DESIGNING A SOLUTION

That utilizes Cosmos DB, Data Lake Gen 2 or Blob Storage

Scenario 1

- **Company: Fortune 500 car rental**
- **Goal: design the appropriate cloud architecture**
- **Details:**
  - **Millions of customer worldwide**
  - **20,000 order per day from locations all over glob**
  - **Car pricing is dynamic, based on demand**
  - **Allow orders from various web portals**
  - **Store data from variety of sources**
  - **Provide reporting for multiple business silos**

- **Options:**
  - **Cosmos DB**
  - **Data Lake Gen 2**
  - **Blob Storage**

Scenario 2

- **Company: Fortune 500 financial planning**
- **Goal: design the appropriate cloud architecture**
- **Details:**
  - **Provide business intelligence to finance, human resources, and project management**
  - **Information coming from servers all over the united states**
  - **Allow business analysts to access raw data and build reports as needed.**
- **Options:**
  - **Cosmos DB**
  - **Data Lake Gen 2**
  - **Blob Storage**

Scenario 3

- **Company: Online news agency**
- **Goal: design the appropriate cloud architecture**
- **Details:**
  - **Stores hundreds of terabytes of videos**
  - **Access videos from locations around the globe**
  - **Protect costs as much as possible**
    - **We don’t get paid for videos but ads**
- **Options:**
  - **Cosmos DB**
  - **Data Lake Gen 2**
  - **Blob Storage**

DESIGNING A SOLUTION

That utilizes SQL Server Database and Data Warehouse

Scenario 1

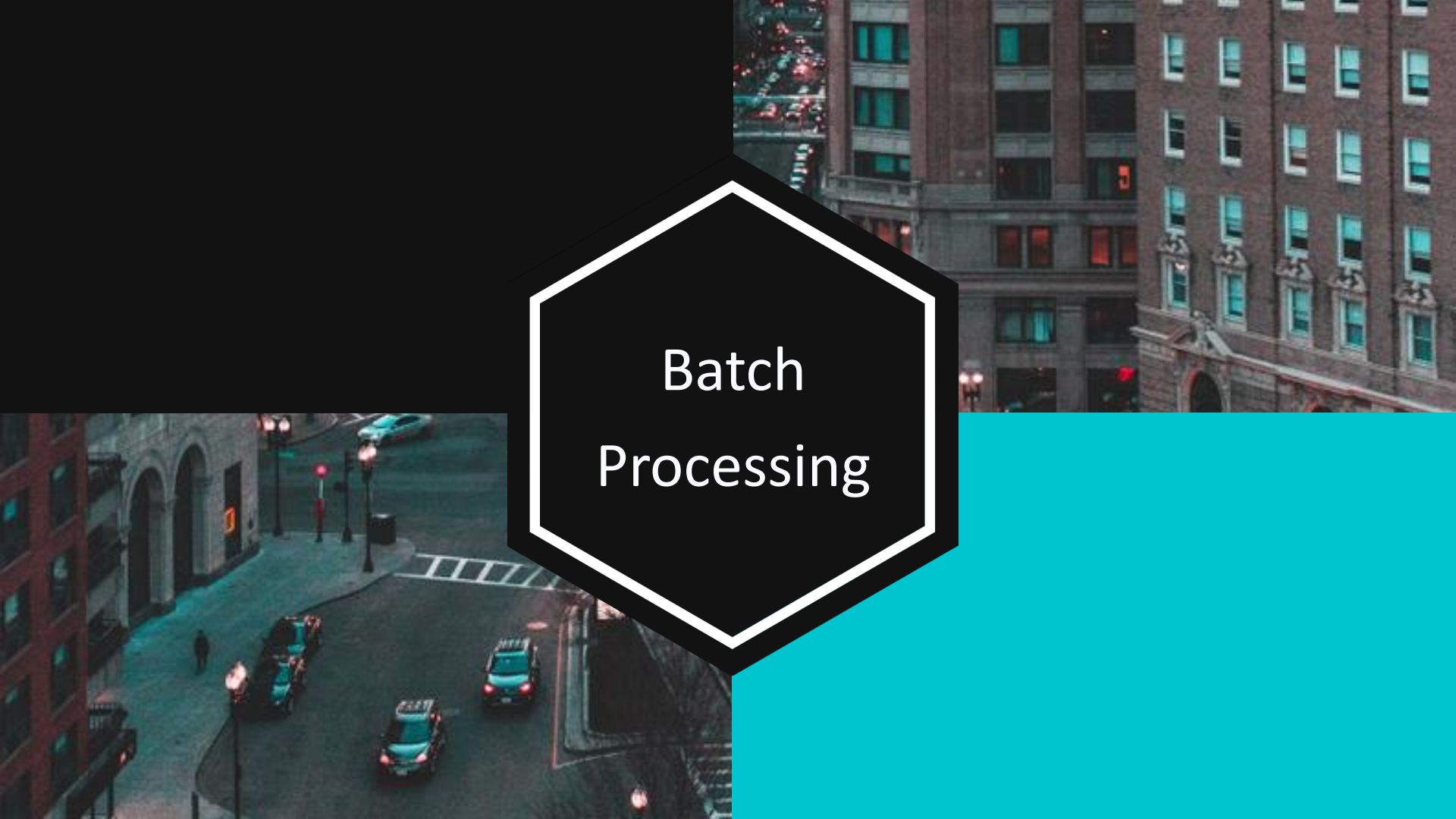
- Company: Craft Company
- Business: Sells to craft shows In different locations
- Need:
  - Collect transactional data at each event.
  - Store this data into 5 different databases based on several factors
    - Only one database is expected to be in use at any given period
  - Client is cost sensitive
  - Build basic executive level reports on data every month

- Options
  - Azure SQL Database – Single
  - Azure SQL Database – Elastic Pool
  - Azure SQL Database – Managed Instance
  - Azure SQL Datawarehouse

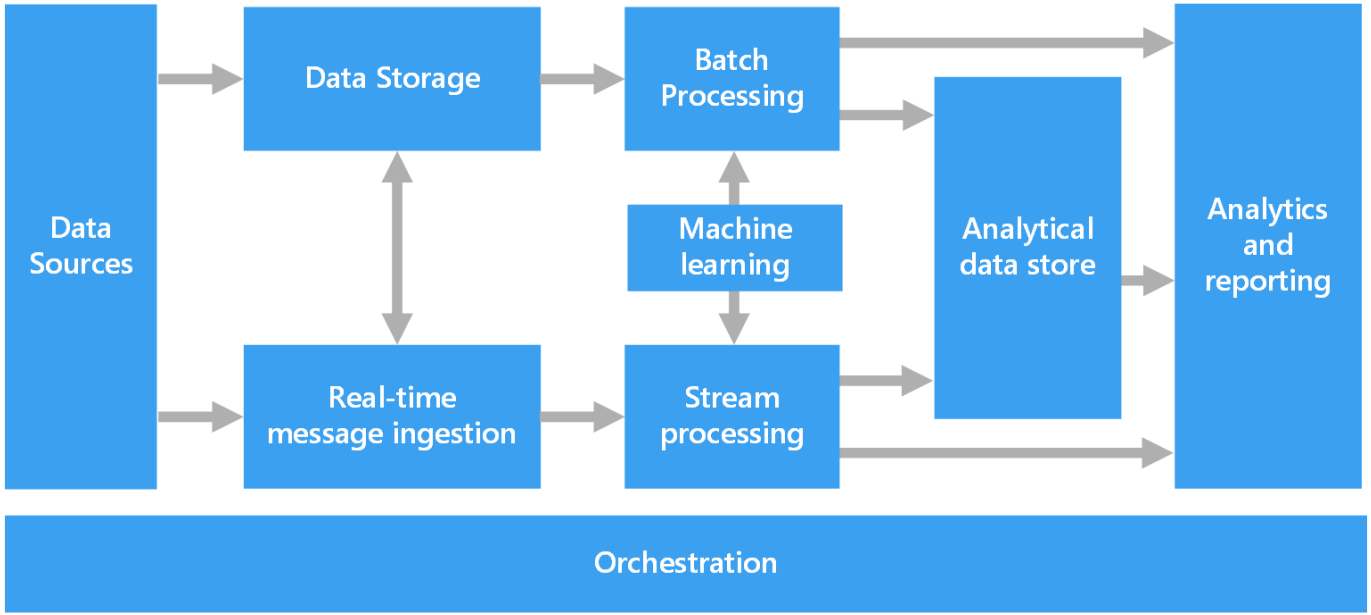
Scenario 2

- Company: Craft Company
- Business: Sells to craft shows In different locations
- Need:
  - Process complex queries from their massive repository of data.
  - Use these complex queries to influence business decisions and determine new opportunities
  - Cost is secondary to answers

