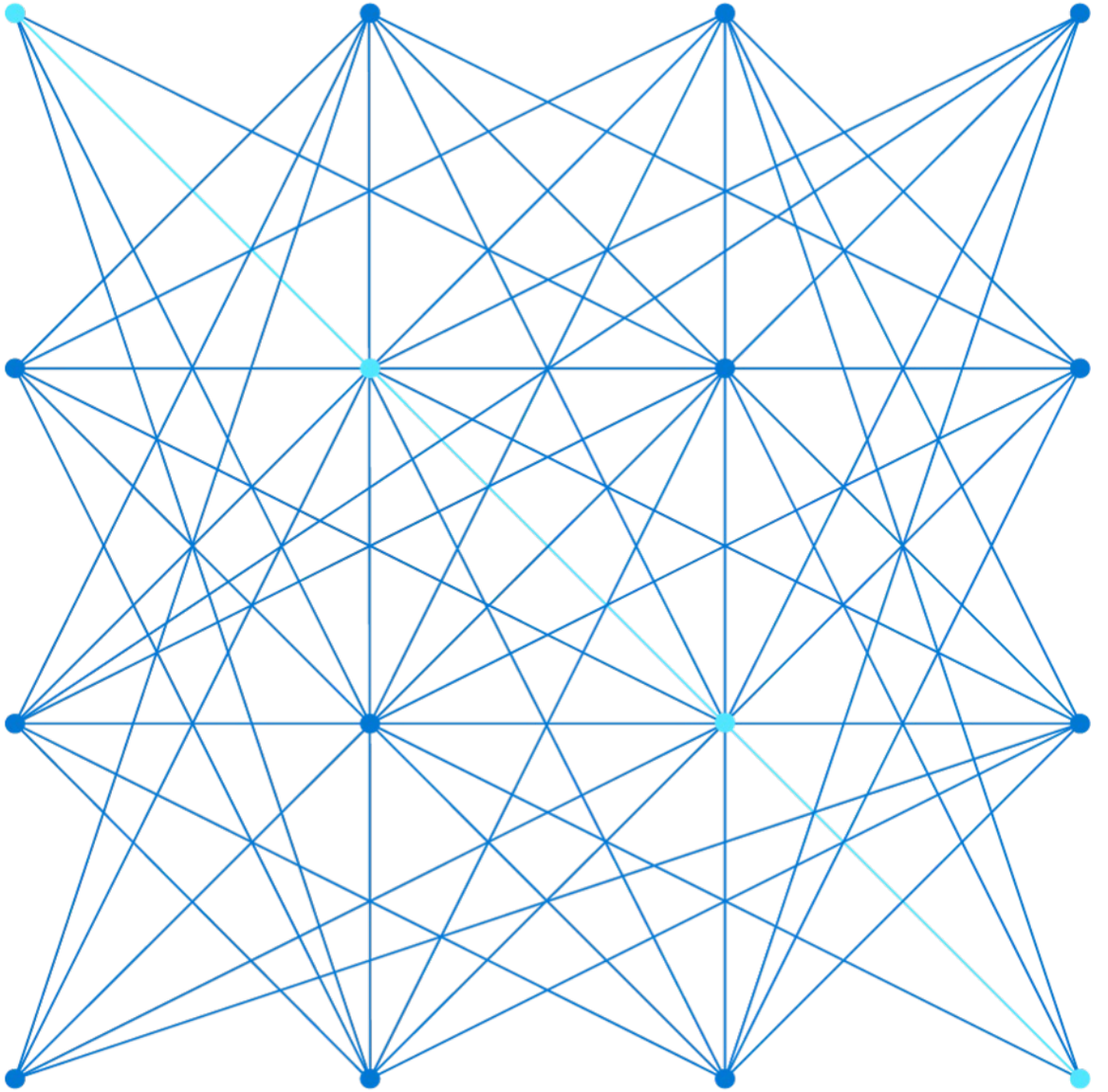
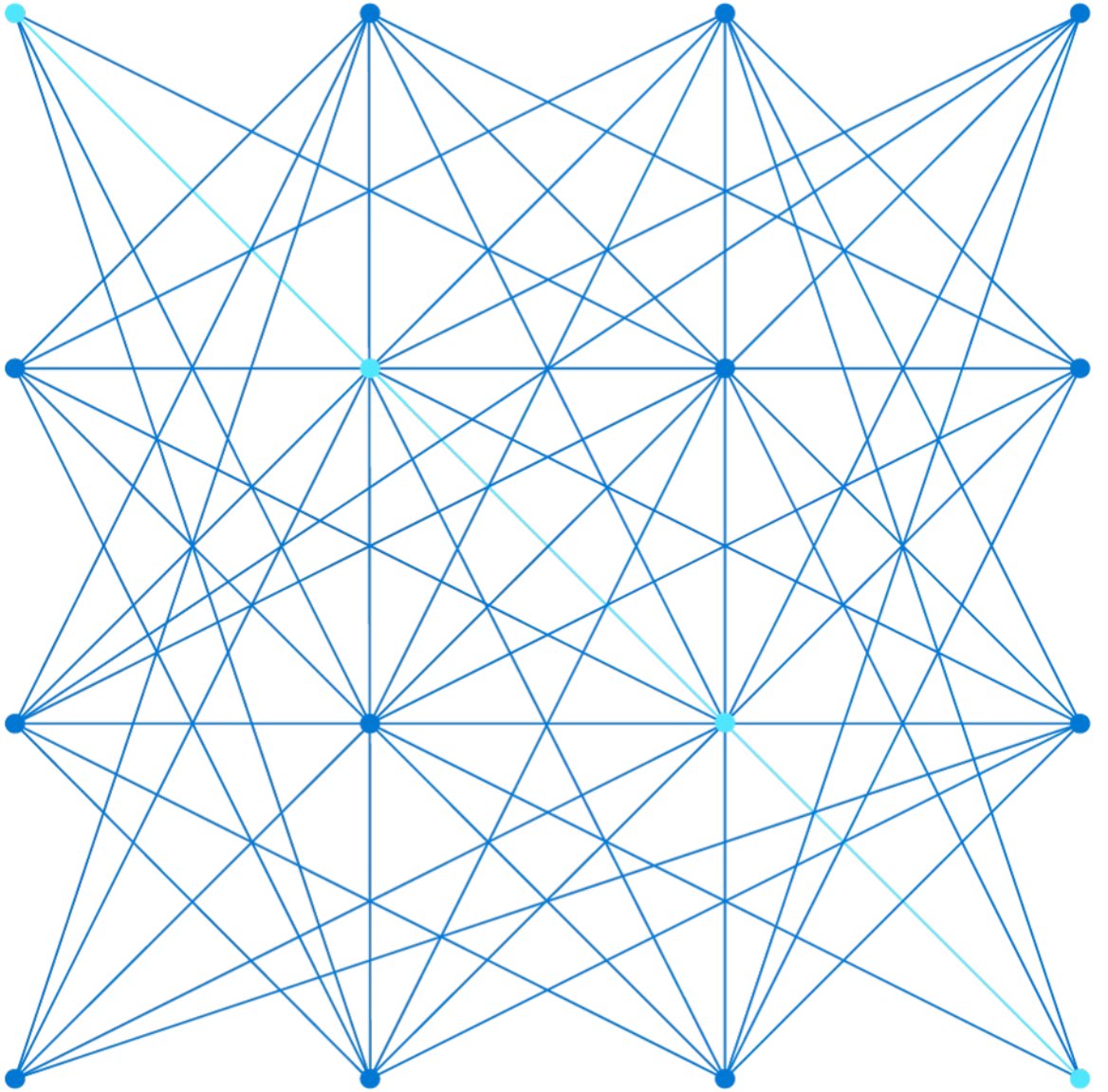
Microsoft Azure

