Essentials of Azure Data Lake Storage Gen 2

Data Saturday Holland
October 5, 2019

Melissa Coates

Hello! A Little About Me…

Data Architect

Consultant

Microsoft MVP

Technical Trainer

Melissa Coates

@SQLChick | @CoatesDS

www.CoatesDataStrategies.com

What You'll Learn About Today

1. Overview & Objectives of a Data Lake
2. Azure Storage Primer
3. ADLS Gen 2 Technical Overview
4. ADLS Gen 2 Integration with Azure Services
5. ADLS Gen 2 Current State & Roadmap

Do you have a data lake now?

-Evaluating or learning?

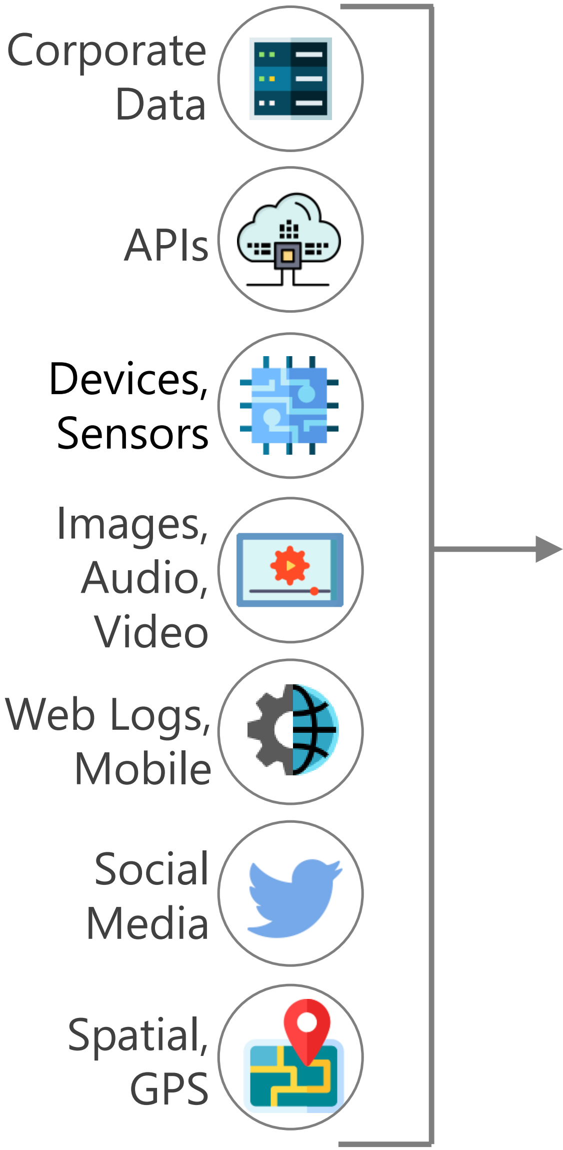
-In dev or test?

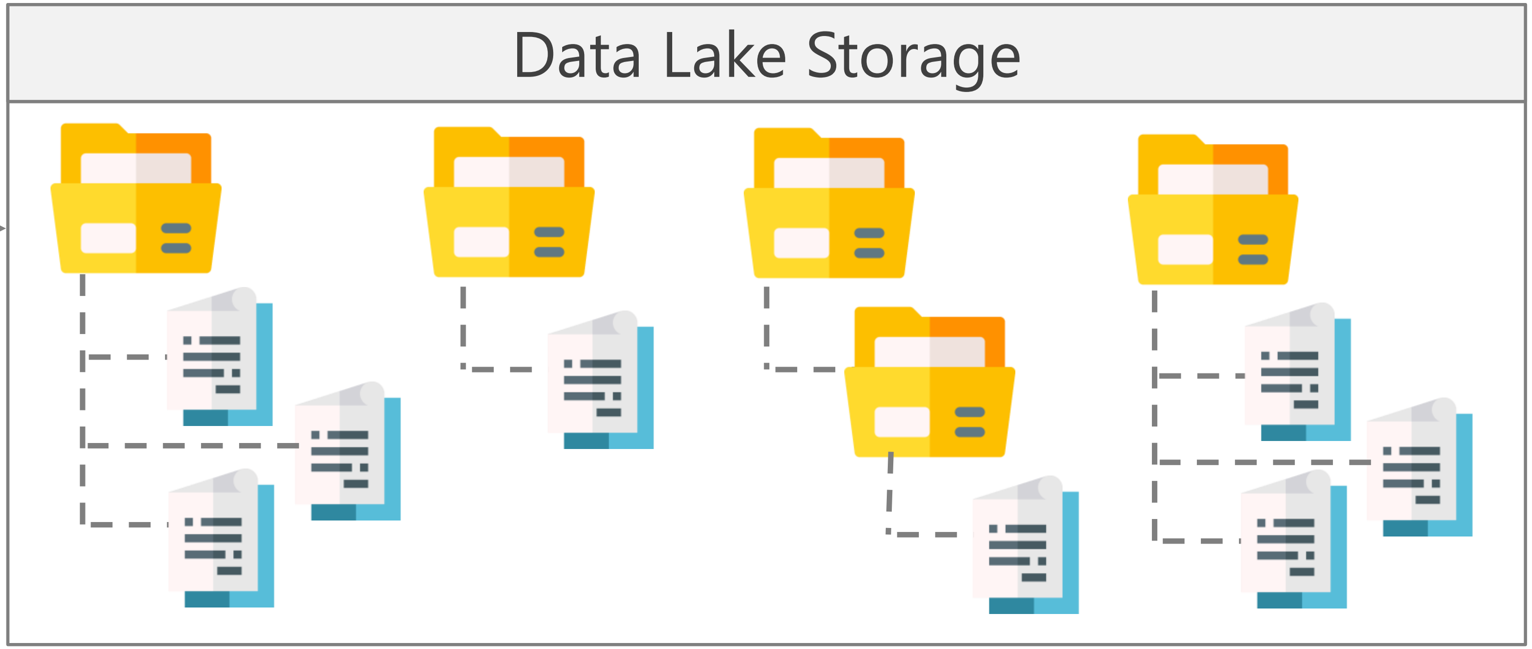
-In production?