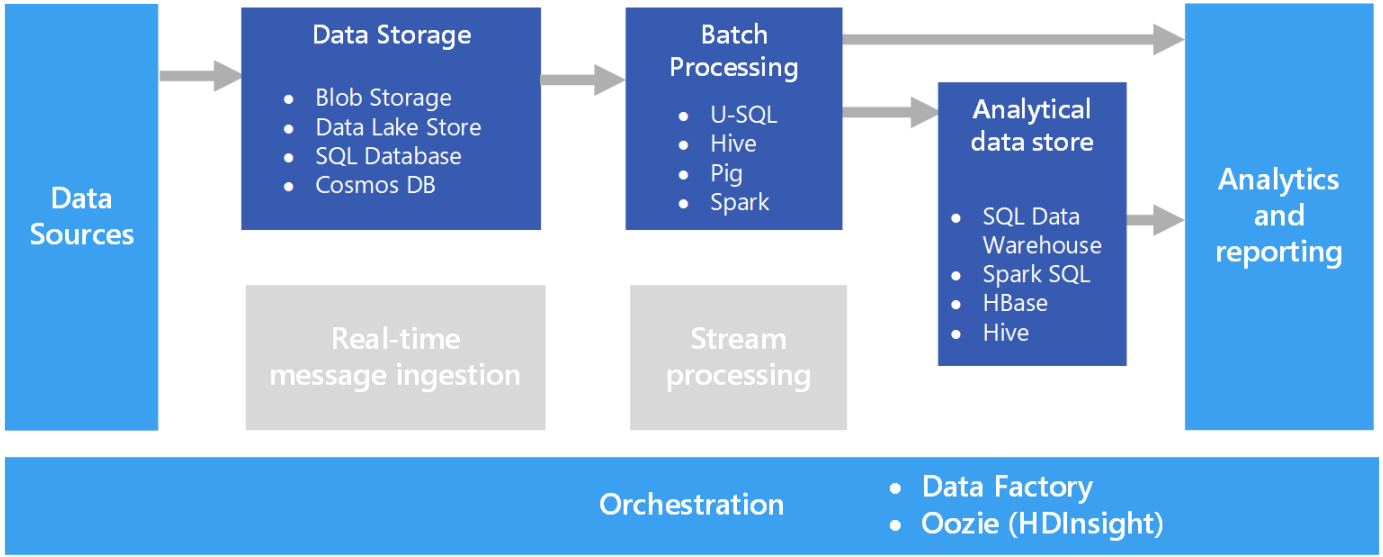
- Options
  - Azure SQL Database – Single
  - Azure SQL Database – Elastic Pool
  - Azure SQL Database – Managed Instance
  - Azure SQL Datawarehouse


Batch
Processing

Big Data Architecture



Batch Processing



Batch Processing:

- Data at rest
- Operate on very large dataset
- Computation takes significant time

Use Cases:

- Example – Web Server logs to Report

Challenges:

- Data Format and encoding
- Orchestration time slices

Azure Databricks

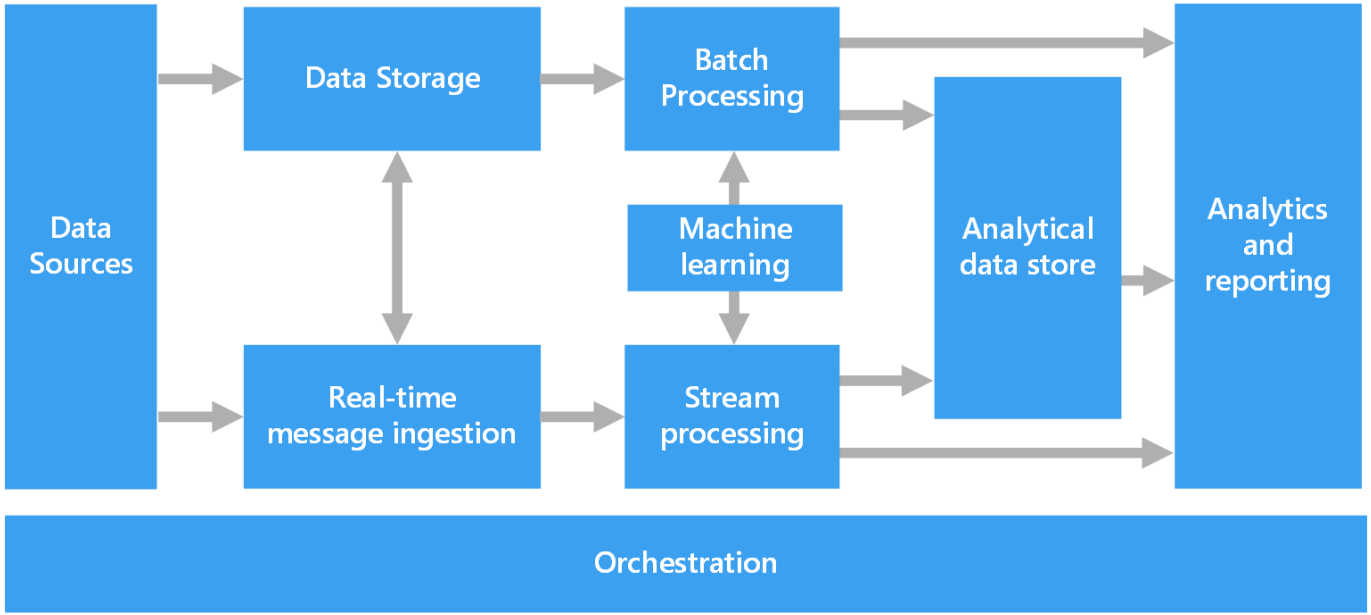
Azure Databricks

- **Azure Databricks is an Apache Spark-based analytics platform**
- **think of it as "Spark as a service."**

Features:

- **Languages: R, Python, Java, Scala, Spark SQL**
- **Fast cluster start times, auto-termination, autoscaling.**
- **Manages the Spark cluster for you.**
- **Built-in integration with Azure Blob Storage, Azure Data Lake Storage (ADLS), Azure Synapse, and other services.**
- **User authentication with Azure Active Directory.**
- **Web-based notebooks for collaboration and data exploration.**
- **Supports GPU-enabled clusters**

Big Data Architecture



Data Pipeline Orchestration

Pipeline Orchestration options:

- Azure Data Factory
- Oozie on HDInsight
- SQL Server Integration Services (SSIS)

Key Selection Criteria:

- Do you need big data capabilities for moving and transforming your data?
- Do you require a managed service that can operate at scale?
- Are some of your data sources located on-premises?
- Is your source data stored in Blob storage on an HDFS filesystem?

