**What is Azure Storage?**

Azure Storage is Microsoft’s **cloud storage solution** that provides **highly available, durable, secure, and scalable storage** for different types of data: unstructured, semi-structured, and structured.

It’s designed for:

* Applications (web, mobile, enterprise)
* Backup and restore
* File sharing
* Big data & analytics
* Streaming content

All Azure Storage data is:

* **Replicated** (locally or across regions)
* **Durable** (survives hardware failures)
* **Secure** (encrypted at rest & in transit)
* **Scalable** (handles petabytes of data)

**🔹 Types of Azure Storage Services**

**1. Blob Storage (Unstructured Data)**

* **Use case**: Images, videos, audio, documents, backups, logs, big data.
* Stores massive amounts of **unstructured data** (no schema, just files).
* Organized in:
  + **Storage Account** → **Container** → **Big large object (file)**
* Types of blobs:
  + **Block blob** → Text & binary files (documents, media files)
  + **Append blob** → Log files (data is always appended)
  + **Page blob** → Virtual hard drives (for Azure VMs)

👉 **Container** in Blob Storage = A logical grouping of blobs (like a folder).

**2. File Storage (SMB File Shares)**

* **Use case**: Lift-and-shift applications, file sharing across VMs, replacing on-premises file servers.
* Provides **fully managed cloud file shares** using **SMB (Server Message Block) protocol**.
* Can be mounted concurrently by multiple VMs.
* Access via:
  + **SMB** (Windows, Linux, MacOS mounting)
  + **REST API**
* Supports **Azure File Sync** → caches frequently used files on-prem while storing everything in the cloud.

👉 **File Share** = A shared network folder in the cloud.

**3. Table Storage (NoSQL Key-Value Store)**

* **Use case**: Fast access to structured, non-relational data.
* Schema-less design → flexible for apps needing scalable key-value storage.
* Example: Storing IoT sensor data, logs, user profiles.
* Access via REST or SDK.

👉 Now part of **Azure Cosmos DB Table API**.

**4. Queue Storage (Messaging)**

* **Use case**: Decoupling applications, background processing, task scheduling.
* Stores **messages** (up to 64 KB each).
* A single queue can hold **millions of messages**.
* Used for **asynchronous communication** between components (Producer → Queue → Consumer).

**5. Disk Storage (Persistent Disks for VMs)**

* **Use case**: Virtual Machine OS & Data disks.
* Types:
  + **Standard HDD** – Low-cost, dev/test workloads.
  + **Standard SSD** – Balance of cost & performance.
  + **Premium SSD** – High performance, low latency.
  + **Ultra Disk** – Mission-critical apps needing high throughput.

**🔹 Quick Comparison**

| **Storage Type** | **Structure** | **Protocol / Access** | **Best For** |
| --- | --- | --- | --- |
| **Blob (Container)** | Container → Blob | REST API, SDKs, HTTPS | Unstructured data (media, backups, logs) |
| **File Share** | File share → Files | SMB, REST API | File server replacement, app sharing |
| **Table** | Table → Entities | OData/REST, SDK | NoSQL key-value, scalable structured storage |
| **Queue** | Queue → Messages | REST API, SDK | Decoupled async messaging |
| **Disk** | Managed Disks | Attach to VM | OS & data disks for VMs |

✅ **In short:**

* **Blob (Container)** → Cloud object storage for any type of file.
* **File Share** → Cloud file server, mountable like a network drive.
* **Table** → NoSQL storage for structured data.
* **Queue** → Messaging between distributed components.
* **Disk** → VM storage like virtual hard drives.

Azure offers a wide range of **Database-as-a-Service (DBaaS)** options, covering **relational, NoSQL, in-memory, and big data** needs.

**🔹 Why Azure Databases?**

* **Fully managed** → Microsoft handles patching, backups, scaling, and high availability.
* **Secure** → Built-in encryption, firewalls, VNET integration, threat detection.
* **Scalable** → Handle small apps to enterprise-scale workloads.
* **Multi-model** → Relational, NoSQL, graph, time-series, etc.
* **Global availability** → Data can be replicated across regions.