Overview and Objectives of a Data Lake

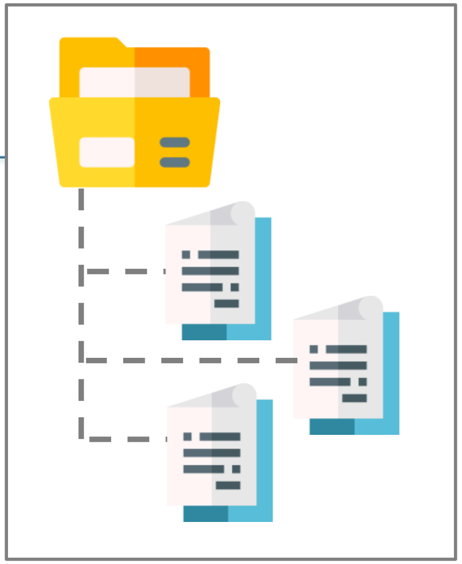
What is a Data Lake?

A repository for storing large quantities of disparate sources of data in any format





Objectives of a Data Lake



✓ Reduce upfront effort to ingest data

✓ Defer work to ‘schematize’ until value is known

✓ Store low latency data & new data types

✓Facilitate advanced analytics scenarios & new use cases

✓ Store large volumes of data cost efficiently

Ways Data Lakes are Commonly Used

Complement Data Warehouse

Staging

Active archive

Federated queries

Access to non-relational data

Analytics

Data exploration

Data science experimentation

Machine learning

Self-Service

Sandbox

Citizen data scientists

Data preparation

Key Characteristics of a Data Lake

Scalable

Linear growth-on demand, petabyte-scale with high throughput

Cost-Effective

Cloud economic model

Flexible Integration

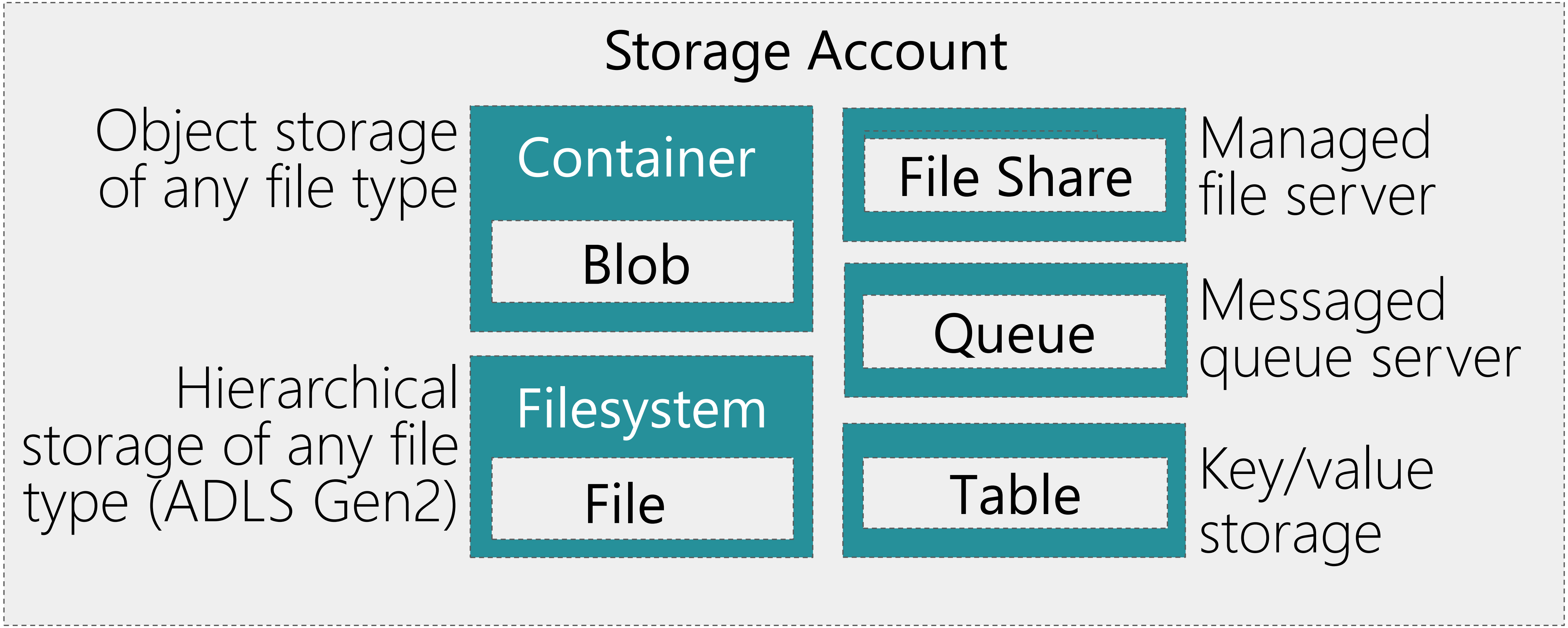
Supports multiple tools, methods, and patterns for data ingress, egress & processing

Granular Security

Several aspects of data protection

Azure Storage Primer

Azure Storage Options



Storage Account

Object storage of any file type

Container

Blob

Hierarchical storage of any file type (ADLS Gen2)