Data Pipeline Orchestration

General capabilities

Capability	Azure Data Factory	SQL Server Integration Services (SSIS)	Oozie on HDInsight
Managed	Yes	No	Yes
Cloud-based	Yes	No (local)	Yes
Prerequisite	Azure Subscription	SQL Server	Azure Subscription, HDInsight cluster
Management tools	Azure Portal, PowerShell, CLI, .NET SDK	SSMS, PowerShell	Bash shell, Oozie REST API, Oozie web UI
Pricing	Pay per usage	Licensing / pay for features	No additional charge on top of running the HDInsight cluster

Data Pipeline Orchestration

Pipeline capabilities

Capability	Azure Data Factory	SQL Server Integration Services (SSIS)	Oozie on HDInsight
Copy data	Yes	Yes	Yes
Custom transformations	Yes	Yes	Yes (MapReduce, Pig, and Hive jobs)
Azure Machine Learning scoring	Yes	Yes (with scripting)	No
HDInsight On-Demand	Yes	No	No
Azure Batch	Yes	No	No
Pig, Hive, MapReduce	Yes	No	Yes
Spark	Yes	No	No
Execute SSIS Package	Yes	Yes	No
Control flow	Yes	Yes	Yes
Access on-premises data	Yes	Yes	No

Data Pipeline Orchestration

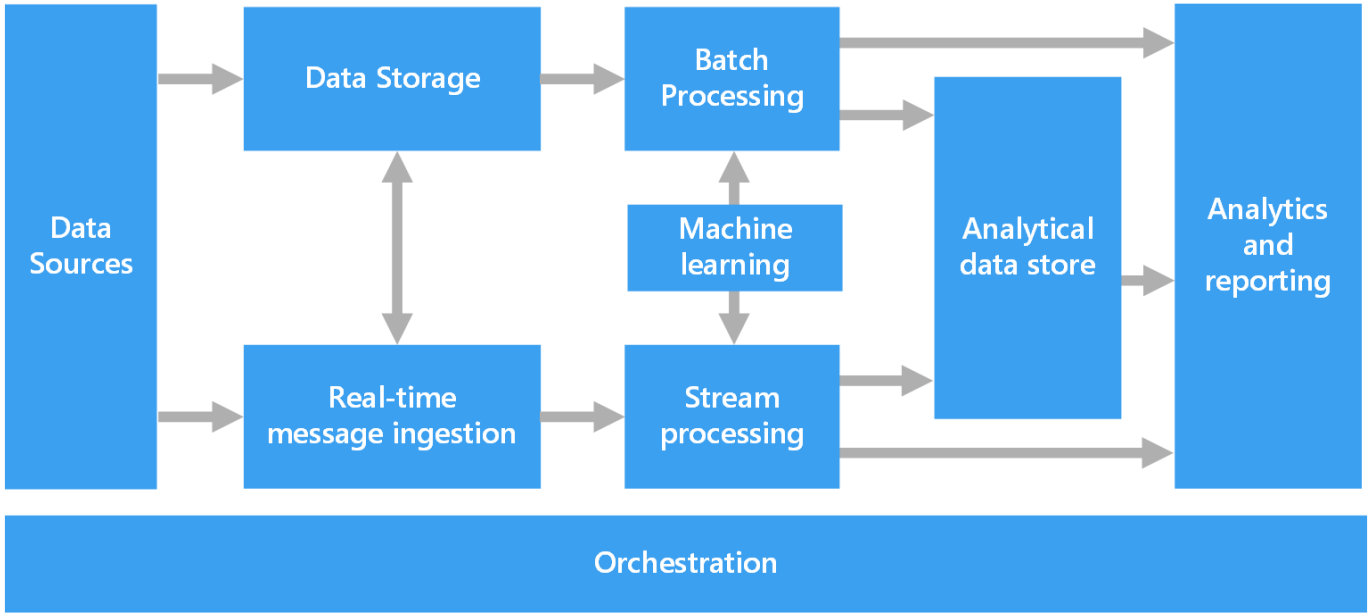
Scalability capabilities

Capability	Azure Data Factory	SQL Server Integration Services (SSIS)	Oozie on HDInsight
Scale up	Yes	No	No
Scale out	Yes	No	Yes (by adding worker nodes to cluster)
Optimized for big data	Yes	No	Yes

Data Ingestion methods

For Batch Processing solution

Big Data Architecture



Data Ingestion tools

Command line tools/APIs

- Azure CLI
- AzCopy
- PowerShell
- AdlCopy
- Distcp
- Sqoop
- PolyBase
- HadoopCommandline

Graphical Interface

- Azure Storage Explorer
- Azure portal

Data pipeline

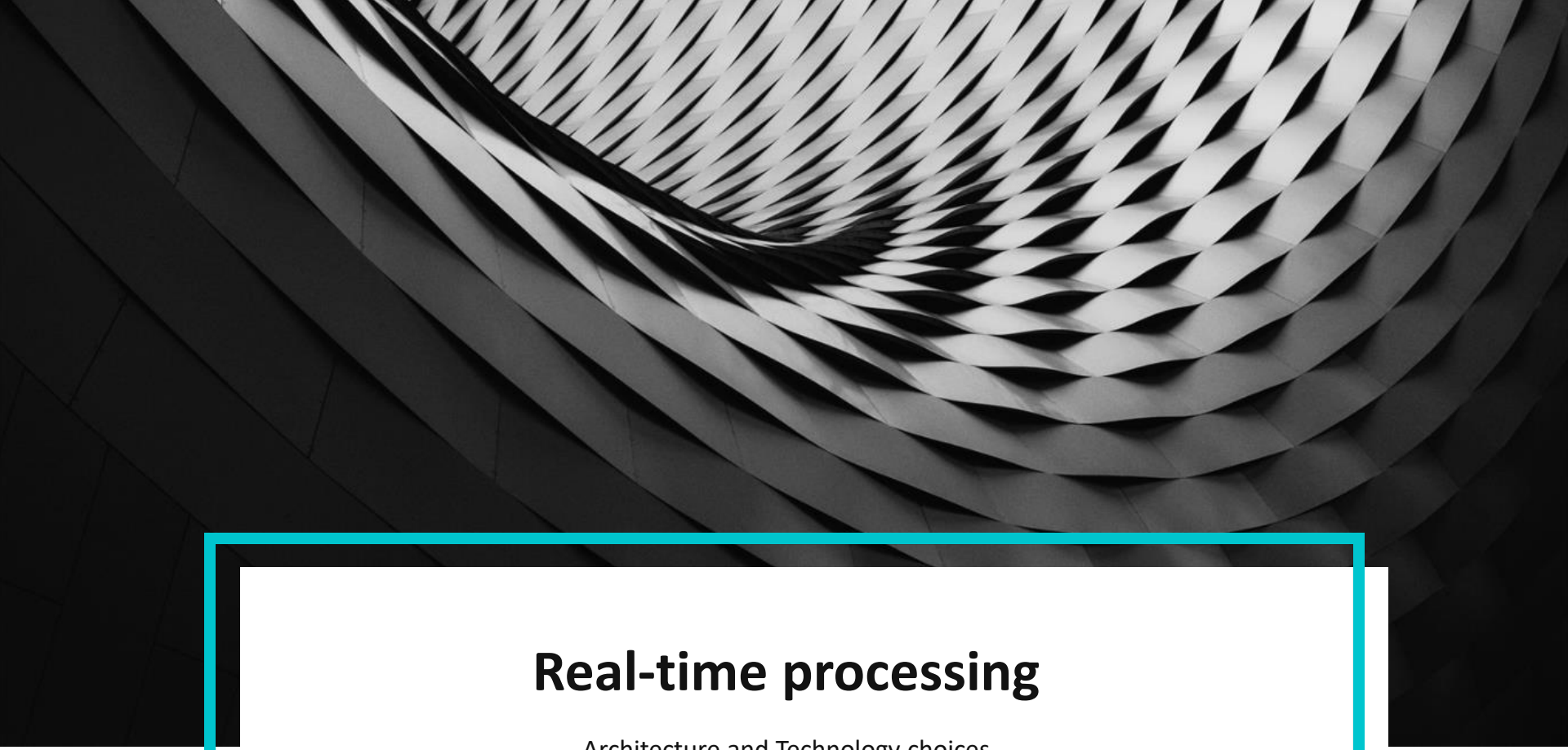
- Azure Data Factory

Command line tools comparison

Capability	Azure CLI	AzCopy	PowerShell	AdlCopy	PolyBase
Compatible platforms	Linux, OS X, Windows	Linux, Windows	Windows	Linux, OS X, Windows	SQL Server, Azure Synapse
Optimized for big data	No	Yes	No	Yes [1]	Yes [2]
Copy to relational database	No	No	No	No	Yes
Copy from relational database	No	No	No	No	Yes
Copy to Blob storage	Yes	Yes	Yes	No	Yes
Copy from Blob storage	Yes	Yes	Yes	Yes	Yes
Copy to Data Lake Store	No	Yes	Yes	Yes	Yes
Copy from Data Lake Store	No	No	Yes	Yes	Yes

