Blob storage can also be accessed via Azure SQL Data Warehouse using its PolyBase feature.

Other features that make Azure Storage a good choice are:

- Multiple concurrency strategies.
- Disaster recovery and high availability options.
- Encryption at rest.
- Role-based access control (RBAC) to control access using Azure Active Directory users and groups.

Azure Data Lake Store

Azure Data Lake Store is an enterprise-wide hyper-scale repository for big data analytic workloads. Data Lake enables you to capture data of any size, type, and ingestion speed in one single secure location for operational and exploratory analytics.

Data Lake Store does not impose any limits on account sizes, file sizes, or the amount of data that can be stored in a data lake. Data is stored durably by making multiple copies and there is no limit on the duration of time that the data can be stored in the Data Lake. In addition to making multiple copies of files to guard against any unexpected failures, Data lake spreads parts of a file over a number of individual storage servers. This improves the read throughput when reading the file in parallel for performing data analytics.

Data Lake Store can be accessed from Hadoop (available through HDInsight) using the WebHDFS-compatible REST APIs. You may consider using this as an alternative to Azure Storage when your individual or combined file sizes exceed that which is supported by Azure Storage. However, there are performance tuning guidelines you should follow when using Data Lake Store as your primary storage for an HDInsight cluster, with specific guidelines for Spark, Hive, MapReduce, and Storm. Also, be sure to check Data Lake Store's regional availability, because it is not available in as many regions as Azure Storage, and it needs to be located in the same region as your HDInsight cluster.

Coupled with Azure Data Lake Analytics, Data Lake Store is specifically designed to enable analytics on the stored data and is tuned for performance for data analytics scenarios. Data Lake Store can also be accessed via Azure SQL Data Warehouse using its PolyBase feature.

Azure Cosmos DB

Azure Cosmos DB is Microsoft’s globally distributed multi-model database. Cosmos DB guarantees single-digit-millisecond latencies at the 99th percentile anywhere in the world, offers multiple well-defined consistency models to fine-tune performance, and guarantees high availability with multi-homing capabilities.

Azure Cosmos DB is schema-agnostic. It automatically indexes all the data without requiring you to deal with schema and index management. It’s also multi-model, natively supporting document, key-value, graph, and column-family data models.

Azure Cosmos DB features:

- Geo-replication
- Elastic scaling of throughput and storage worldwide
- Five well-defined consistency levels

HBase on HDInsight

Apache HBase is an open-source, NoSQL database that is built on Hadoop and modeled after Google BigTable. HBase provides random access and strong consistency for large amounts of unstructured and semi-structured data in a schemaless database organized by column families.

Data is stored in the rows of a table, and data within a row is grouped by column family. HBase is schemaless in the sense that neither the columns nor the type of data stored in them need to be defined before using them. The open-source code scales linearly to handle petabytes of data on thousands of nodes. It can rely on data redundancy, batch processing, and other features that are provided by distributed applications in the Hadoop ecosystem.

The HDInsight implementation leverages the scale-out architecture of HBase to provide automatic sharding of tables, strong consistency for reads and writes, and automatic failover. Performance is enhanced by in-memory caching for reads and high-throughput streaming for writes. In most cases, you'll want to create the HBase cluster inside a virtual network so other HDInsight clusters and applications can directly access the tables.

Key selection criteria

To narrow the choices, start by answering these questions:

- Do you need managed, high speed, cloud-based storage for any type of text or binary data? If yes, then select one of the file storage options.
- Do you need file storage that is optimized for parallel analytics workloads and high throughput/IOPS? If yes, then choose an option that is tuned to analytics workload performance.
- Do you need to store unstructured or semi-structured data in a schemaless database? If so, select one of the non-relational options. Compare options for indexing and database models. Depending on the type of data you need to store, the primary database models may be the largest factor.
- Can you use the service in your region? Check the regional availability for each Azure service. See Products available by region.

Capability matrix

The following tables summarize the key differences in capabilities.

File storage capabilities

Capability	Azure Data Lake Store	Azure Blob Storage containers
Purpose	Optimized storage for big data analytics workloads	General purpose object store for a wide variety of storage scenarios
Use cases	Batch, streaming analytics, and machine learning data such as log files, IoT data, click streams, large datasets	Any type of text or binary data, such as application back end, backup data, media storage for streaming, and general purpose data
Structure	Hierarchical file system	Object store with flat namespace
Authentication	Based on Azure Active Directory Identities	Based on shared secrets Account Access Keys and Shared Access Signature Keys, and role-based access control (RBAC)
Authentication protocol	OAuth 2.0. Calls must contain a valid JWT (JSON web token) issued by Azure Active Directory	Hash-based message authentication code (HMAC). Calls must contain a Base64-encoded SHA-256 hash over a part of the HTTP request.
Authorization	POSIX access control lists (ACLs). ACLs based on Azure Active Directory identities can be set file and folder level.	For account-level authorization use Account Access Keys. For account, container, or blob authorization use Shared Access Signature Keys.

Capability	Azure Data Lake Store	Azure Blob Storage containers
Auditing	Available.	Available
Encryption at rest	Transparent, server side	Transparent, server side; Client-side encryption
Developer SDKs	.NET, Java, Python, Node.js	.Net, Java, Python, Node.js, C++, Ruby
Analytics workload performance	Optimized performance for parallel analytics workloads, High Throughput and IOPS	Not optimized for analytics workloads
Size limits	No limits on account sizes, file sizes or number of files	Specific limits documented here
Geo-redundancy	Locally-redundant (multiple copies of data in one Azure region)	Locally redundant (LRS), globally redundant (GRS), read-access globally redundant (RA-GRS). See here for more information

NoSQL database capabilities

Capability	Azure Cosmos DB	HBase on HDInsight
Primary database model	Document store, graph, key-value store, wide column store	Wide column store
Secondary indexes	Yes	No
SQL language support	Yes	Yes (using the Phoenix JDBC driver)
Consistency	Strong, bounded-staleness, session, consistent prefix, eventual	Strong
Native Azure Functions integration	Yes	No
Automatic global distribution	Yes	No HBase cluster replication can be configured across regions with eventual consistency
Pricing model	Elastically scalable request units (RUs) charged per-second as needed, elastically scalable storage	Per-minute pricing for HDInsight cluster (horizontal scaling of nodes), storage