**AZURE SYNAPSE ANALYTICS**

Azure Synapse Analytics is a cloud-based big data and analytics service from Microsoft, combining enterprise data warehousing and big data analytics into a single platform. It enables efficient querying and analysis of both structured and unstructured data using SQL, Spark, and other analytics tools.

### Why Use Azure Synapse?

### Azure Synapse is a limitless analytics service that merges enterprise data warehousing with big data analytics. It allows flexible querying of data on either serverless or dedicated resources at scale.

### Key Components of Azure Synapse Analytics:

**Unified Analytics Platform -**   
Combines SQL-based data warehousing and big data analytics.

**Synapse SQL**

* **Serverless SQL Pool**: Pay-per-query model for on-demand analysis.
* **Dedicated SQL Pool**: Provisioned capacity for large-scale data processing.

**Spark Integration -**   
Supports Apache Spark for big data processing and machine learning.

**Data Integration -**   
Built-in Azure Data Factory (ADF) for ETL (Extract, Transform, Load).

**Synapse Pipelines -**   
Automates data movement and transformation.

**Synapse Studio -**   
A web-based workspace for managing data, writing SQL queries, and creating reports.

**When to Use Which?**

**Use Dedicated SQL Pool When:**

* You have a consistent workload requiring high performance.
* You need optimized performance for large-scale ETL processing.
* You want full control over resources and predictable pricing.

**Use Serverless SQL Pool When**:

* You work with ad-hoc queries and need flexibility.
* You need cost-efficient querying of raw data in Azure Data Lake.
* You don’t want to manage infrastructure and just need quick insights.

A screenshot of a computer

AI-generated content may be incorrect.

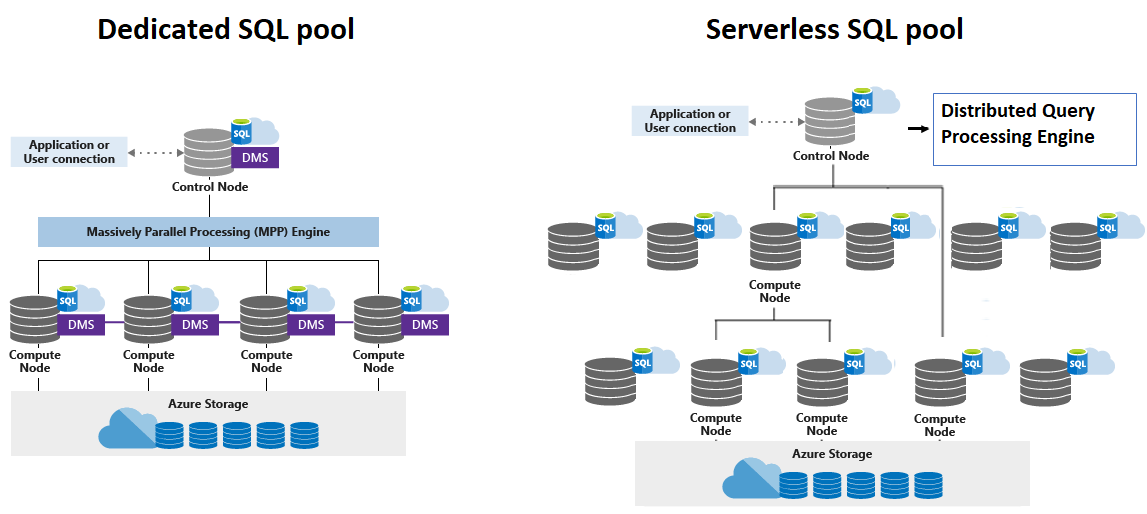
**Control Node** –

Control node is the brain of the architecture. In Synapse SQL, the distributed query engine runs on the Control node to optimize and coordinate parallel queries. When you submit a T-SQL query to dedicated SQL pool, the Control node transforms it into queries that run against each distribution in parallel.

In serverless SQL pool, the DQP engine runs on Control node to optimize and coordinate distributed execution of user query by splitting it into smaller queries that will be executed on Compute nodes. It also assigns sets of files to be processed by each node.

**Compute Nodes** –

The Compute nodes provide the computational power. In a dedicated SQL pool, distribute maps to Compute nodes for processing. As you pay for more compute resources, pool remaps the distributions to the available Compute nodes. The number of computer nodes ranges from 1 to 60 and is determined by the service level for the dedicated SQL pool. Each Compute node has a node ID that is visible in system views.



**Dedicated SQL Pool:**

A Dedicated SQL Pool (formerly known as SQL Data Warehouse) in Azure Synapse Analytics is a fully managed, cloud-based data warehouse service designed to handle large-scale data storage and complex analytical queries. It provides high-performance querying, data loading, and analytics on large datasets by scaling compute resources.

***Distributed Tables:***

Table where records are distributed among compute nodes. There are 3 different distributions Strategies we have in Dedicated SQL Pool.