Graphical interface and Azure Data Factory

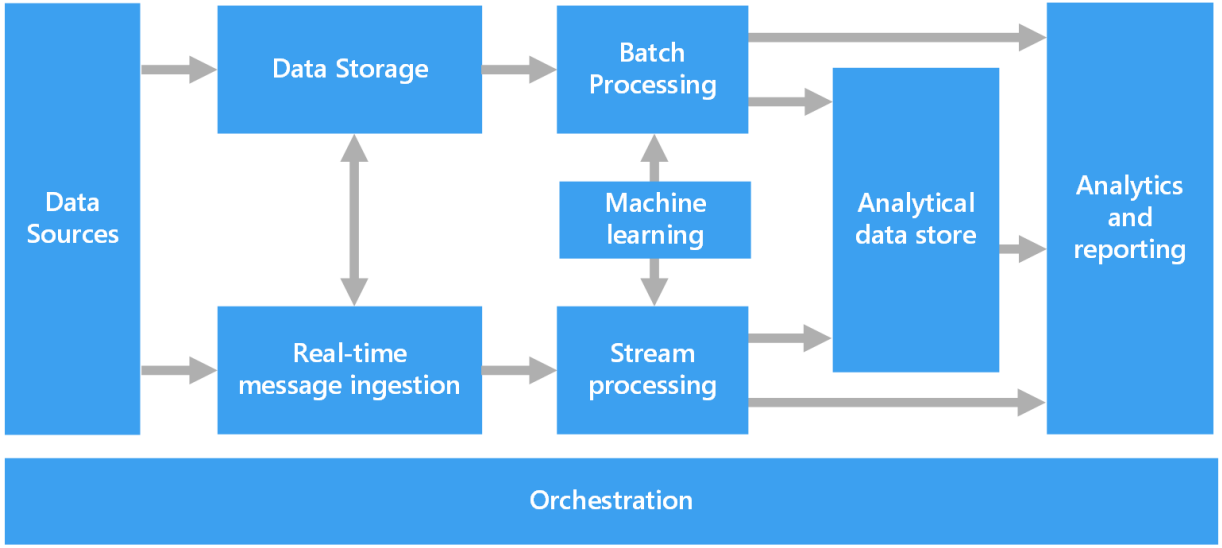
Capability	Azure Storage Explorer	Azure portal *	Azure Data Factory
Optimized for big data	No	No	Yes
Copy to relational database	No	No	Yes
Copy from relational database	No	No	Yes
Copy to Blob storage	Yes	No	Yes
Copy from Blob storage	Yes	No	Yes
Copy to Data Lake Store	No	No	Yes
Copy from Data Lake Store	No	No	Yes
Upload to Blob storage	Yes	Yes	Yes
Upload to Data Lake Store	Yes	Yes	Yes
Orchestrate data transfers	No	No	Yes
Custom data transformations	No	No	Yes
Pricing model	Free	Free	Pay per usage



Real-time processing

Architecture and Technology choices

Real Time Processing Architecture



Real time processing:

- Deals with streams of data that are captured in real-time
- Processed with minimal latency
- Incoming data typically arrives in an unstructured or semi-structured format, such as JSON
- Generate real-time (or near-real-time) reports or automated responses.

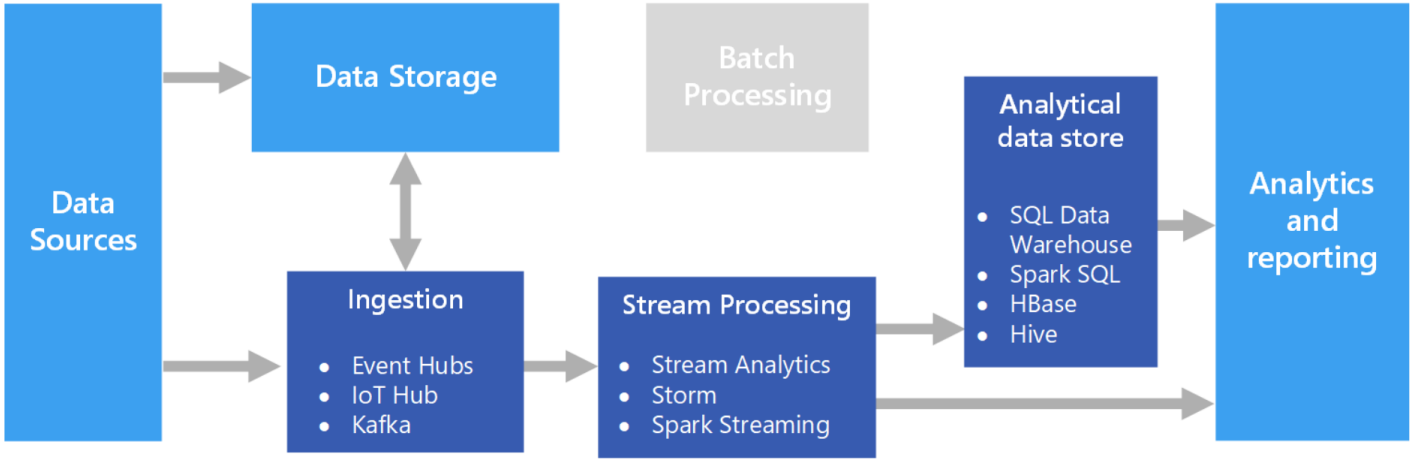
For example:

- Use sensor data to detect high traffic volumes

Challenges:

- Ingest, process, and store messages in real time, especially at high volumes

Real Time Processing Architecture



Real-time message ingestion:

Azure Event Hubs: Messaging solution for ingesting millions of event messages per second.

- Can be processed by multiple consumers in parallel
- natively supports AMQP (Advanced Message Queuing Protocol 1.0)

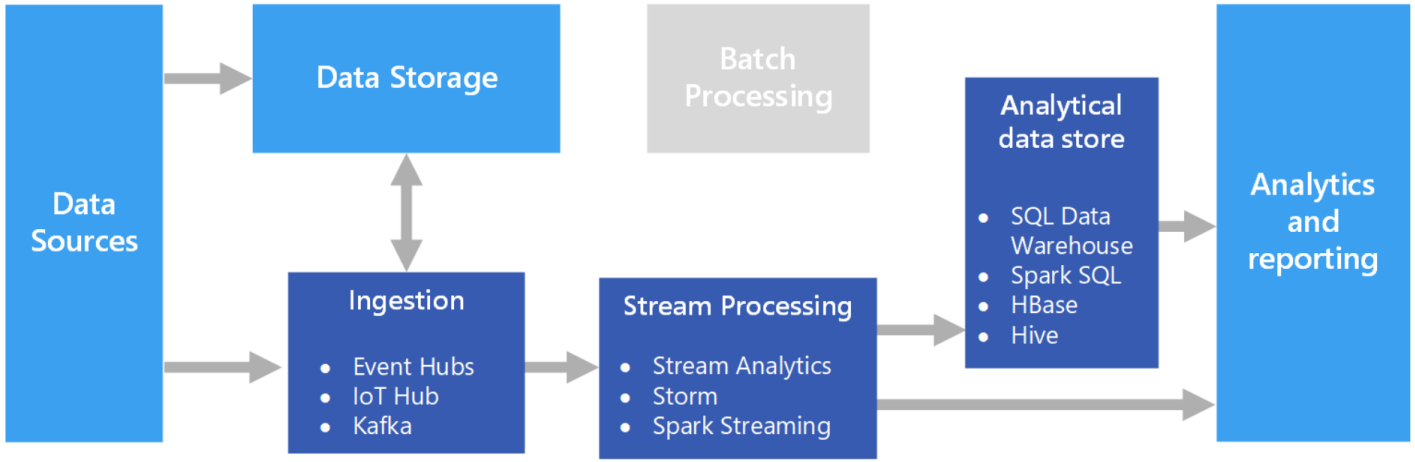
Azure IoT Hub: Provides bi-directional communication between Internet-connected devices

- Scalable message queue that can handle millions of simultaneously connected devices.