**🔹 Azure Database Services**

**1. Azure SQL Database**

* **Relational database** based on Microsoft SQL Server.
* PaaS (Platform as a Service) → No need to manage servers.
* Features:
  + High availability and backups.
  + Built-in intelligence (auto-tuning, performance recommendations).
  + Hyperscale option → grow up to 100 TB.
* Use cases: Traditional business apps, web apps, ERP/CRM systems.

👉 Think of it as **SQL Server in the cloud without server management**.

**2. Azure Cosmos DB**

* **Globally distributed, multi-model NoSQL database**.
* Supports multiple APIs:
  + SQL API (document model)
  + MongoDB API
  + Cassandra API
  + Gremlin API (graph data)
  + Table API (key-value store)
* Features:
  + Single-digit millisecond latency.
  + Global replication with multi-master writes.
  + Elastic scaling of throughput (RU/s).
* Use cases: IoT, real-time analytics, gaming leaderboards, e-commerce catalogs.

👉 **Cosmos DB = NoSQL + Global distribution + Multi-model flexibility**.

**3. Azure Database for MySQL**

* Fully managed MySQL community edition in the cloud.
* Features:
  + Automatic patching, backups, scaling.
  + High availability.
  + Flexible server deployment model.
* Use cases: Open-source apps (WordPress, Drupal, Magento), LAMP stack apps.

**4. Azure Database for PostgreSQL**

* Fully managed PostgreSQL in the cloud.
* Deployment options:
  + **Single Server** (basic workloads).
  + **Flexible Server** (customized, high availability, zone redundancy).
  + **Hyperscale (Citus)** → Distributed PostgreSQL for massive scalability.
* Use cases: Open-source and enterprise apps, analytics, GIS apps.

**5. Azure Database for MariaDB**

* Fully managed MariaDB (community edition).
* Features:
  + High availability.
  + Automatic scaling.
* Use cases: Applications already using MariaDB, open-source web apps.

**6. Azure Synapse Analytics (formerly SQL Data Warehouse)**

* **Big data + data warehouse** service.
* Combines **enterprise data warehousing** with **big data analytics**.
* Supports querying data from multiple sources (SQL, Spark, Data Lake).
* Use cases: Business intelligence (BI), advanced analytics, reporting dashboards.

**7. Azure Cache for Redis**

* **In-memory data store**.
* Fully managed Redis (open-source).
* Features:
  + Extremely fast (sub-millisecond response).
  + Supports session caching, pub/sub messaging.
* Use cases: Caching, gaming leaderboards, chat/messaging apps.

**8. Azure SQL Managed Instance**

* Managed SQL Server instance in the cloud.
* Offers **almost 100% compatibility** with on-premises SQL Server.
* Supports cross-database queries, SQL Agent, Linked Servers.
* Use cases: Lift-and-shift SQL Server workloads to the cloud.

**9. Other Specialized Databases**

* **Azure Data Explorer (ADX)** → Time-series + log analytics at scale.
* **Azure Database Migration Service** → Migrate databases to Azure with minimal downtime.

**🔹 Quick Comparison**

| **Service** | **Type** | **Best For** |
| --- | --- | --- |
| **Azure SQL Database** | Relational (SQL Server) | Web apps, enterprise apps |
| **SQL Managed Instance** | Relational (SQL Server full features) | Lift & shift SQL workloads |
| **Cosmos DB** | NoSQL, Multi-model | IoT, real-time analytics, global apps |
| **MySQL** | Relational | Open-source apps, LAMP stack |
| **PostgreSQL** | Relational | Enterprise, GIS, analytics |
| **MariaDB** | Relational | Open-source workloads |

**1. Azure Pricing Basics**

Azure uses a **pay-as-you-go** model:

* You **pay only for what you use** (no upfront cost).
* Charges depend on:
  + Service type (VM, storage, database, etc.)
  + Usage (compute hours, storage size, transactions, bandwidth).
  + Region (pricing differs per Azure region).
  + Tier/SKU (basic, standard, premium).
  + Licensing (bring your own license for SQL/Windows, or buy through Azure).