Filesystem

File

File Share

Managed file server

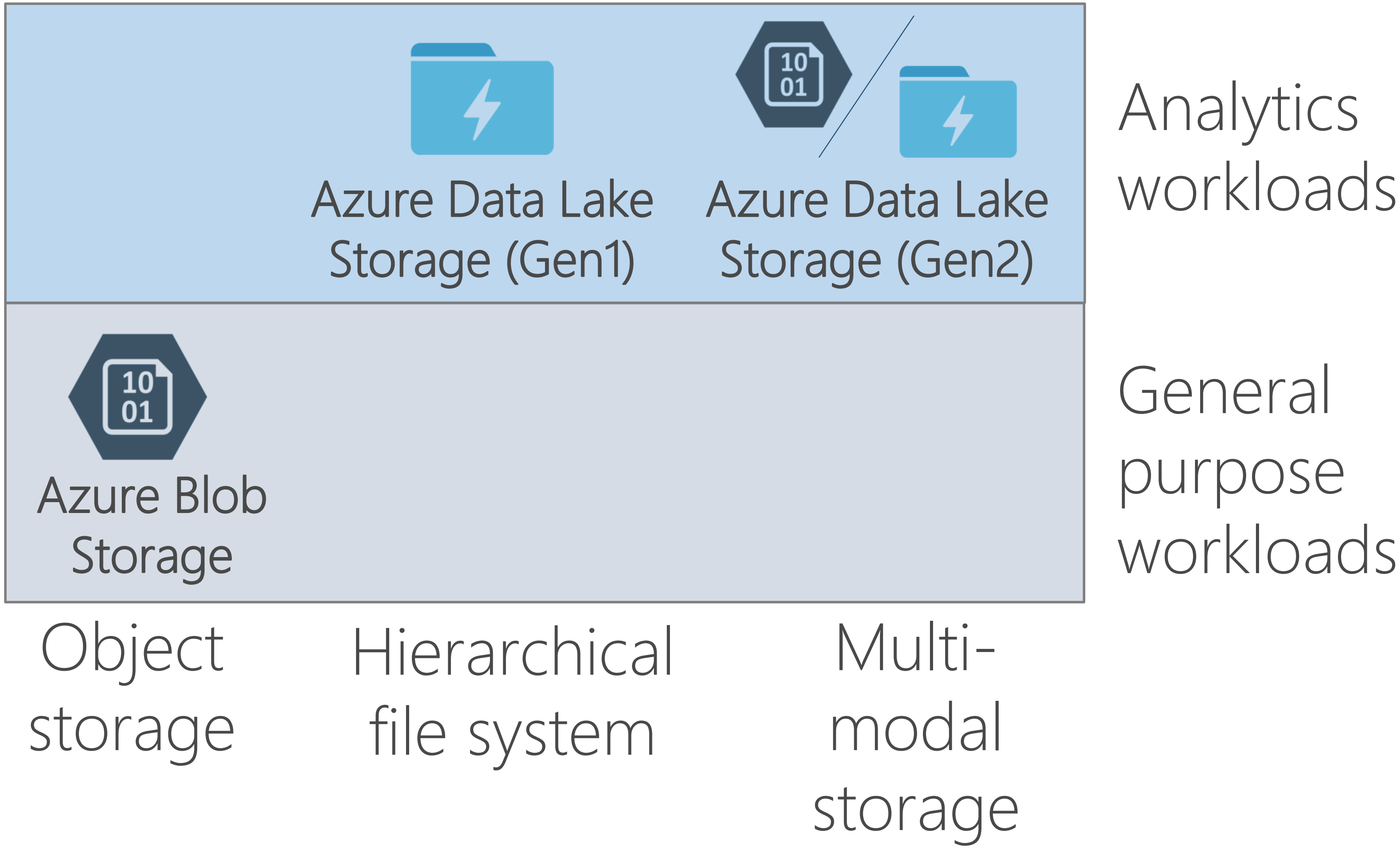
Queue

Messaged queue server

Table

Key/value storage

Data Lake Services in Azure



	Object storage	Hierarchical file system	Multi-modal storage
Analytics workloads		Azure Data Lake Storage (Gen1)	Azure Data Lake Storage (Gen2)
General purpose workloads	Azure Blob Storage		

Azure Blob Storage

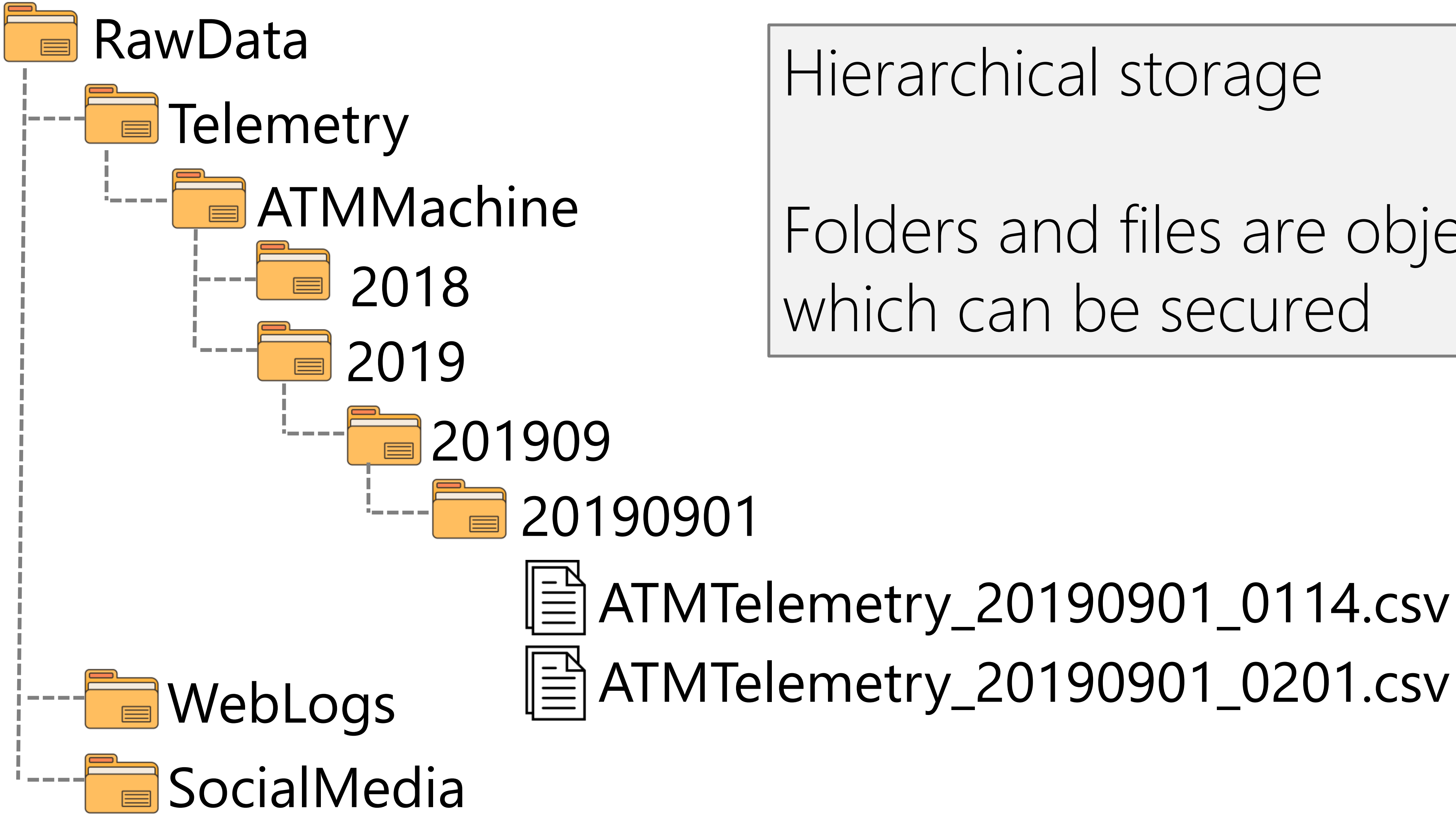
RawData/Telemetry/ATMMachine/2019/201909/20190901/ATMTelemetry_20190901_0114.csv