Apache Kafka: Open source message queuing and stream processing application

Azure Storage Blob Containers or Azure Data Lake Store: can be use for static reference data, or output destination for captured real-time data for archiving

Real Time Processing Architecture



Stream processing

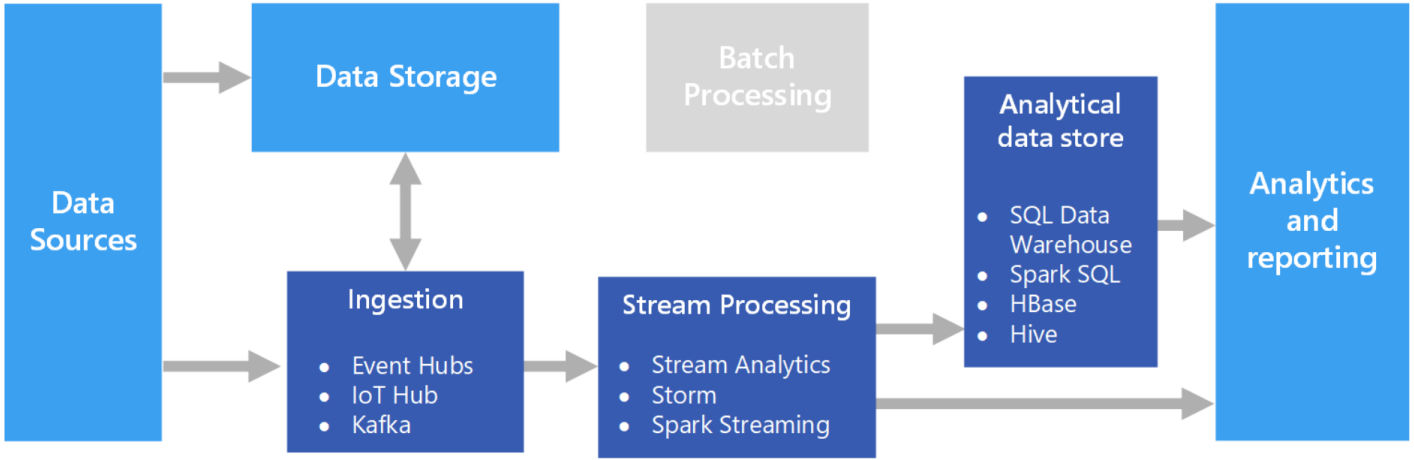
Azure Stream Analytics: can run perpetual queries against an unbounded stream of data

- Consume streams of data from storage or message brokers
- Filter and aggregate the data based on temporal windows
- Write the results to sinks such as storage, databases, or directly to reports in Power BI
- Uses a SQL-based query language

Storm: Open source framework that uses a topology of spouts and bolts to consume, process, and output the results

Spark Streaming (Databricks): Open source distributed platform for general data processing, supported Spark language, including Java, Scala, and Python

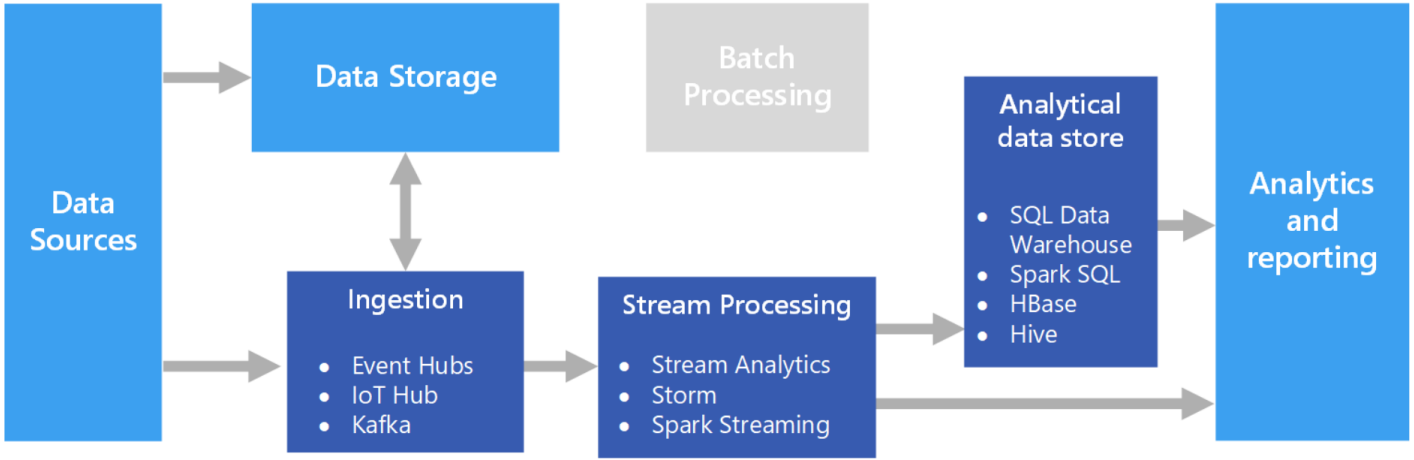
Real Time Processing Architecture



Analytical data store

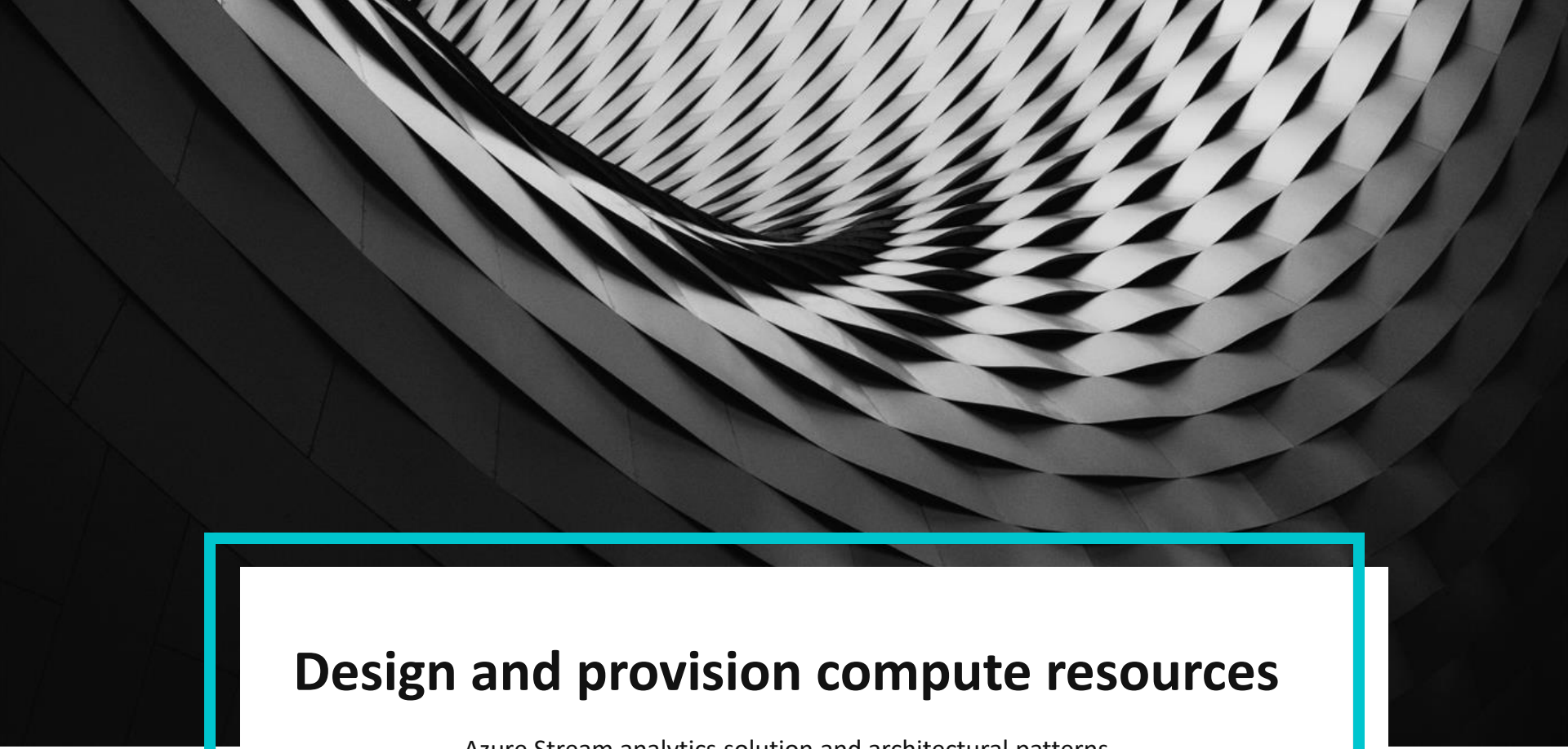
- **Azure Synapse Analytics:** Relational database
- **Hbase:** NoSQL Store
- **Spark/Hive:** files