Azure Data Engineering – Day 4



Microsoft Azure

Compute and storage options for data engineering workloads



Agenda

Lesson 01 – Azure Synapse Analytics

Lesson 02 – Azure Databricks

Lesson 03 – Azure Data Lake storage

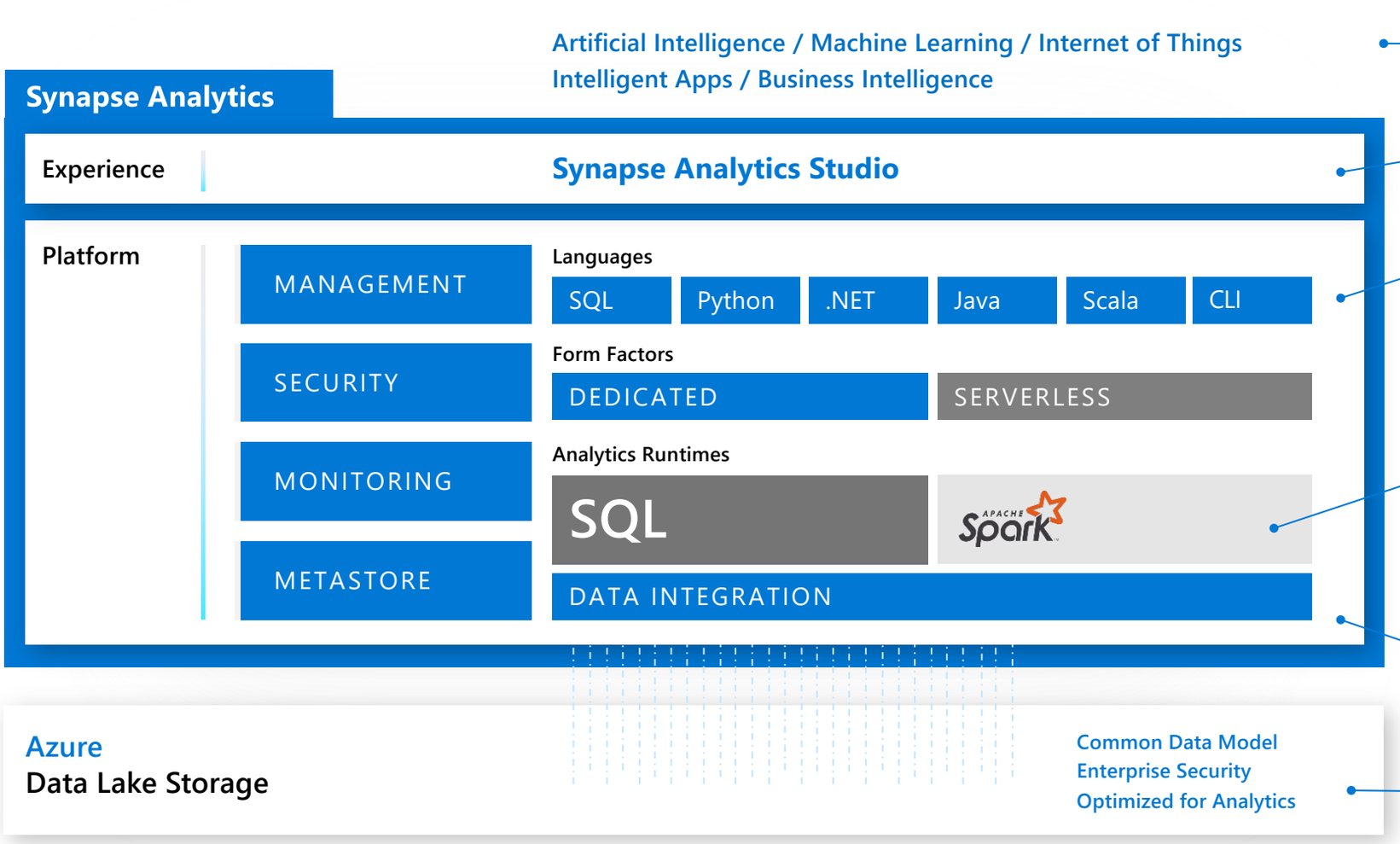
Lesson 04 – Describe Delta Lake architecture

Lesson 05 – Work with data streams by using Azure Stream Analytics

Lesson 01: Introduction to Azure Synapse Analytics

Azure Synapse Analytics

Limitless analytics service with unmatched time to insight



Designed for analytics **workloads at any scale**

SaaS **developer experiences** for code free and code first

Multiple **languages** suited to different analytics workloads

Integrated analytics runtimes available as a dedicated and serverless offering

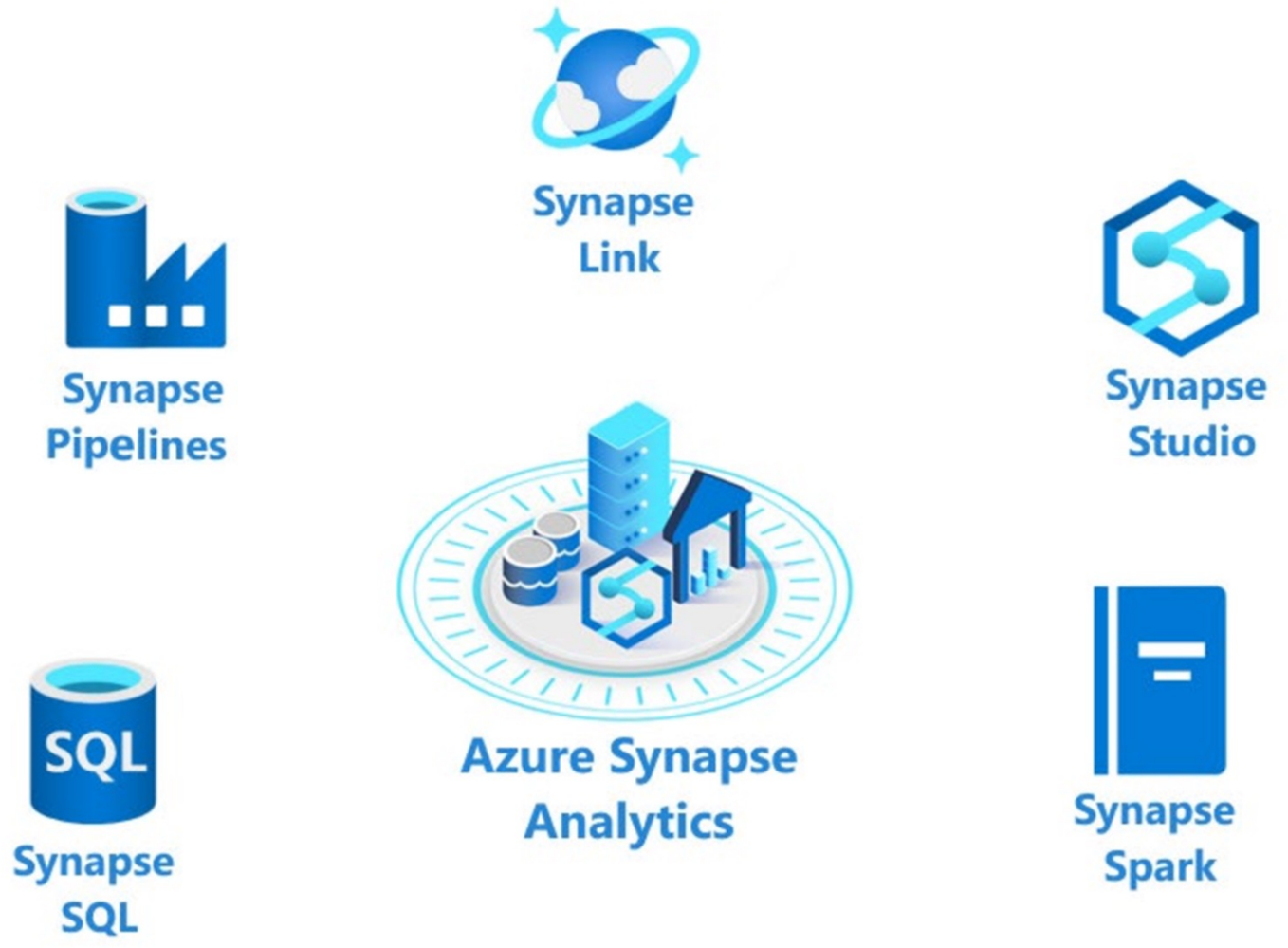
SQL pools offering T-SQL for batch, streaming and interactive processing

Spark pools for big data processing with Python, Scala, R and .NET

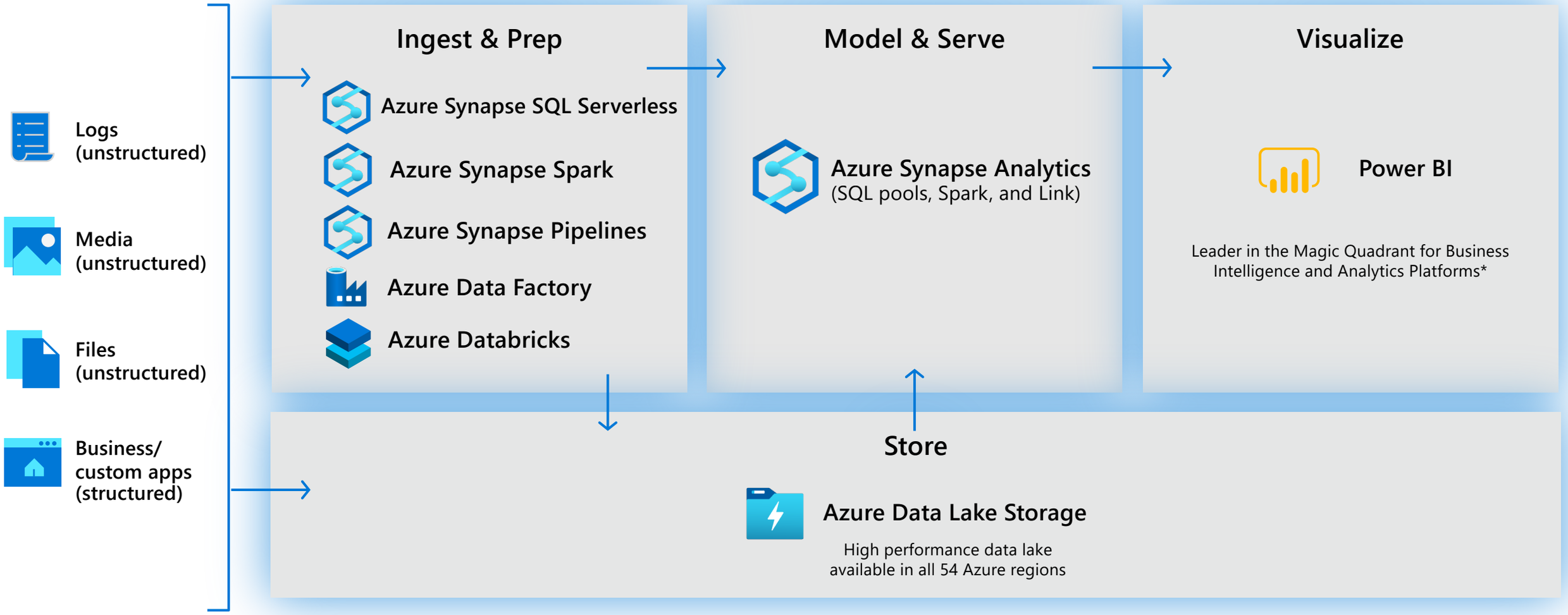
Integrated **platform services** for, management, security, monitoring, and metastore