RawData/Telemetry/ATMMachine/2019/201909/20190901/ATMTelemetry_20190901_0201.csv

Object storage: a flat namespace

Folders and files are virtual objects

Azure Data Lake Storage Gen 1

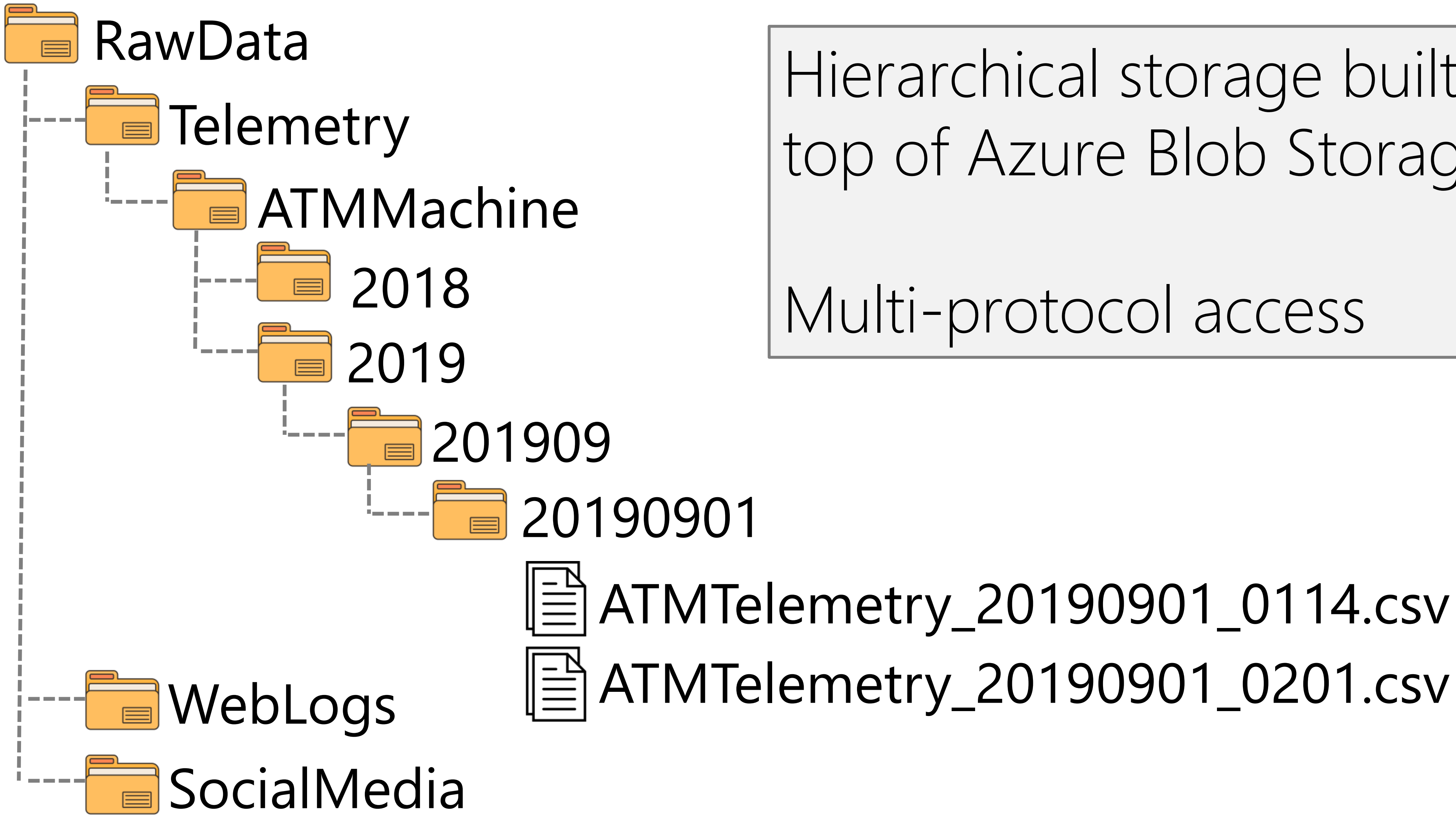


- RawData
  - Telemetry
    - ATMMachine
      - 2018
      - 2019
        - 201909
          - 20190901
            - ATMTelemetry_20190901_0114.csv
            - ATMTelemetry_20190901_0201.csv
  - WebLogs
  - SocialMedia