Real Time Processing Architecture



Analytics and reporting

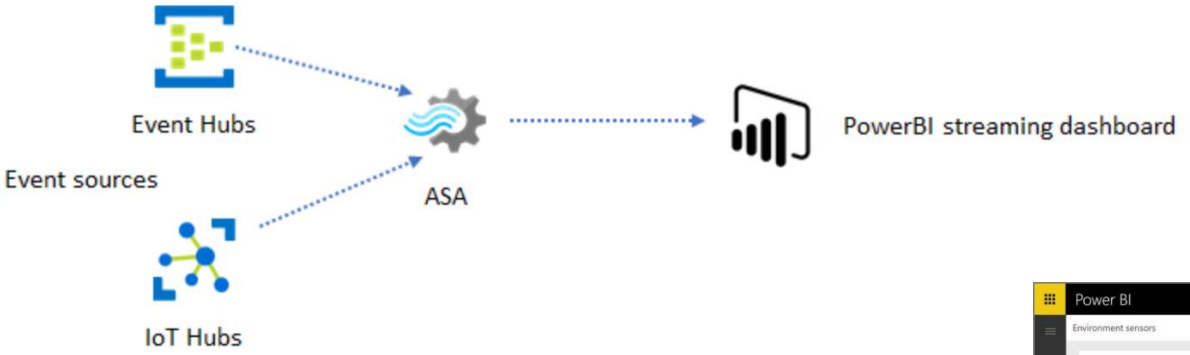
- **Azure Analysis Services**
- **Power BI**
- **Microsoft Excel**



Design and provision compute resources

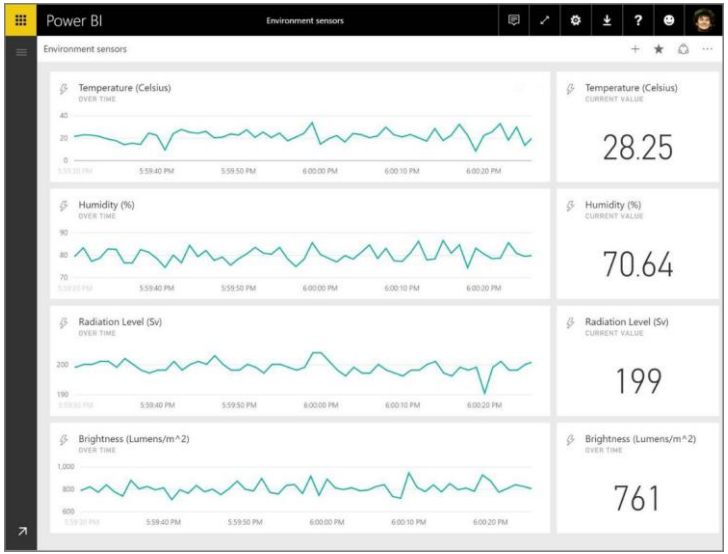
Azure Stream analytics solution and architectural patterns

Real time dashboard

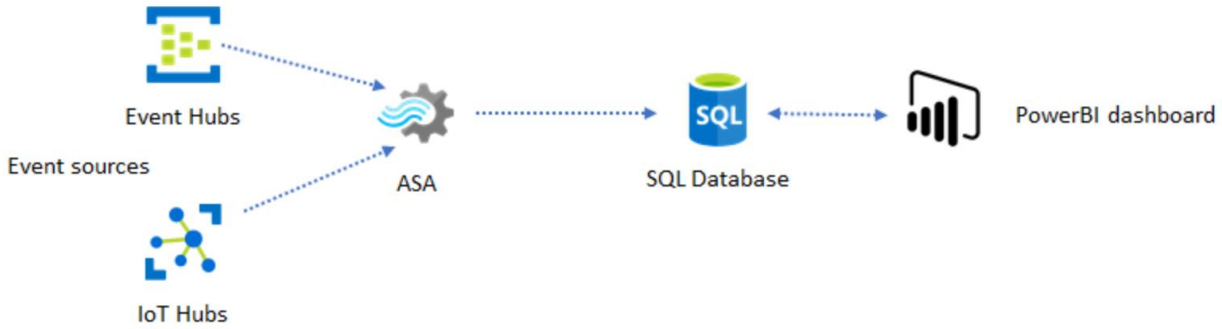


Streaming data can be:

- Factory sensors
- Social media sources
- Service usage metrics
- Or many other time-sensitive data collectors or transmitters

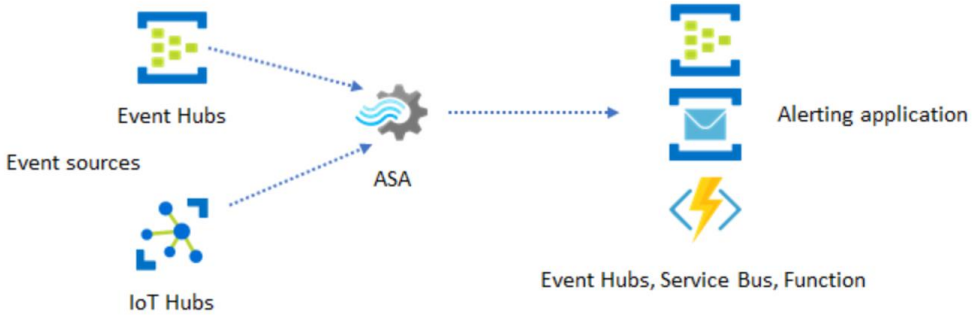


Use SQL for Dashboard



- More flexibility
- Slightly higher latency
- Maximize Power BI capabilities to further slice and dice the data for reports
- Flexibility of using other dashboard solutions, such as Tableau

Real-time insights into your application with event messaging



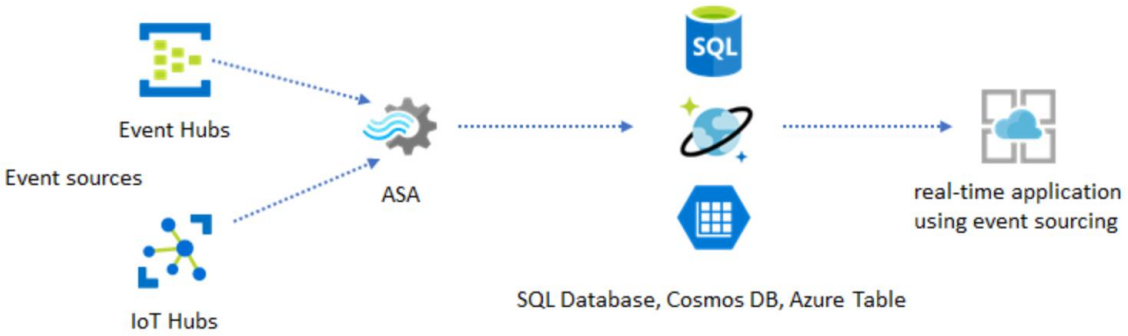
Why Azure Functions?

- Custom logic can also be implemented in Azure Functions
- Azure Functions also supports various types of notifications including text and email.

Why Event hubs?