Data **lake integrated** and Common Data Model aware

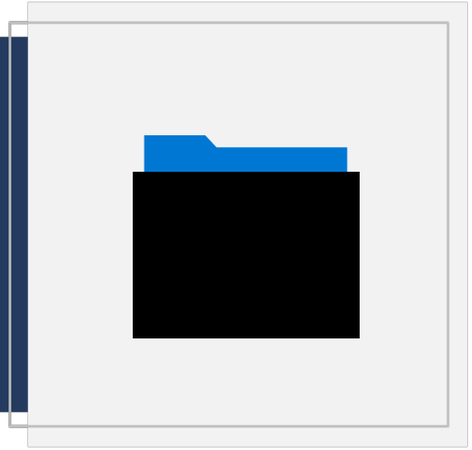
Introduction to Azure Synapse Analytics



Modern data warehousing pattern with Azure Synapse Analytics



Lesson 01: Describe Azure Databricks



What is Azure Databricks

Apache Spark-based analytics platform:

Simplifies the provisioning and collaboration of Apache Spark-based analytical solutions dealing with batch and streaming data

Comprehensive Spark library support:

Support includes SQL, DataFrames, MLlib, Hyperspace and MSSparkUtil

Enterprise Security:

Utilizes the security capabilities of Azure

Integration with other Cloud Services:

Can integrate with a variety of Azure data platform services and Power BI

What is Apache Spark

Apache Spark emerged to provide a parallel processing framework that supports in-memory processing to boost the performance of big-data analytical applications on massive volumes of data

Interactive Data Analysis:

Used by business analysts or data engineers to analyze and prepare data

Streaming Analytics:

Ingest data from technologies such as Kafka and Flume to ingest data in real-time

Machine Learning:

Contains a number of libraries that enables a Data Scientist to perform Machine Learning

Why use Azure Databricks?

Azure Databricks is a wrapper around Apache Spark that simplifies the provisioning and configuration of a Spark cluster in a GUI interface

Azure Databricks components:

Spark SQL and DataFrames

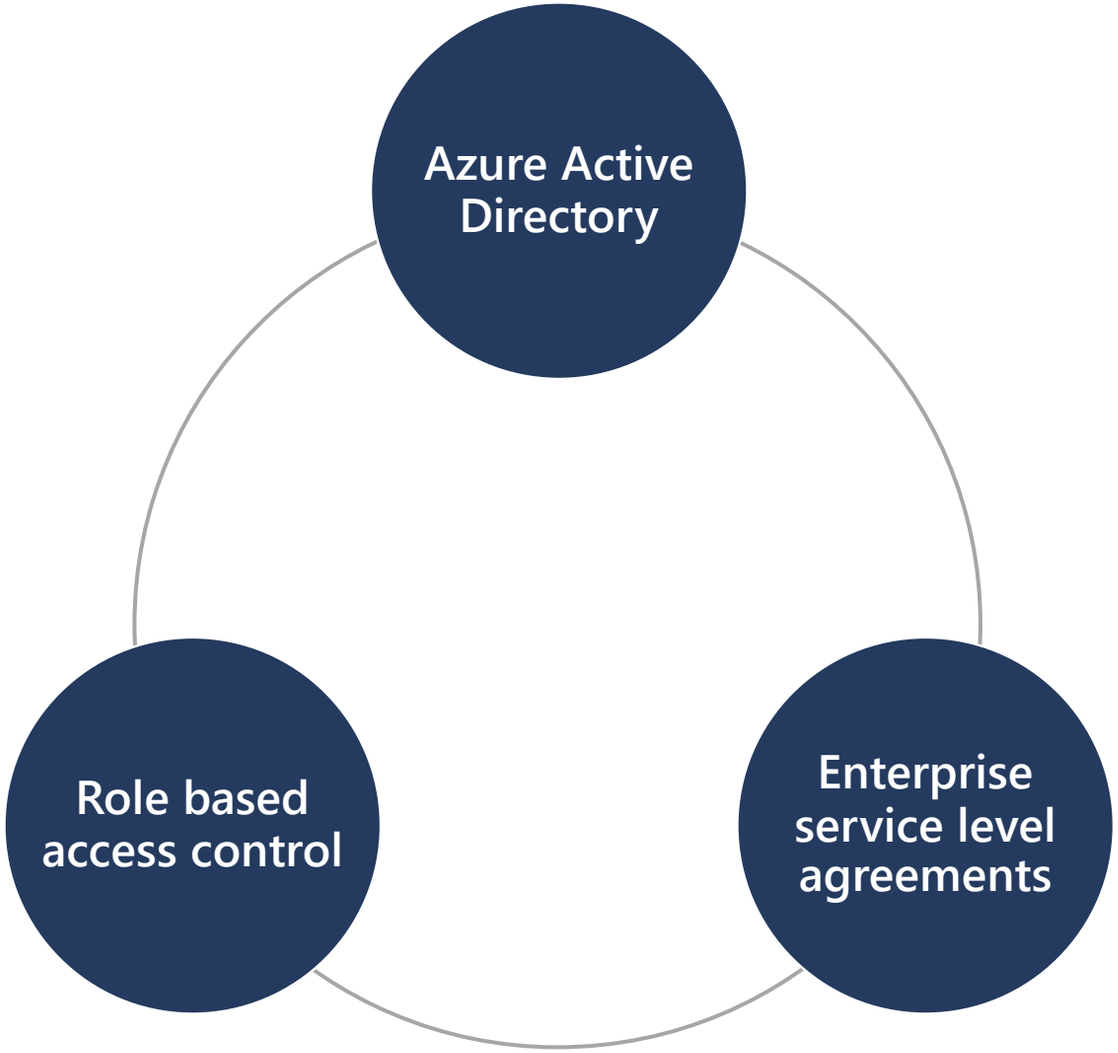
Streaming

Mlib

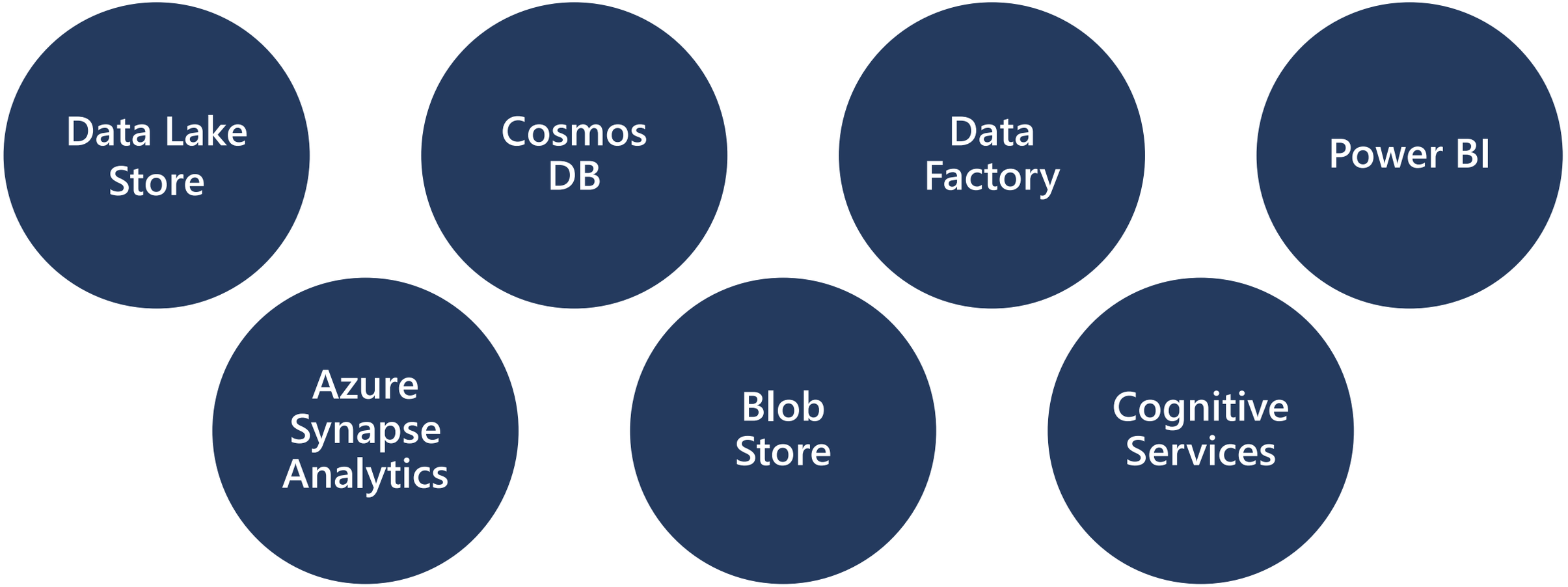
GraphX

Spark Core API

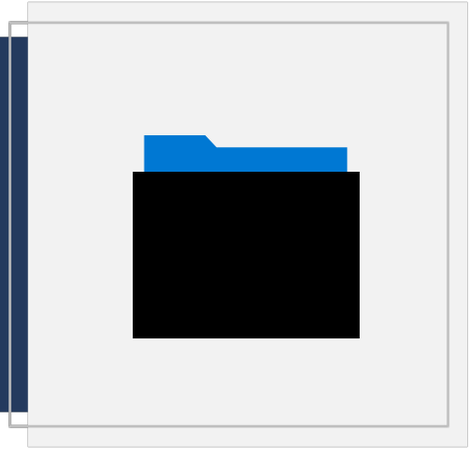
Enterprise security



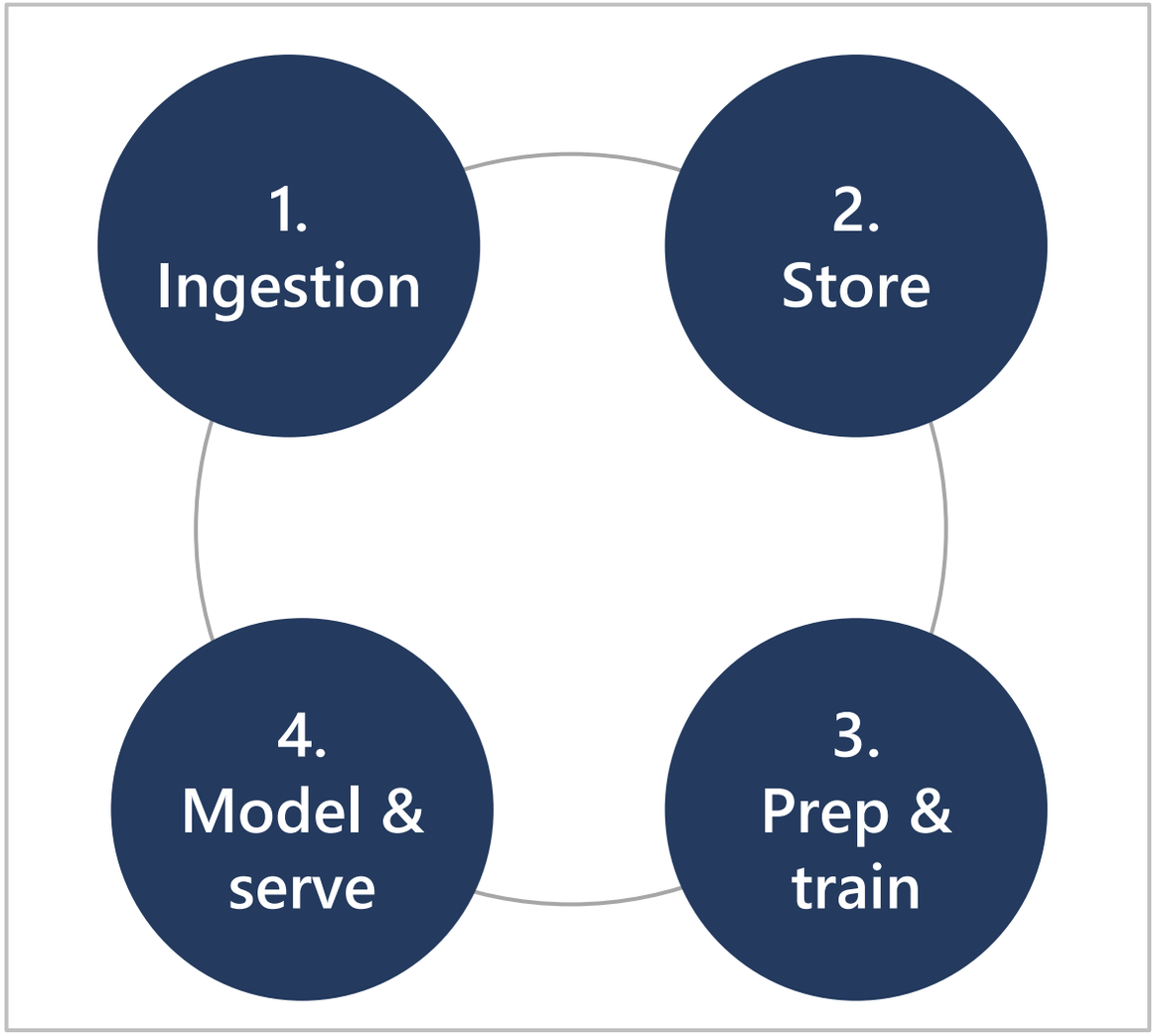
Integration with cloud services



Lesson 01: Introduction to Azure Data Lake storage

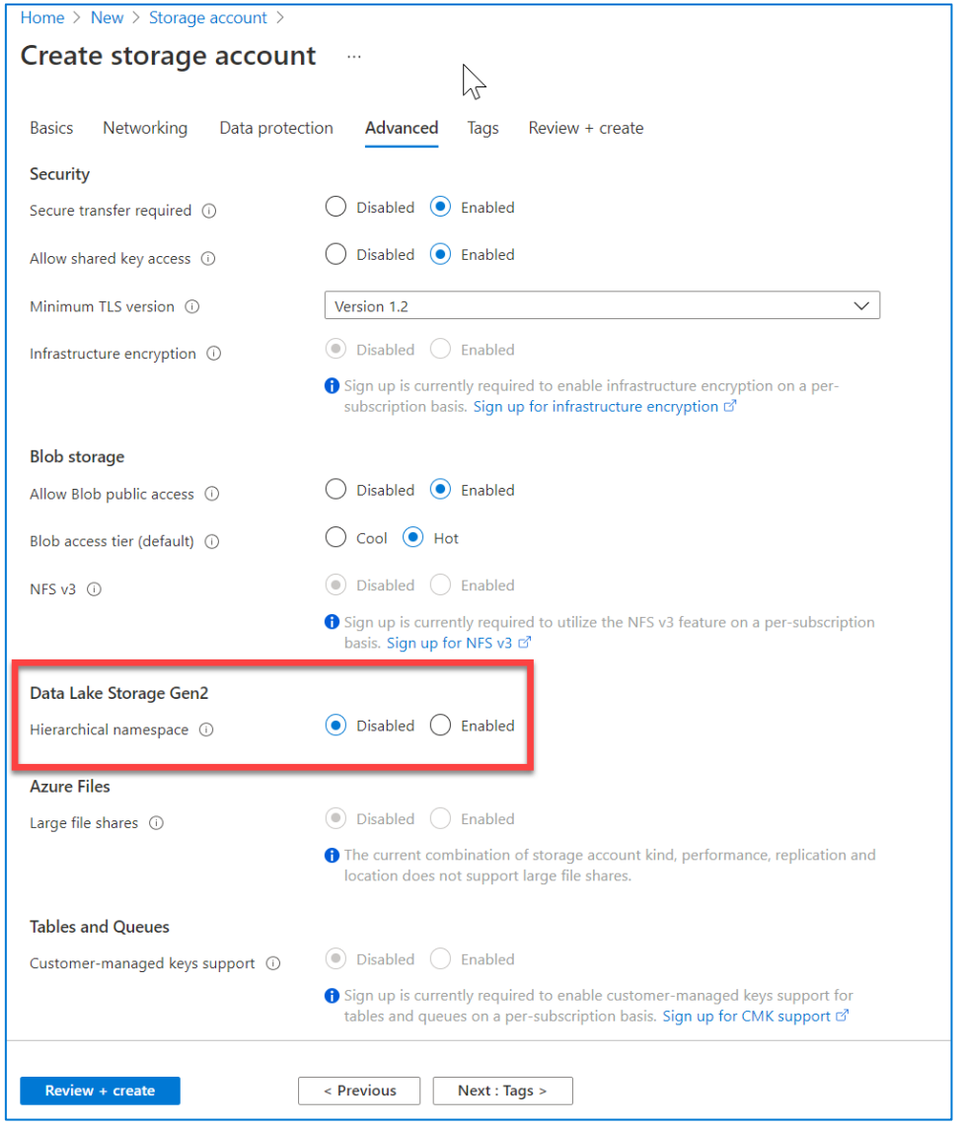


Processing Big Data with Azure Data Lake Store

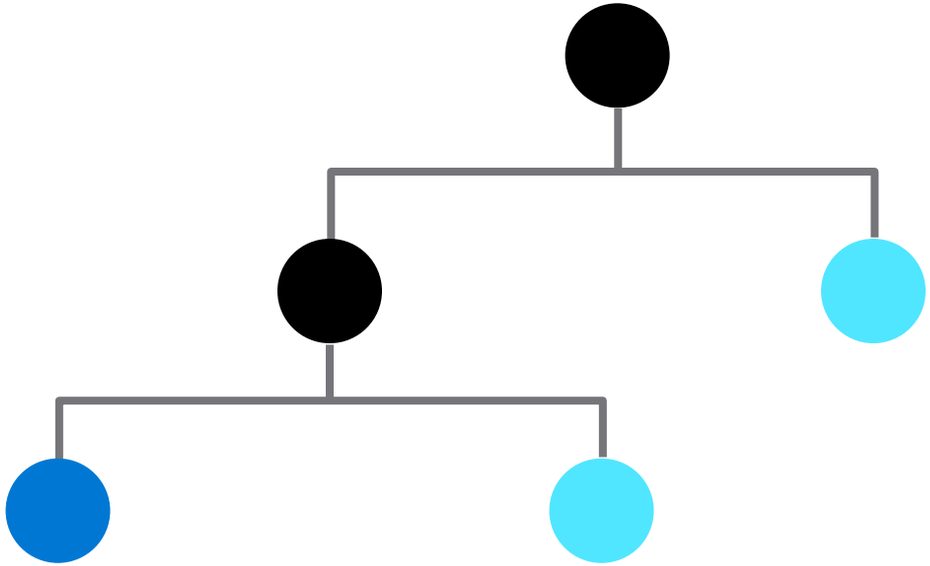


Introduction to Azure Data Lake storage

- Hadoop compatible
- Security
- Performance
- Redundancy



Compare Azure Blob Storage and Data Lake Store Gen 2

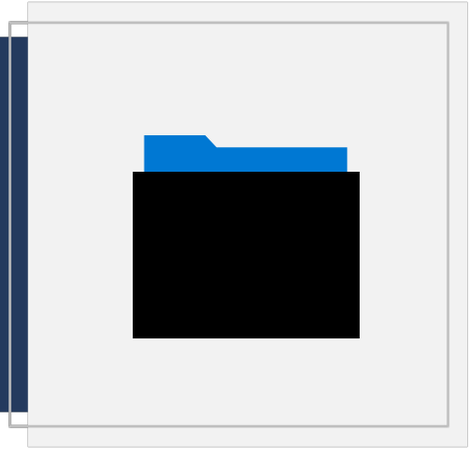
Azure Blob Flat namespace	Data Lake (Gen II) Hierarchical namespace
	