**Hash Distribution**:

Large Table (Fact Table).

Consider using a hash-distributed table when:

* The table size on disk is more than 2 GB.
* The table has frequent insert, update, and delete operations.

Diagram of a computer server

AI-generated content may be incorrect.

**Round-Robin:**

A round-robin distributed table distributes table rows evenly across all distributions. The assignment of rows to distributions is random. Unlike hash-distributed tables, rows with equal values are not guaranteed to be assigned to the same distribution.

It is used for stagging tables. When you do not want to query the data and want to write the data as quickly as you can.

Consider using the round-robin distribution for your table in the following scenarios:

* When getting started as a simple starting point since it is the default
* If there is no obvious joining key
* If there is no good candidate column for hash distributing the table
* If the table does not share a common join key with other tables
* If the join is less significant than other joins in the query
* When the table is a temporary staging table

**Replicated Distribution:**

We use replicated when we are working with small dimensions. We will save Copy of the data in all the distributions which is called replicated. That copy of the will get replicated to all the distribution.

**Key Features of Dedicated SQL Pool:**

1. **Massive Parallel Processing (MPP)**:

- It distributes the workload across multiple nodes, which enables fast query performance for large datasets.

1. **Scalable**:

- You can scale the compute resources up or down based on your workload requirements. This flexibility ensures optimal performance and cost efficiency.

1. **Data Distribution**:

- Data is distributed across multiple storage nodes to ensure fast data retrieval. You can control how data is distributed, such as using round-robin, hash, or replicated distribution methods.

1. **Security**:

- It supports built-in security features such as firewall rules, data encryption at rest, and auditing. It also integrates with Azure Active Directory (AAD) for identity management.

1. **Integration with Azure Synapse Studio**:

- Dedicated SQL pools are tightly integrated with Azure Synapse Studio, providing a user-friendly environment to manage data pipelines, queries, and analysis.

1. **T-SQL Compatibility**:

- It supports T-SQL, enabling developers and data analysts familiar with SQL Server to easily query data, perform analytics, and use existing SQL-based tools.

1. **Workload Management**:

- Dedicated SQL pools allow you to manage resources based on workload. You can pause or resume your data pool as needed to optimize costs.

**Use Cases:**

* **Data Warehousing**: Storing and analysing large datasets from various sources.
* **Business Intelligence**: Running complex queries for dashboards and reports.
* **Analytics**: Performing large-scale data analysis to derive insights from structured data.

**Serverless SQL Pool:**

A Serverless SQL Pool in Azure Synapse Analytics is a distributed, on-demand query processing service that allows you to run T-SQL queries over data stored in Azure Data Lake, Azure Blob Storage, or other supported storage solutions, without the need to provision or manage any compute resources.

**Key Features of Serverless SQL Pool:**

1. **On-Demand Querying**:

- You can run queries on your data directly from storage (like Azure Data Lake or Blob Storage) without the need to load data into a data warehouse or provision any compute infrastructure upfront.

1. **No Infrastructure Management**:

- There is no need to manage or provision compute resources. The compute is automatically handled by Azure, making it suitable for scenarios where you don’t need constant processing power.

1. **Pay-Per-Query**:

- Unlike the Dedicated SQL Pool, which charges for provisioned compute, Serverless SQL Pool uses a **pay-per-query model**, where you only pay for the data processed by the queries, making it cost-efficient for infrequent or exploratory queries.

1. **Scalable**:

- The underlying architecture of Serverless SQL Pool can scale automatically based on the size of the data and the complexity of the query, so you don't have to worry about performance scaling.

1. **T-SQL Compatibility**:

- Serverless SQL Pools support standard T-SQL, meaning you can use familiar SQL syntax to query your data. It supports complex queries, joins, and aggregations over structured and semi-structured data formats (like CSV, Parquet, or JSON).

1. **Supports Data in Azure Data Lake and Blob Storage**:

- You can directly query data stored in **Azure Data Lake Storage Gen2** or **Azure Blob Storage**. This enables you to perform data analysis without the need for traditional ETL (Extract, Transform, Load) processes to move data into a database.

1. **Integrated with Azure Synapse Studio**:

- Serverless SQL Pools can be accessed and managed through **Azure Synapse Studio**, where you can perform queries, manage data, and monitor query execution.

**Use Cases:**

* **Ad-Hoc Queries**: Perfect for scenarios where you need to run quick, exploratory queries without worrying about long-term infrastructure or ongoing costs.
* **Data Exploration**: Ideal for data scientists or analysts who need to explore datasets without requiring the data to be loaded into a traditional relational database.
* **Cost-Effective Analytics**: If you don’t need continuous querying and need to query large datasets occasionally, Serverless SQL Pools are an excellent, cost-effective solution.
* **Data Transformation**: You can run queries to perform transformations on data before loading it into a more structured storage system.