- Most flexible integration point - Azure Data Explorer and Time Series Insights can consume events from Event Hubs
- Services can be connected directly to the Event Hubs sink from Azure Stream Analytics to complete the solution
- Event Hubs is also the highest throughput messaging broker available on Azure for such integration scenarios.

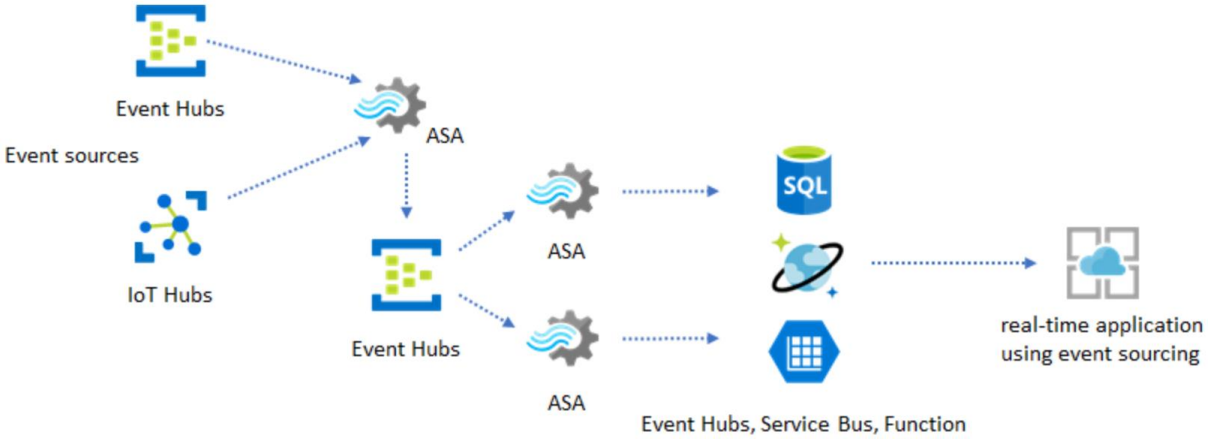
Real-time insights through data stores



Dataflow-based architecture

- Modern high-volume data driven applications often adopt a dataflow-based architecture
- Events are processed and aggregated into data stores by Azure Stream Analytics
- The application layer interacts with data stores using the traditional request/response pattern.

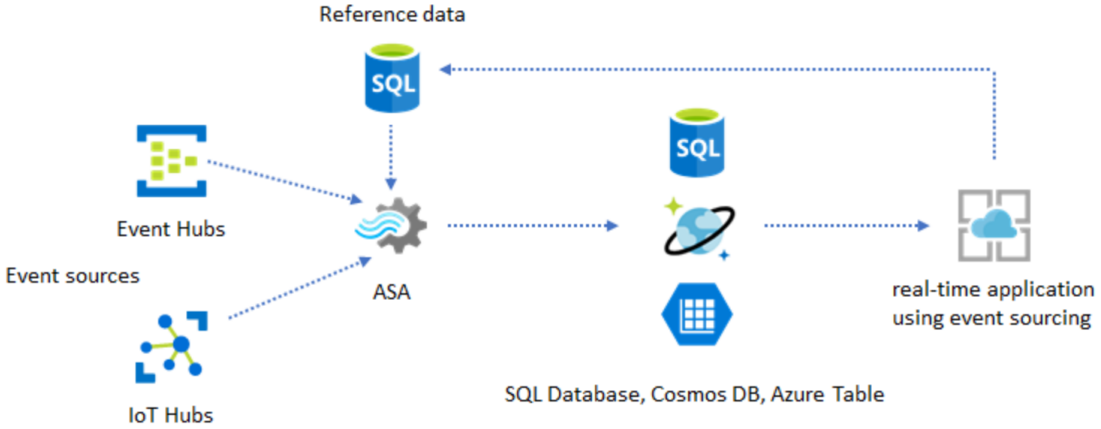
Real-time insights through data stores



Dataflow-based architecture

- Modern high-volume data driven applications often adopt a dataflow-based architecture
- Events are processed and aggregated into data stores by Azure Stream Analytics
- The application layer interacts with data stores using the traditional request/response pattern.

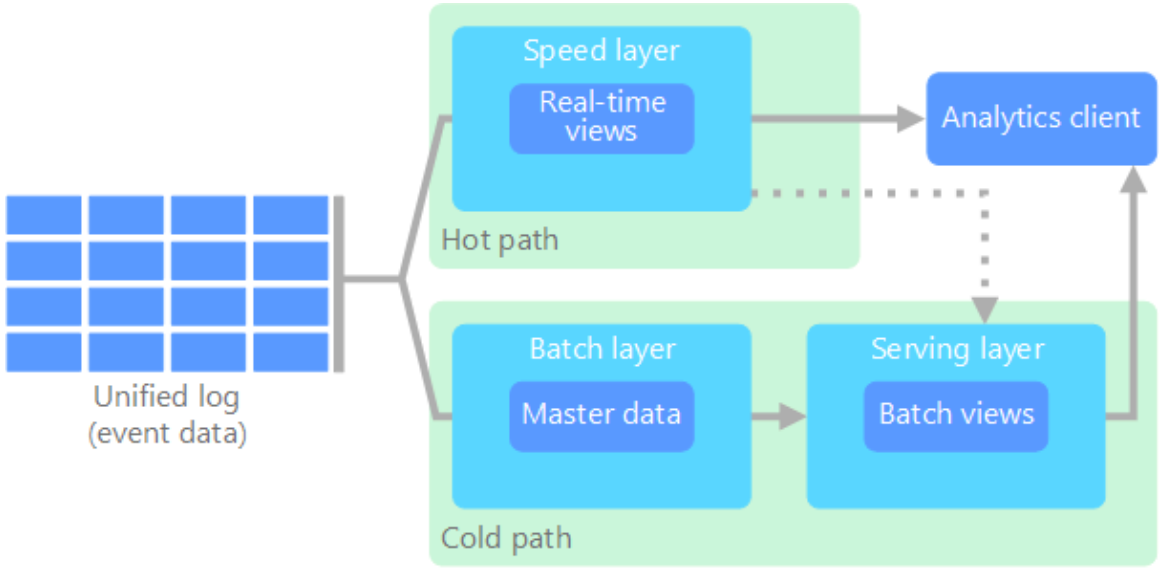
Reference data for application customization



Reference data

- Reference data feature is designed specifically for end-user customization like alerting threshold and processing rules
- Reference data (also known as a lookup table) is a finite data set that is static or slowly changing in nature

Lambda architecture



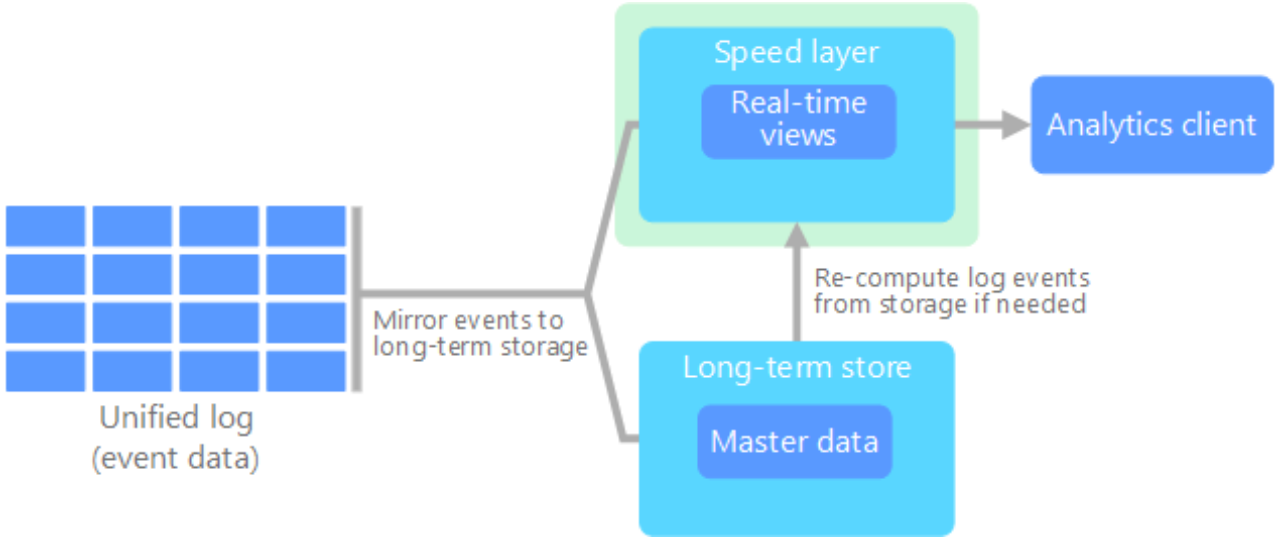
Batch layer (cold path)

- Stores all of the incoming data in its raw form Performs batch processing on the data.
- Result stored as a batch view.