Big Data use cases

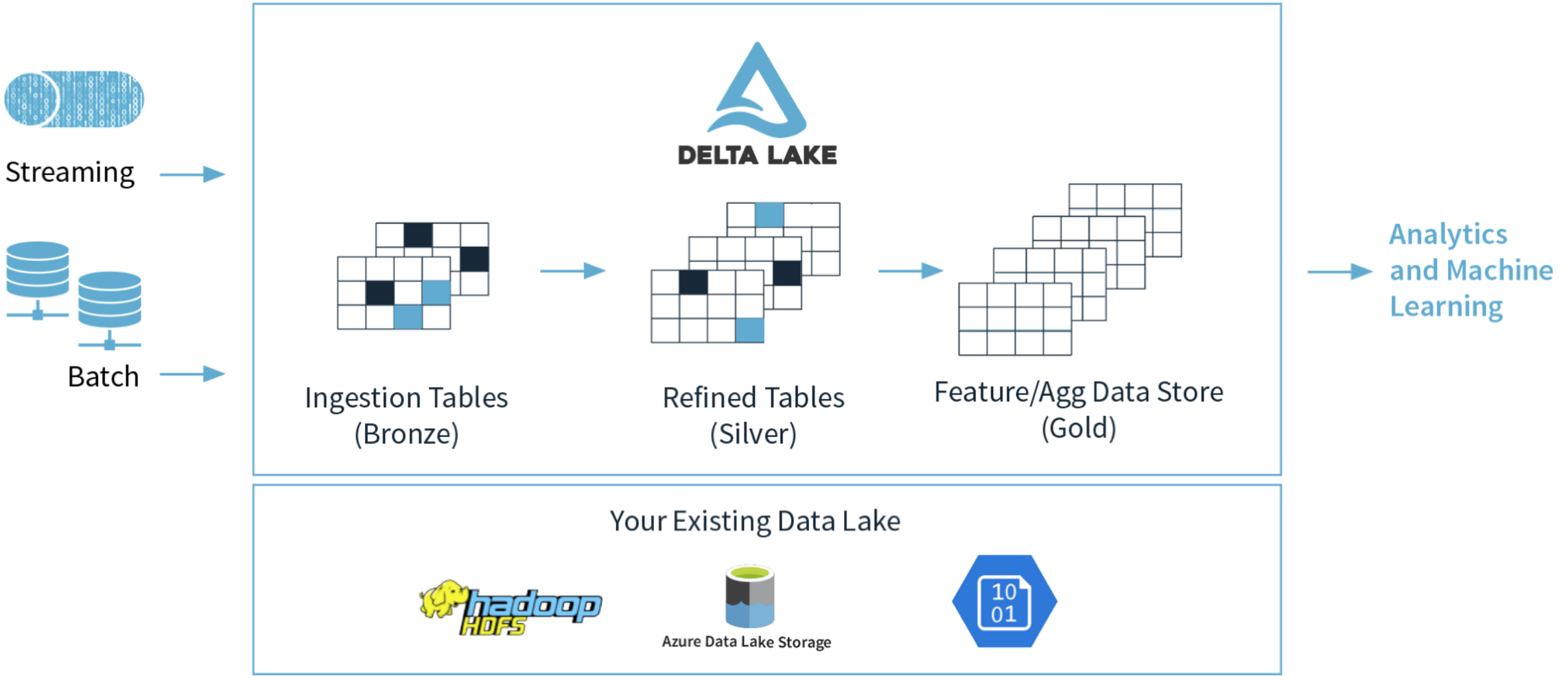
Let’s examine three use cases for leveraging an Azure Data Lake Store

Modern data warehouse	This architecture sees Azure Data Lake Storage at the heart of the solution for a modern data warehouse. Using Azure Data Factory to ingest data into the Data Lake from a business application, and predictive models built in Azure Databricks, using Azure Synapse Analytics as a serving layer
Advanced analytics	In this solution, Azure Data factory is transferring terabytes of web logs from a web server to the Data Lake on an hourly basis. This data is provided as features to the predictive model in Azure Databricks, which is then trained and scored. The result of the model is then distributed globally using Azure Cosmos DB, that an application uses
Real time analytics	In this architecture, there are two ingestion streams. Azure Data Factory is used to ingest the summary files that are generated when the HGV engine is turned off. Apache Kafka provides the real-time ingestion engine for the telemetry data. Both data streams are stored in Data Lake store for use in the future

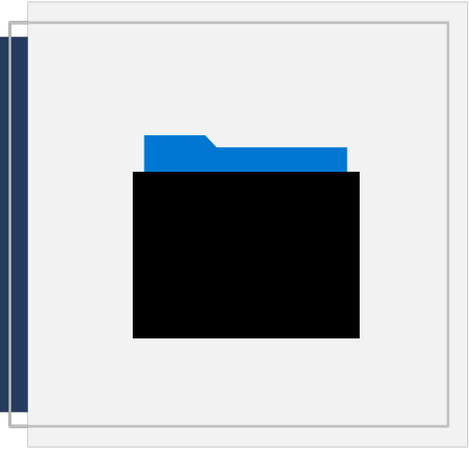
Lesson 01: Describe Delta Lake architecture



Describe a Delta Lake architecture



Lesson 01: Work with data streams by using Azure Stream Analytics



What are data streams

Data streams:

In the context of analytics, data streams are event data generated by sensors or other sources that can be analyzed by another technology

Data stream processing approach:

There are two approaches. Reference data is streaming data that can be collected over time and persisted in storage as static data. In contrast, streaming data have relatively low storage requirements. And run computations in sliding windows

Data streams are used to:

Analyze data:
Continuously analyze data to detect issues and understand or respond to them

Understand systems:
Understand component or system behavior under various conditions to fuel further enhancements of said system

Trigger actions:
Trigger specific actions when certain thresholds are identified

Event processing

The process of consuming data streams, analyzing them, and deriving actionable insights out of them is called Event Processing and has three distinct components:

Event producer	Examples include sensors or processes that generate data continuously such as a heart rate monitor or a highway toll lane sensor
Event processor	An engine to consume event data streams and deriving insights from them. Depending on the problem space, event processors either process one incoming event at a time (such as a heart rate monitor) or process multiple events at a time (such as a highway toll lane sensor)
Event consumer	An application which consumes the data and takes specific action based on the insights. Examples of event consumers include alert generation, dashboards, or even sending data to another event processing engine

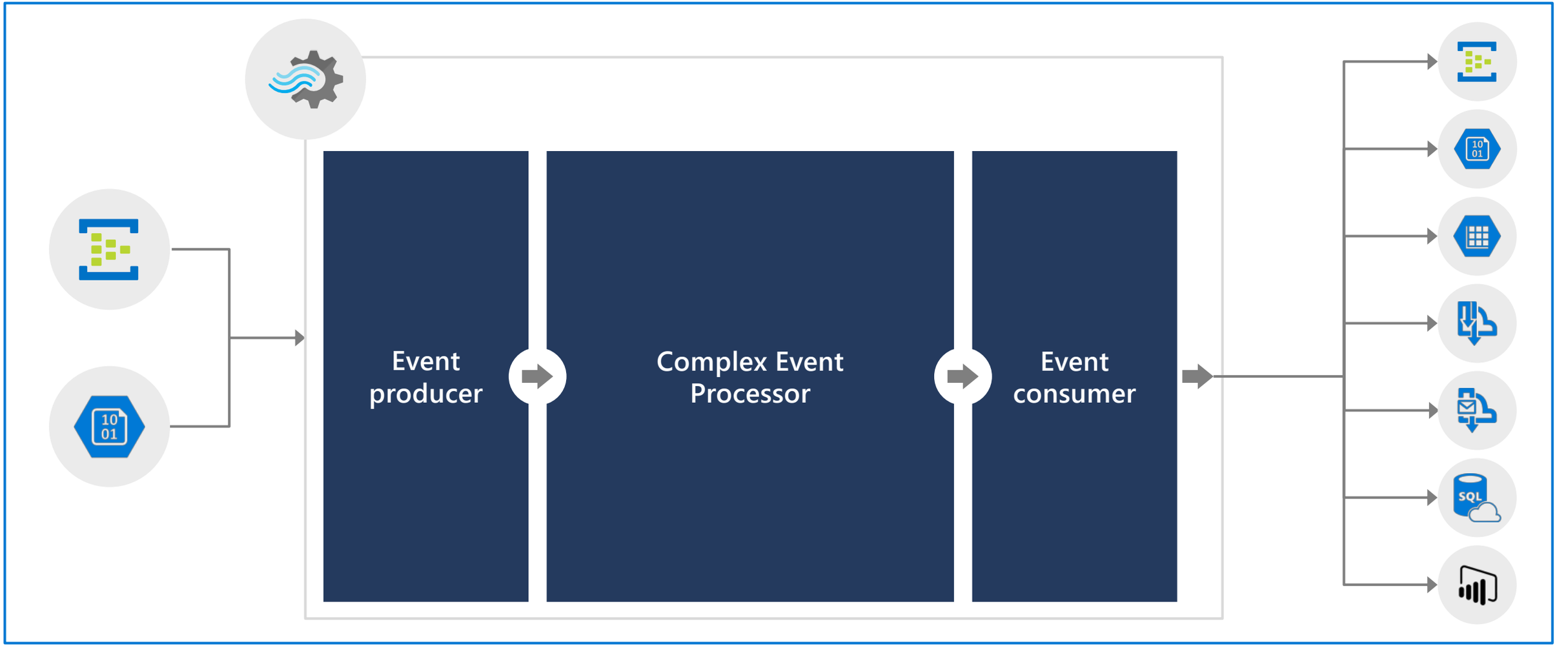
Processing events with Azure Stream Analytics

Microsoft Azure Stream Analytics is an event processing engine. It enables the consumption and analysis of high volumes of streaming data in real time

Source	Ingestion	Analytical engine	Destination
Sensors	Event Hubs	Stream Analytics Query Language	Azure Data Lake
Systems	IoT Hubs	.NET SDK	Cosmos DB
Applications	Azure Blob Store		SQL Database
			Blob Store
			Power BI

Work with data streams by using Azure Stream Analytics

Complex event processing of Stream Data in Azure



Review questions

Q01 – Azure Synapse Analytics offers Synapse SQL in two offerings. What are they?