Hierarchical storage

Folders and files are objects which can be secured

Azure Data Lake Storage Gen 2



RawData

Telemetry

ATMMachine

2018

2019

201909

20190901

ATMTelemetry_20190901_0114.csv

ATMTelemetry_20190901_0201.csv

WebLogs

SocialMedia

Hierarchical storage built on top of Azure Blob Storage

Multi-protocol access

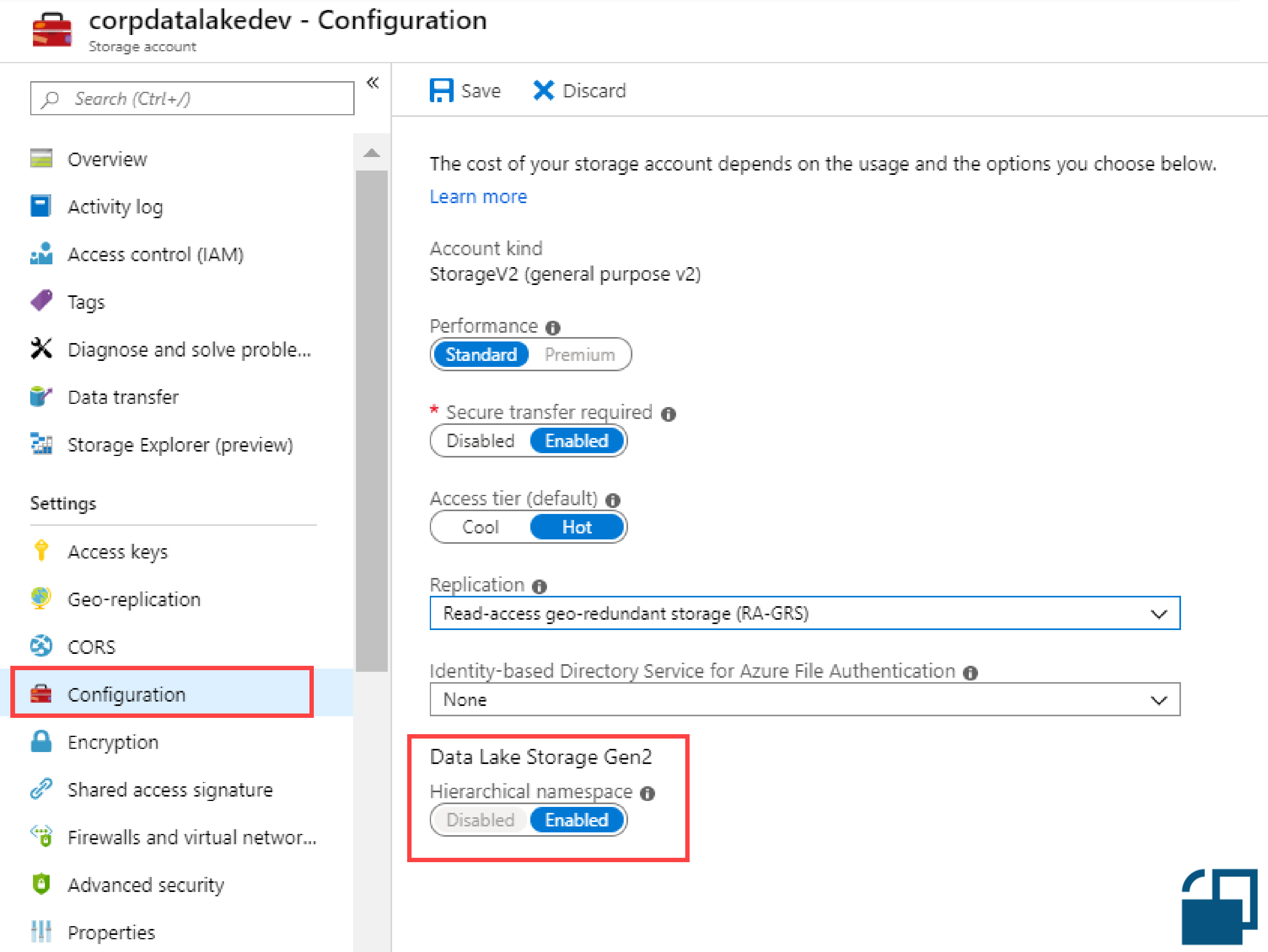
Azure Data Lake Storage Gen 2: Technical Overview

Hierarchical Namespace Enabled

ADLS Gen 2 =

An Azure Storage account with the hierarchical namespace enabled.

ADLS Gen 2 is not a separate Azure service like ADLS Gen 1.



The cost of your storage account depends on the usage and the options you choose below. Learn more

Account kind
StorageV2 (general purpose v2)

Performance
[✓ Standard] Premium

* Secure transfer required
Disabled [✓ Enabled]

Access tier (default)
Cool [✓ Hot]