Speed layer (hot path)

- Analyzes data in real time
- Designed for low latency
- Low accuracy.

Kappa architecture

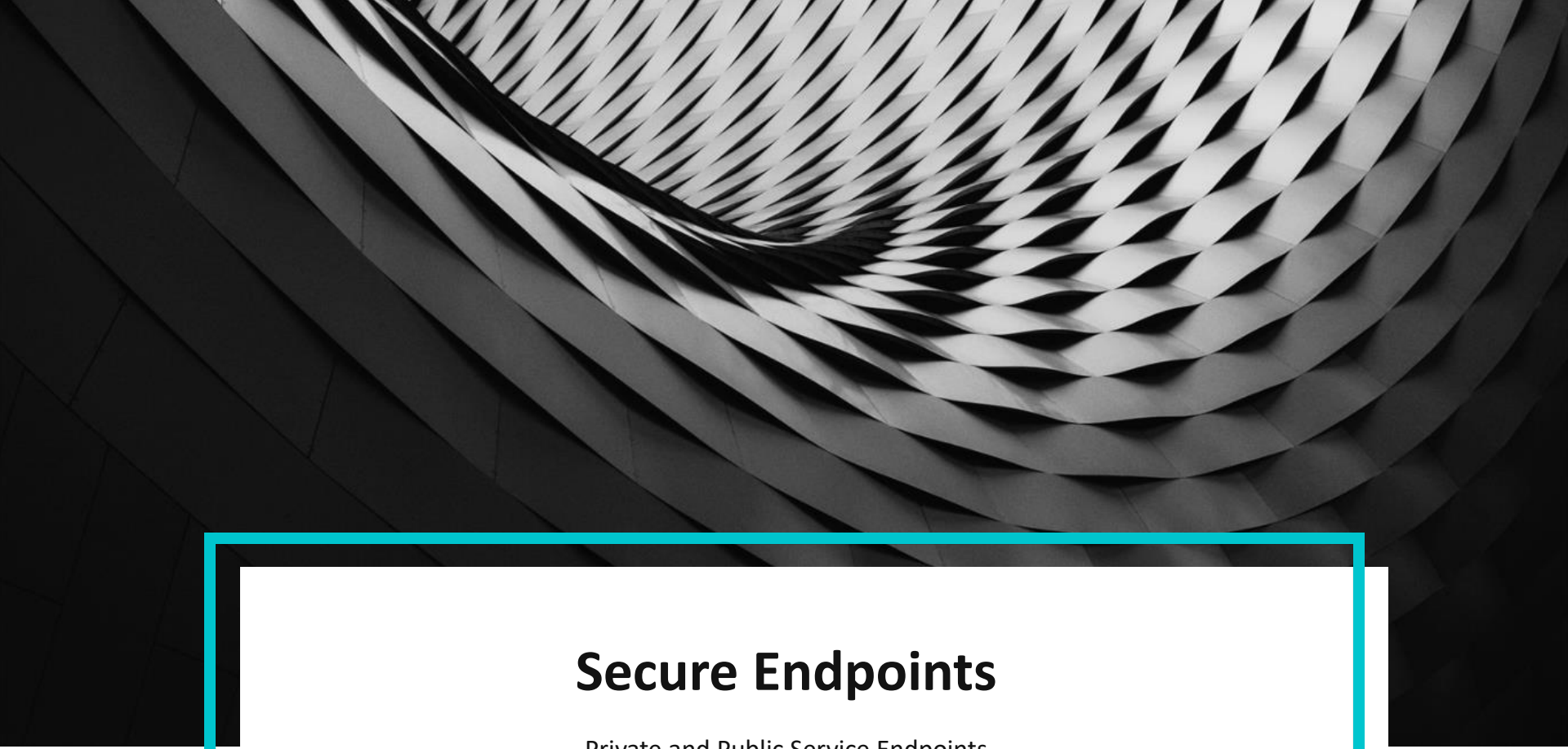


Batch layer (cold path)

- Stores all of the incoming data in its raw form Performs batch processing on the data.
- Result stored as a batch view.

Speed layer (hot path)

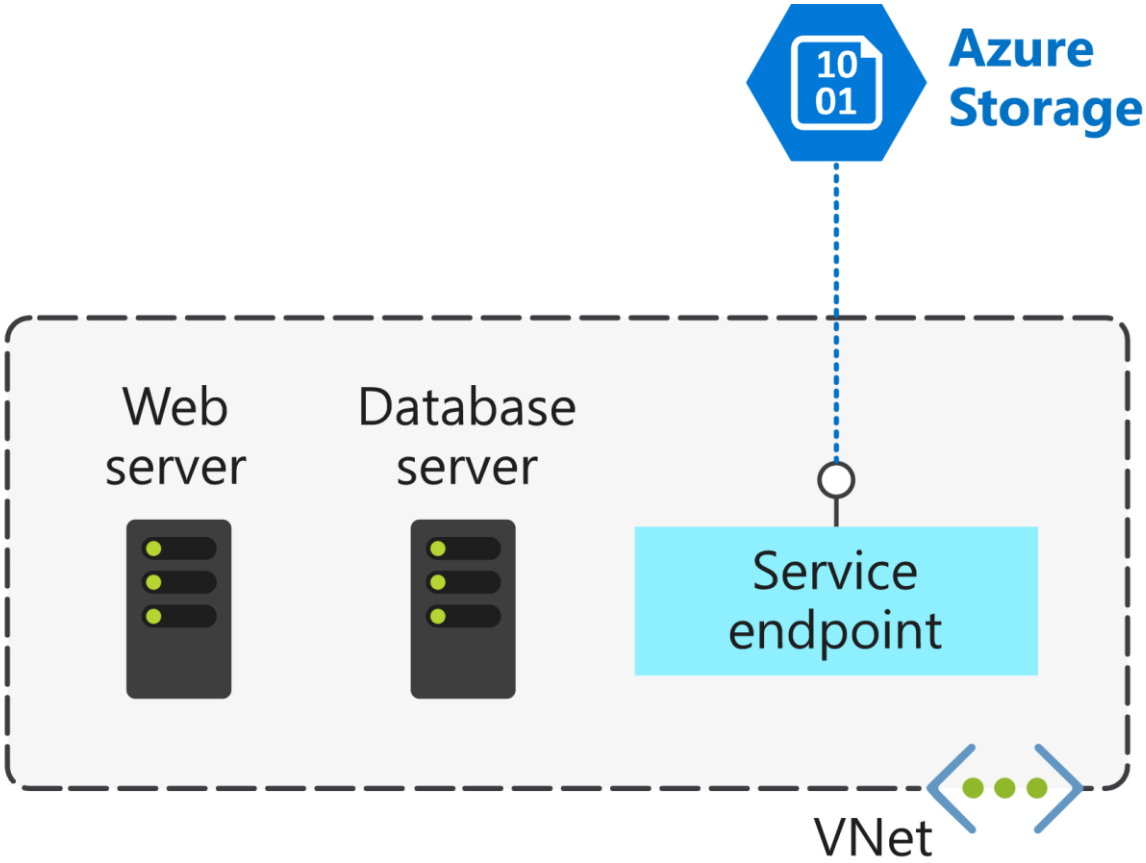
- Analyzes data in real time
- Designed for low latency
- Low accuracy.



Secure Endpoints

Private and Public Service Endpoints

Virtual network service endpoints



Service endpoints