A01 – Dedicated SQL pools, and serverless SQL pools.

Q02 – Which Azure Storage Account option must be enabled to optimize the storage account as an Azure Data Lake for analytical workloads?

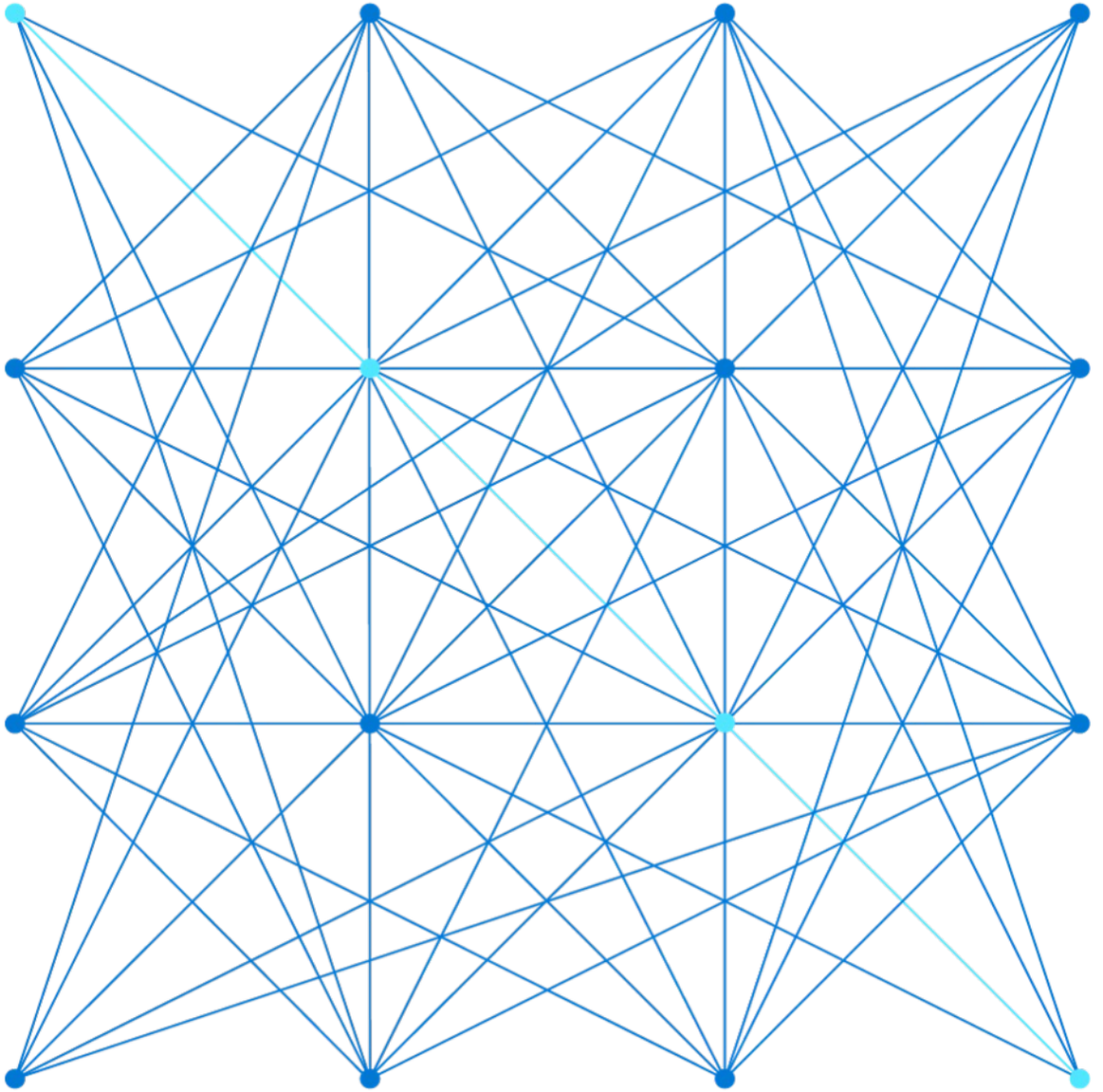
A02 – Hierarchical namespace.

Q03 – Which architecture enriches data through a unified pipeline that allows you to combine batch and streaming workflows?

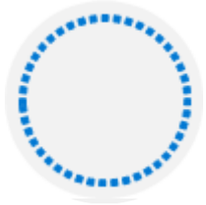
A03 – Delta lake architecture.

Microsoft Azure

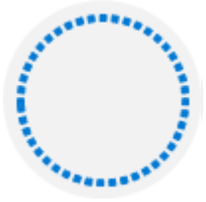
Data Exploration and Transformation in Azure Databricks