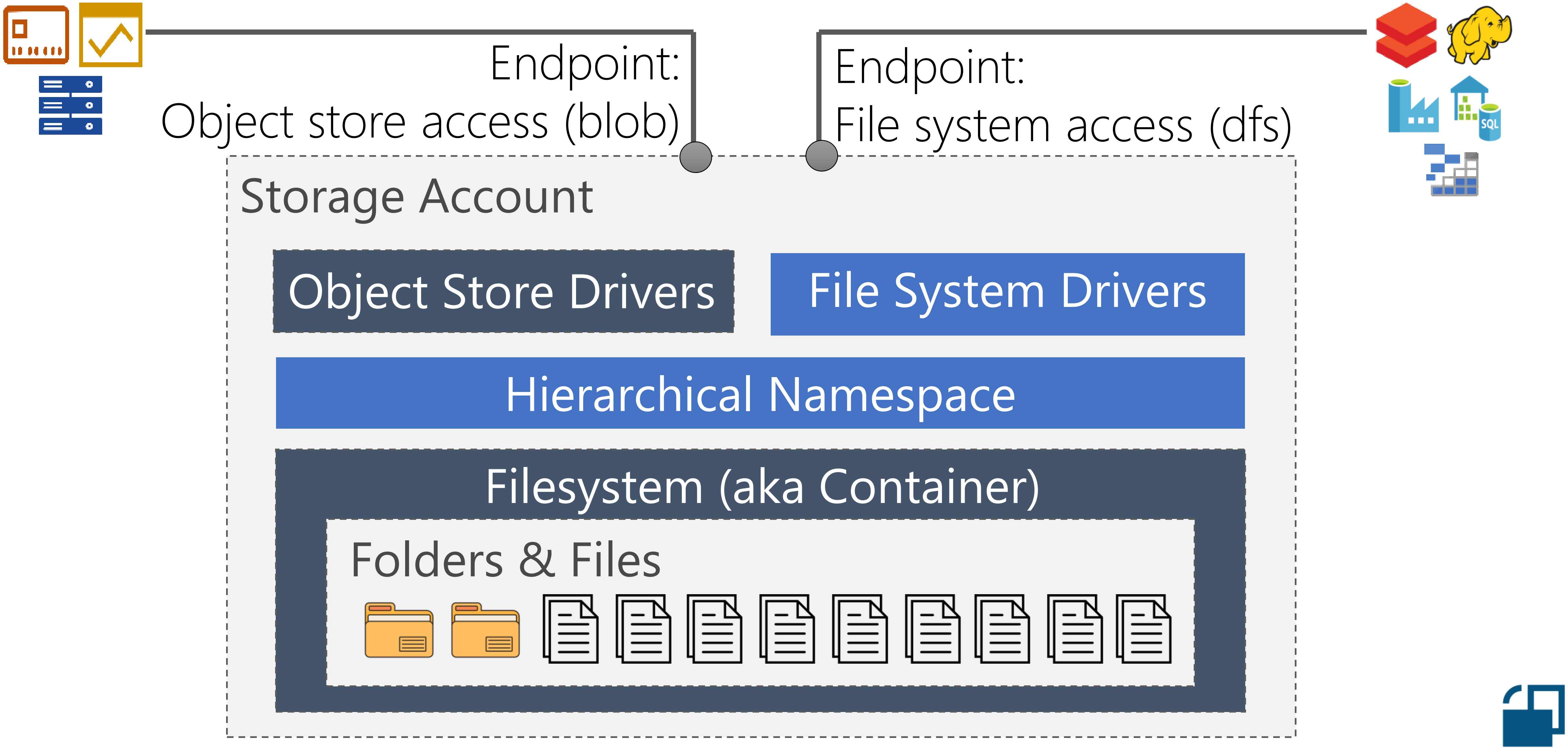
Replication
Read-access geo-redundant storage (RA-GRS)

Identity-based Directory Service for Azure File Authentication
None

Data Lake Storage Gen2

Hierarchical namespace
Disabled [✓ Enabled]

Components of ADLS Gen 2



Endpoint:
Object store access (blob)

Endpoint:
File system access (dfs)