💡 Example:

* A Virtual Machine cost = VM size (CPU/RAM) + OS license + storage (disk) + networking (bandwidth).

**🔹 2. Azure Pricing Models**

1. **Pay-As-You-Go (PAYG)**
   * No upfront commitment.
   * Good for startups, dev/test, unpredictable workloads.
2. **Reserved Instances (RI)**
   * Commit for **1 or 3 years** for services like VMs, SQL Databases.
   * Saves **up to 72%** compared to PAYG.
   * Best for predictable workloads.
3. **Spot Pricing**
   * Uses **unused Azure capacity** at big discounts.
   * Risk: VM may be evicted when capacity is needed.
   * Best for batch jobs, non-critical tasks.
4. **Azure Hybrid Benefit**
   * Bring your own licenses (Windows Server, SQL Server).
   * Saves up to 40%.
5. **Dev/Test Pricing**
   * Special discounted rates for non-production environments.

**🔹 3. Azure Cost Management Tools**

Azure provides built-in tools to track and optimize costs:

**✅ Azure Pricing Calculator**

* Website: Azure Pricing Calculator
* Estimate monthly costs before deploying.
* You select services (VMs, Storage, Databases), and it shows costs.

**✅ Azure Total Cost of Ownership (TCO) Calculator**

* Compares **on-premises costs vs Azure costs**.
* Useful for migrations.

**✅ Azure Cost Management + Billing (built-in portal service)**

* Free service inside Azure Portal.
* Features:
  + View **cost analysis** (per service, per resource, per subscription).
  + Set **budgets** → alerts when nearing limits.
  + Assign **tags** → track costs by department/project.
  + Optimize with **recommendations** (e.g., unused VMs, underutilized resources).

**✅ Azure Advisor**

* Gives recommendations to reduce cost:
  + Shut down idle VMs.
  + Resize underutilized resources.
  + Use Reserved Instances.
  + Optimize storage tiers.

**🔹 4. Cost Optimization Strategies**

1. **Right-size resources** → Use proper VM size, avoid overprovisioning.
2. **Auto-scaling** → Scale up/down based on demand.
3. **Use Reserved Instances or Savings Plans** for predictable workloads.
4. **Spot Instances** for non-critical workloads.
5. **Storage Tiers**:
   * Hot (frequent access)
   * Cool (infrequent)
   * Archive (rare, cheapest).
6. **Shut down unused resources** → Schedule off-hours shutdown.
7. **Use Azure Hybrid Benefit** if you already have licenses.
8. **Tagging & Budgets** → Track by team/project.

**🔹 5. Example of Cost Breakdown (Virtual Machine)**

Let’s say you run a VM in **East US region**:

* VM type: Standard\_DS2\_v2 (2 vCPU, 7GB RAM) → ~$75/month (Linux), ~$100/month (Windows).
* Managed Disk (128 GB SSD) → ~$19/month.
* Bandwidth (egress) → First 5GB free, then ~$0.087/GB.
* Backup → Extra cost.

👉 Total ~ $120–150/month depending on usage.

**🔹 6. Key Terms**

* **Subscription** → Billing container (your Azure account).
* **Resource Group** → Logical grouping of resources (can track costs here).
* **Tags** → Metadata (e.g., Department=Finance) → helps split costs.
* **Billing Account** → Where payments are processed.

✅ **In short:**

* Azure pricing = pay for what you use (compute, storage, bandwidth).
* Pricing models = Pay-as-you-go, Reserved, Spot, Hybrid Benefit.
* Tools = Pricing Calculator, TCO Calculator, Azure Cost Management + Billing, Advisor.
* Optimize with auto-scaling, right-sizing, reserved instances, and storage tiers.

Top of Form

Bottom of Form