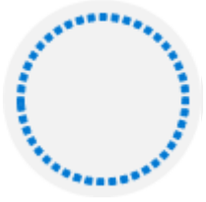
Agenda



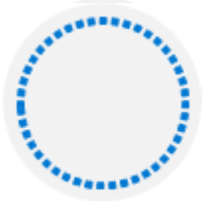
Lesson 01 – Understand Azure Databricks



Lesson 02 – Read and write data in Azure Databricks

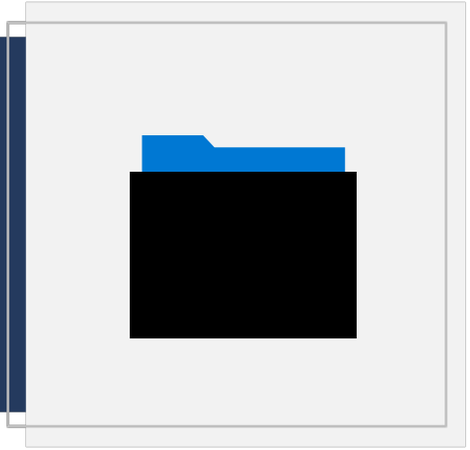


Lesson 03 – Work with DataFrames in Azure Databricks

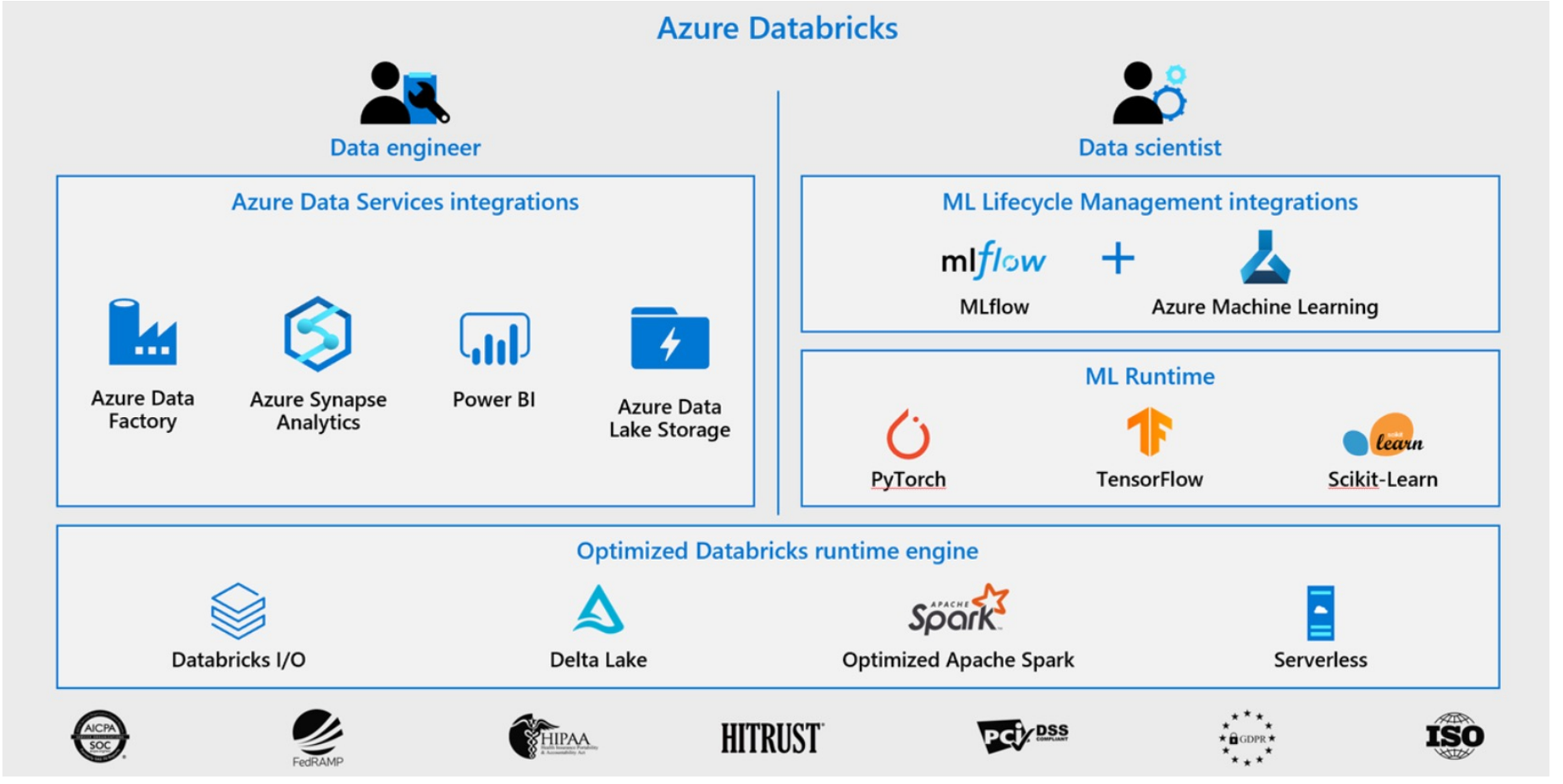


Lesson 04 – Work with DataFrames advanced methods in Azure Databricks

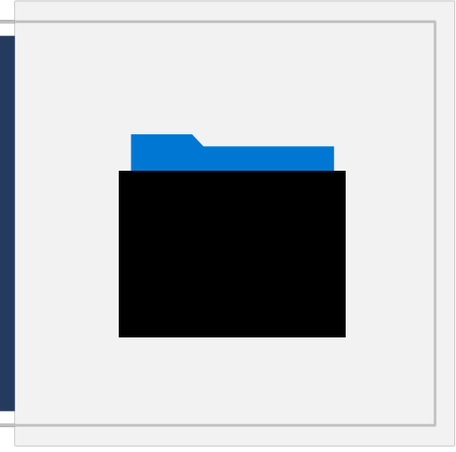
Lesson 01: Understand Azure Databricks



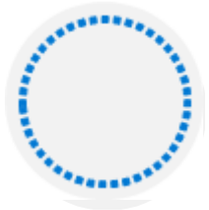
Understand Azure Databricks



Lesson 02: Read and write data in Azure Databricks

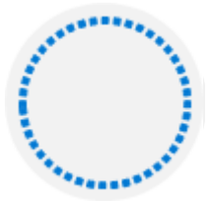


Read and write data in Azure Databricks



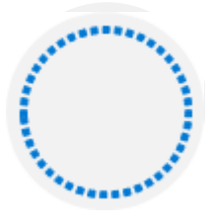
Multiple format support

Reading data from CSV, PARQUET, JSON and many others



Integrated with several Azure Data Services

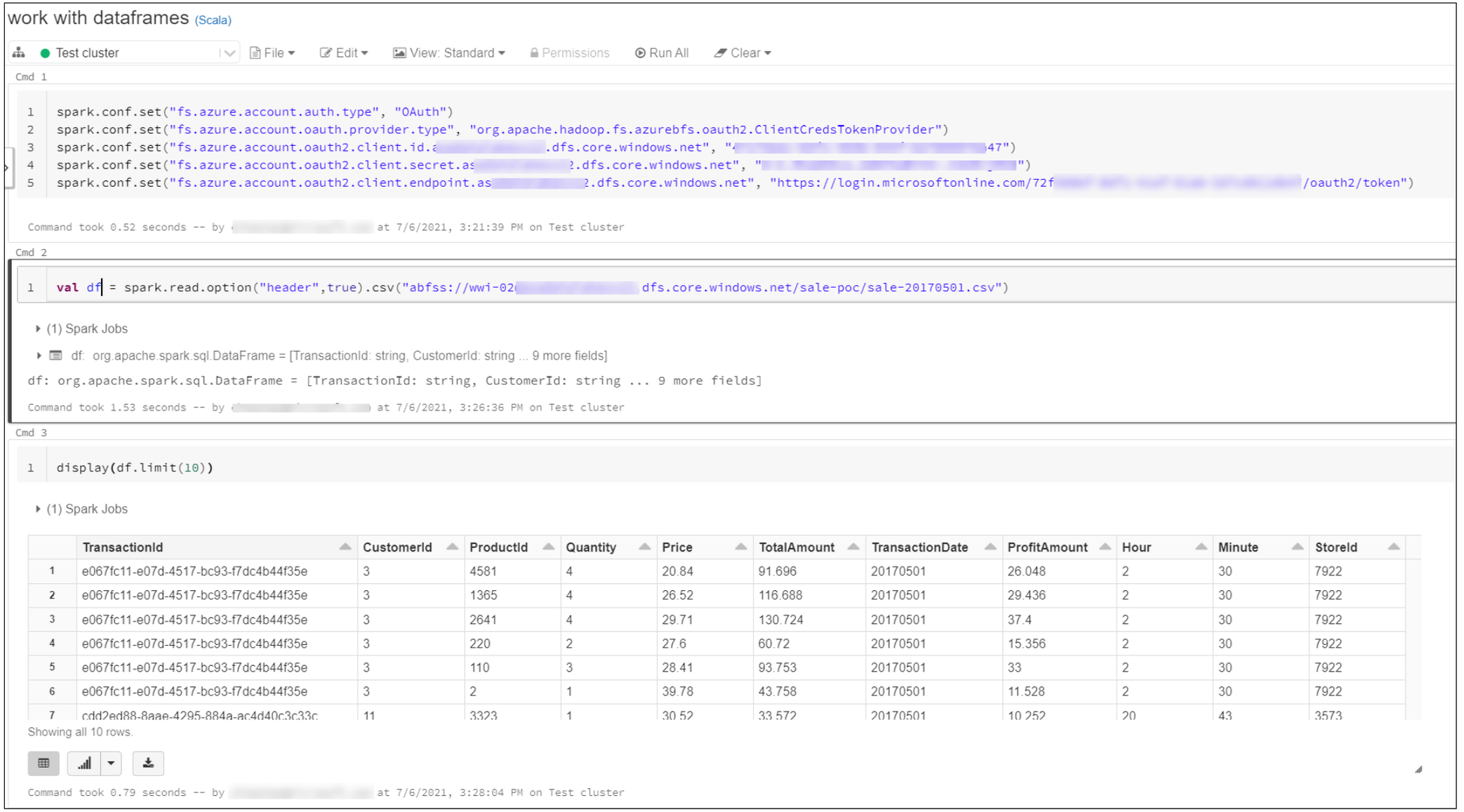
Reading and writing from and to Azure Data Lake Storage, Azure Synapse Analytics, etc.



Notebook experience

Reading and writing by simply writing code in a shared notebook experience

Read data in Azure Databricks



work with dataframes (Scala)

Test cluster | File | Edit | View: Standard | Permissions | Run All | Clear

Cmd 1

```
spark.conf.set("fs.azure.account.auth.type", "OAuth")
spark.conf.set("fs.azure.account.oauth.provider.type", "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider")
spark.conf.set("fs.azure.account.oauth2.client.id.a                .dfs.core.windows.net", "                                47")
spark.conf.set("fs.azure.account.oauth2.client.secret.as          2.dfs.core.windows.net", "                                ")
spark.conf.set("fs.azure.account.oauth2.client.endpoint.as          2.dfs.core.windows.net", "https://login.microsoftonline.com/72f                                /oauth2/token")
```

Command took 0.52 seconds -- by [redacted] at 7/6/2021, 3:21:39 PM on Test cluster

Cmd 2

```
val df = spark.read.option("header",true).csv("abfss://wwi-02                dfs.core.windows.net/sale-poc/sale-20170501.csv")
```

(1) Spark Jobs

df: org.apache.spark.sql.DataFrame = [TransactionId: string, CustomerId: string ... 9 more fields]

```
df: org.apache.spark.sql.DataFrame = [TransactionId: string, CustomerId: string ... 9 more fields]
```

Command took 1.53 seconds -- by [redacted] at 7/6/2021, 3:26:36 PM on Test cluster