Storage Account

Object Store Drivers

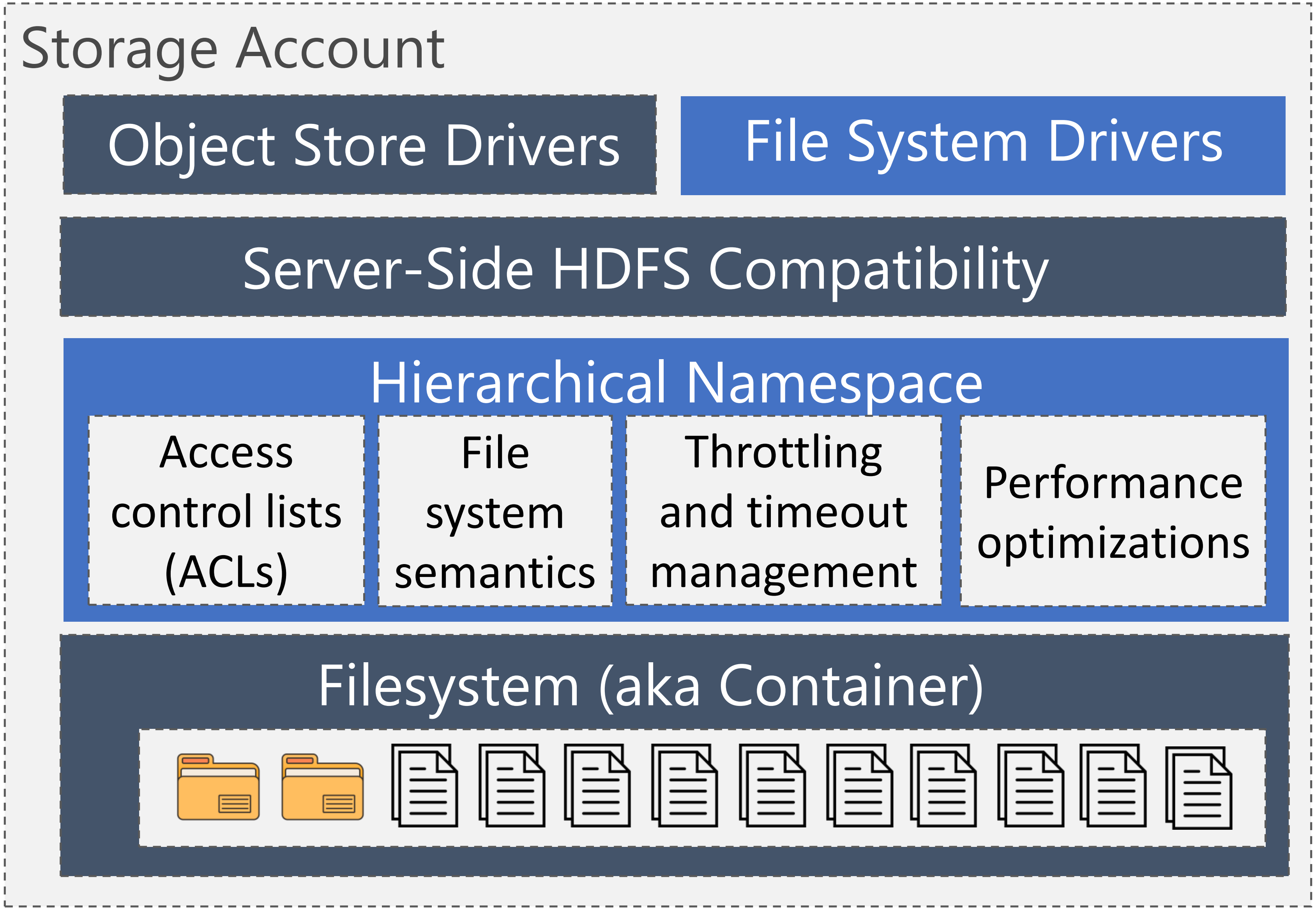
File System Drivers

Hierarchical Namespace

Filesystem (aka Container)

Folders & Files

Hierarchical Namespace



Storage Account

Object Store Drivers

File System Drivers

Server-Side HDFS Compatibility

Hierarchical Namespace

Access control lists (ACLs)

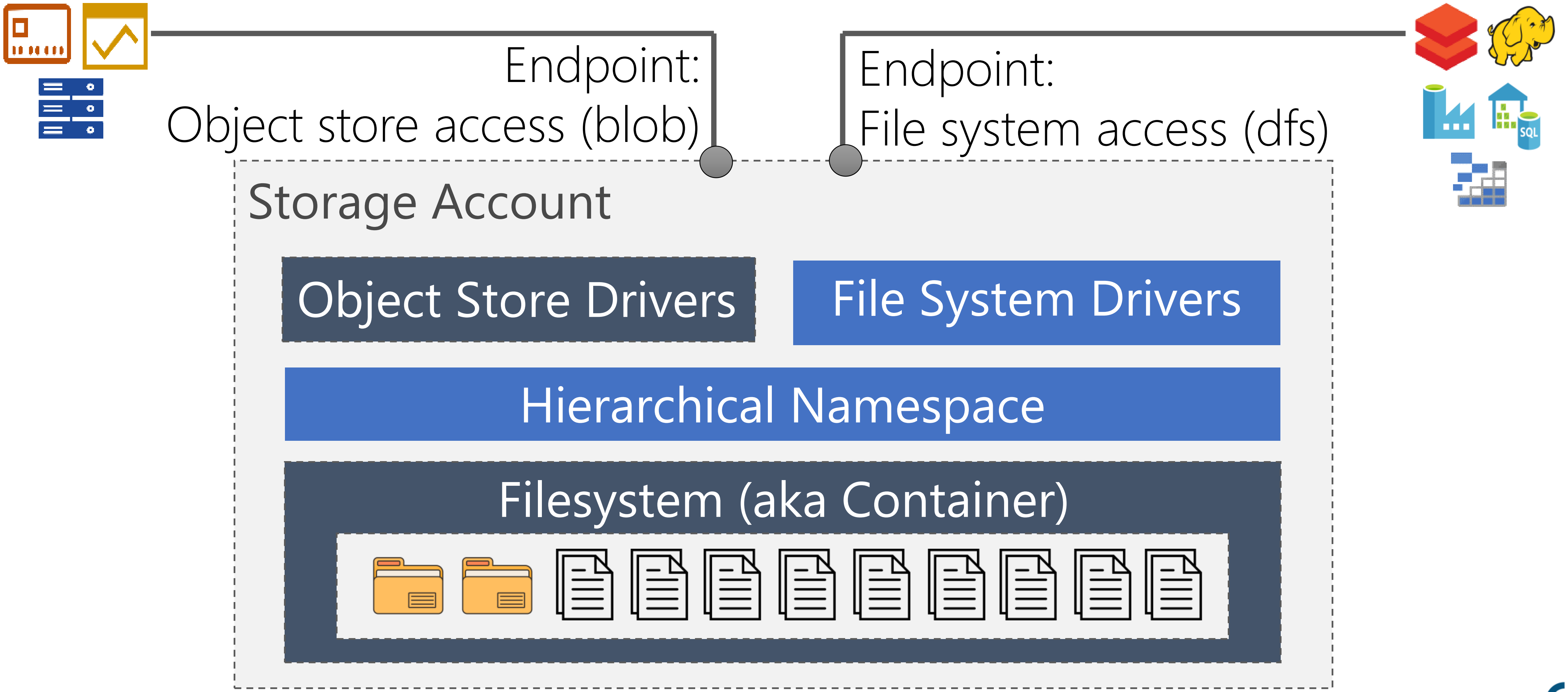
File system semantics

Throttling and timeout management

Performance optimizations

Filesystem (aka Container)

Multi-Protocol Access (MPA)



Endpoint:
Object store access (blob)

Endpoint:
File system access (dfs)