Cmd 3

```
display(df.limit(10))
```

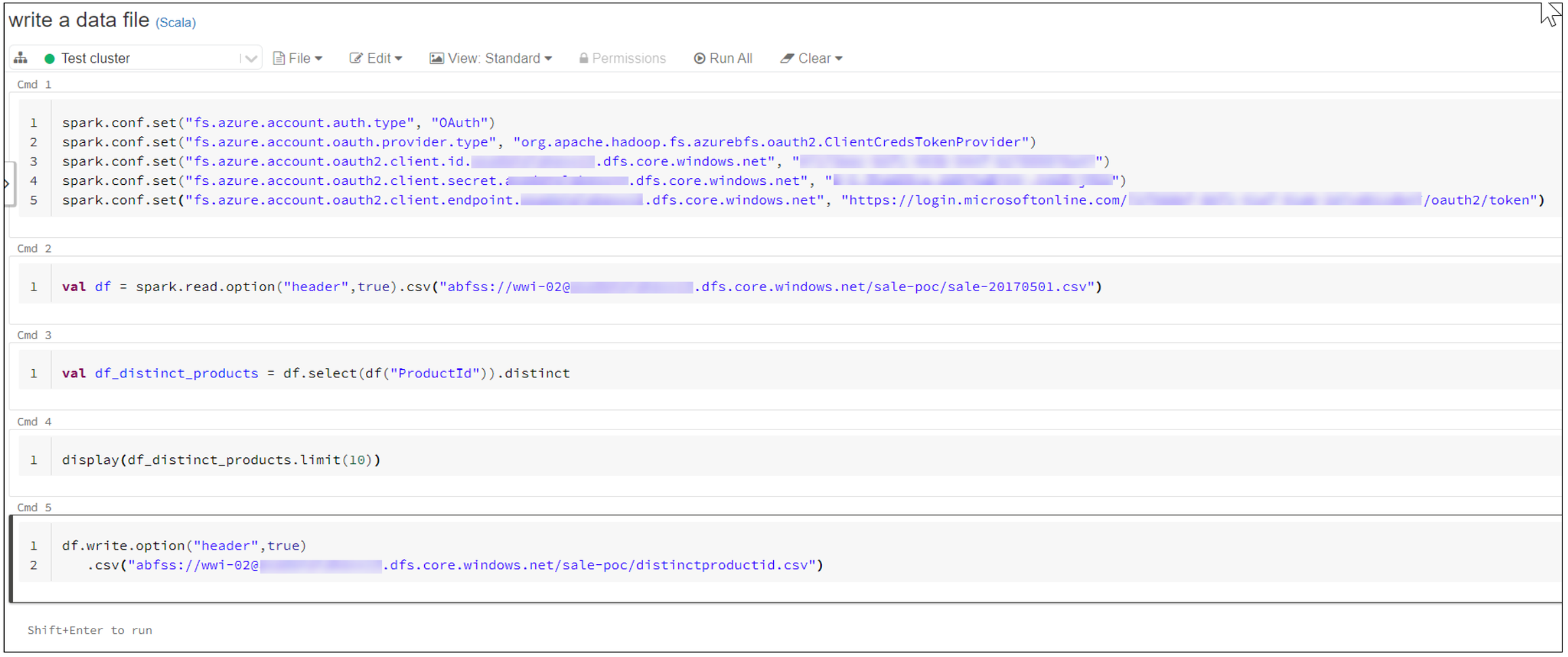
(1) Spark Jobs

	TransactionId	CustomerId	ProductId	Quantity	Price	TotalAmount	TransactionDate	ProfitAmount	Hour	Minute	StoreId
1	e067fc11-e07d-4517-bc93-f7dc4b44f35e	3	4581	4	20.84	91.696	20170501	26.048	2	30	7922
2	e067fc11-e07d-4517-bc93-f7dc4b44f35e	3	1365	4	26.52	116.688	20170501	29.436	2	30	7922
3	e067fc11-e07d-4517-bc93-f7dc4b44f35e	3	2641	4	29.71	130.724	20170501	37.4	2	30	7922
4	e067fc11-e07d-4517-bc93-f7dc4b44f35e	3	220	2	27.6	60.72	20170501	15.356	2	30	7922
5	e067fc11-e07d-4517-bc93-f7dc4b44f35e	3	110	3	28.41	93.753	20170501	33	2	30	7922
6	e067fc11-e07d-4517-bc93-f7dc4b44f35e	3	2	1	39.78	43.758	20170501	11.528	2	30	7922
7	cdd2ed88-8aae-4295-884a-ac4d40c3c33c	11	3323	1	30.52	33.572	20170501	10.252	20	43	3573

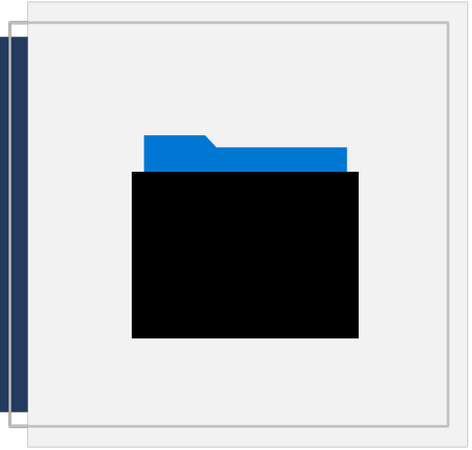
Showing all 10 rows.

Command took 0.79 seconds -- by [redacted] at 7/6/2021, 3:28:04 PM on Test cluster

Write data in Azure Databricks



Lesson 03: Work with DataFrames in Azure Databricks



Work with DataFrames in Azure Databricks

- Apache Spark DataFrame API reading data in a single command

```
parquetDir = source + "/wikipedia/pagecounts/staging_parquet_en_only_clean/"

pagecountsEnAllDF = (spark  # Our SparkSession & Entry Point
  .read                     # Our DataFrameReader
  .parquet(parquetDir)      # Returns an instance of DataFrame
)
print(pagecountsEnAllDF)    # Python hack to see the data type
```

Working with transformations in Azure Databricks

Transformations	Description
Select(…)	The select(…) command enables you to specify the columns to include in a query
drop(…)	The drop(…) command enables you to specify the columns you don’t want
distinct(…)	The distinct(…) command returns a distinct set of values in a DataFrame
dropDuplicates(…)	The dropDuplicates(…) command is an alias of the distinct(…) command.
show(…)	The show(..) command is part of the core Spark API and simply prints the results to the console
display(…)	The display(…) command provides more flexibility than show(…) such as downloading results against csv, rendering charts and showing up to 100 rows
limit(…)	The limit(…) command can be used to control the number of records that are returned to a DataFrame

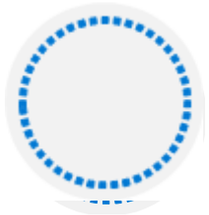
Optimize DataFrames in Azure Databricks

- Mix DataFrame operations

```
(pagecountsEnAllDF
  .cache()          # Mark the DataFrame as cached
  .count()          # Materialize the cache
)
```

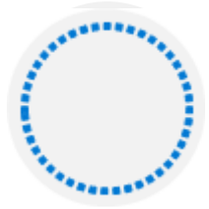
Lesson 04: Work with DataFrames advanced methods in Azure Databricks

Work with DataFrames advanced methods in Azure Databricks



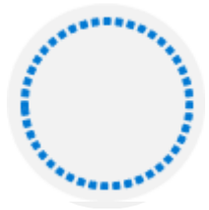
DateTime manipulation

Enabling different DateTime techniques to use across DataFrames



Aggregate Functions

groupBy() function, sum(), count(), avg(), min(), max() functions



Deduplication of Data

Removing duplicates, by ensuring you only keep 1 record

Review questions

Q01 – How do you list files in DBFS within a notebook?

A01 – %fs ls /my-file-path

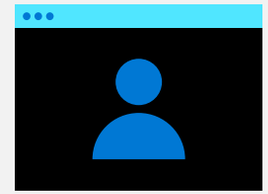
Q02 – How do you create a DataFrame object?

A02 – Introduce a variable name and equate it to something like myDataFrameDF =

Q03 – You need to find the average of sales transactions by storefront. Which of the following aggregates would you use?

A03 – df.groupBy(col("storefront")).avg("completedTransactions")

Lab: Explore compute and storage options for data engineering workloads



Lab overview

This lab teaches ways to structure the data lake, and to optimize the files for exploration, streaming, and batch workloads. The student will learn how to organize the data lake into levels of data refinement as they transform files through batch and stream processing. The students will also experience working with Apache Spark in Azure Synapse Analytics. They will learn how to create indexes on their datasets, such as CSV, JSON, and Parquet files, and use them for potential query and workload acceleration using Spark libraries including Hyperspace and MSSParkUtils.

Lab objectives

After completing this lab, you will be able to:

Work with a Delta Lake architecture

Working with Apache Spark in Azure Synapse Analytics

Module summary

In this module, you have learned about:

- Azure Synapse Analytics
- Azure Databricks
- Azure Data Lake
- Delta Lake architectures
- Azure Stream Analytics

Next steps

After the course, consider visiting [the Microsoft Customer Case Study site]. Use the search bar to search by an industry such as healthcare or retail, or by a technology such as Azure Synapse Analytics or Azure Databricks. Read through some of the customers stories

Microsoft Azure

Azure Data Platform technologies

Storage Account

- When you need a **low cost, high throughput** data store
- When you need to store **No-SQL** data
- When you **do not need to query** the data directly. **No ad hoc query** support
- Suits the storage of archive or **relatively static data**
- Suits acting as a **HDInsight Hadoop** data store

Data Lake Store

- When you need a **low cost, high throughput** data store
- **Unlimited storage for No-SQL** data
- When you **do not need to query** the data directly. **No ad hoc query** support
- Suits the storage of archive or **relatively static data**
- Suits acting as a **Databricks , HDInsight** and **IoT** data store

Azure Databricks

- **Eases the deployment** of a Spark based cluster
- Enables the **fastest processing** of Machine Learning solutions
- **Enables collaboration** between data engineers and data scientists
- Provides **tight enterprise security integration** with Azure Active Directory
- **Integration with other Azure Services and Power BI**

Azure COSMOS DB

- Provides **global distribution** for both structured and unstructured data stores
- **Millisecond query response** time
- **99.999% availability** of data
- **Worldwide elastic scale** of both the storage and throughput
- **Multiple consistency levels** to control data integrity with concurrency

Azure SQL Database

- When you require a **relational** data store
- When you need to manage **transactional workloads**
- When you need to manage a **high volume on inserts and reads**
- When you need a service that **requires high concurrency**
- When you require a solution that can scale **elastically**

Azure Data Platform technologies (continued)

Azure Synapse Analytics

- When you require an integrated **relational** and **big data** store
- When you need to manage **data warehouse** and **analytical workloads**
- When you need **low cost storage**
- When you require the ability to **pause and restart the compute**
- When you require a solution that can scale **elastically**

Azure Stream Analytics

- When you require a **fully managed event processing** engine
- When you require **temporal analysis of streaming** data
- Support for analyzing **IoT streaming** data
- Support for analyzing application data through **Event Hubs**
- Ease of use with a **Stream Analytics Query Language**

Azure Data Factory

- When you want to **orchestrate the batch movement** of data
- When you want to connect to **wide range of data platforms**
- When you want to **transform or enrich** the data in movement
- When you want to **integrate with SSIS packages**
- Enables **verbose logging** of data processing activities

Azure HDInsight

- When you need a **low cost, high throughput** data store
- When you need to store **No-SQL** data
- Provides a Hadoop **Platform as a Service** approach
- Suits acting as a **Hadoop, Hbase, Storm or Kafka** data store
- **Eases the deployment and management** of clusters

Azure Purview

- When you require **documentation** of your data stores
- When you require a **multi user** approach to documentation
- When you need to **classify data sources**
- A **fully managed cloud service** whose users can discover the data sources
- When you require **a solution that can help any user** understand their data