Storage Account

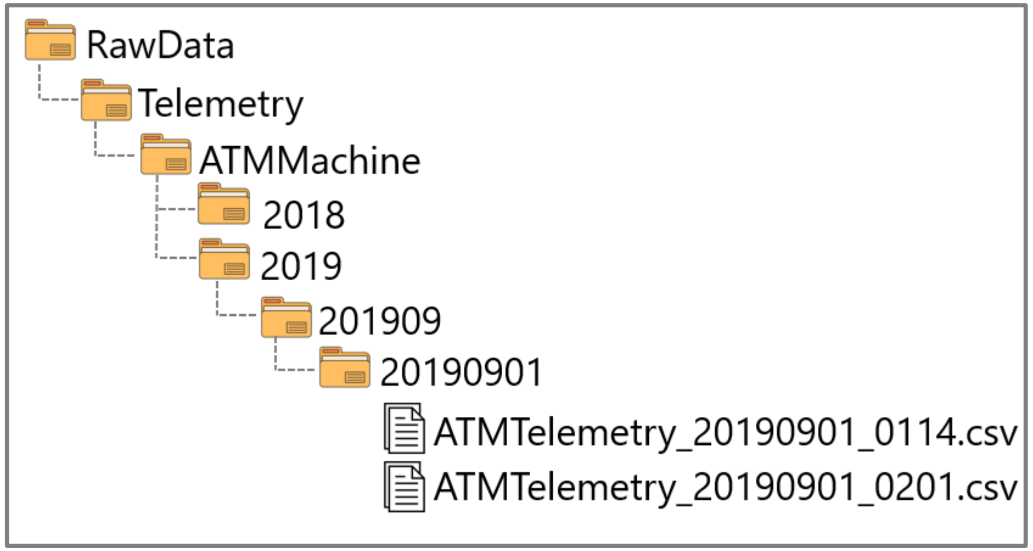
Object Store Drivers

File System Drivers

Hierarchical Namespace

Filesystem (aka Container)

Connectivity Option 1: File System Endpoint



```
RawData
  Telemetry
    ATMMachine
      2018
      2019
        201909
          20190901
            ATMTelemetry_20190901_0114.csv
            ATMTelemetry_20190901_0201.csv
```

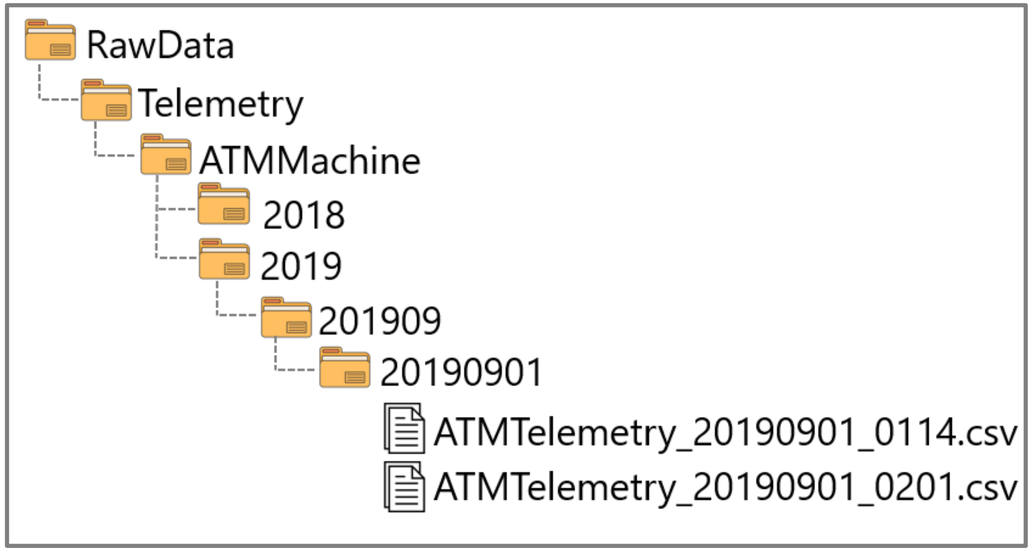
afbs = Azure Blob File System

abfs is the driver
dfs is the endpoint

URI scheme to address a directory:

abfs[s]://filesystemname@accountname.dfs.core.windows.net
/RawData/Telemetry/ATMMachine/2019

Connectivity Option 2: Object Store Endpoint



- RawData
  - Telemetry
    - ATMMachine
      - 2018
      - 2019
        - 201909
          - 20190901
            - ATMTelemetry_20190901_0114.csv
            - ATMTelemetry_20190901_0201.csv

wasb = Windows Azure Storage Blob

wasb is the driver
blob is the endpoint

URI scheme to address a directory:

wasb[s]://containername@accountname.blob.core.windows.net
/RawData/Telemetry/ATMMachine/2019

Multi Protocol Access (MPA) Advantages

1 Object store access provides backwards compatibility with a variety of compute tools and frameworks, such as:

- Azure Stream Analytics
- Azure Event Hubs
- Azure IoT Hub
- Azure Search
- Azure Data Box
- Custom applications
- Third parties & partners

(Other services, such as Azure Data Factory, Azure Databricks, HDInsight, Azure SQL DW are already compatible with the DFS endpoint)

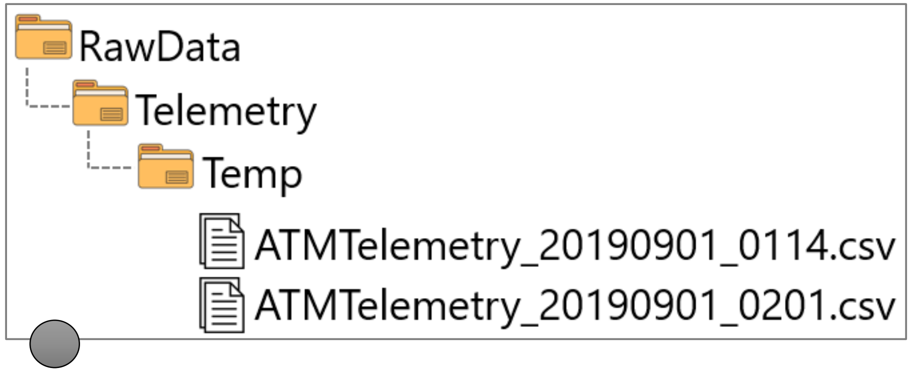
Multi Protocol Access (MPA) Advantages

2 Enables features not previously available, such as:

- Hot / cold / archive access tiers
- Lifecycle management policies
- Diagnostic logs
- SDKs for .NET, Java, Python
- PowerShell, CLI
- Notifications (Azure Event Grid)

Multi Protocol Access (MPA) Advantages

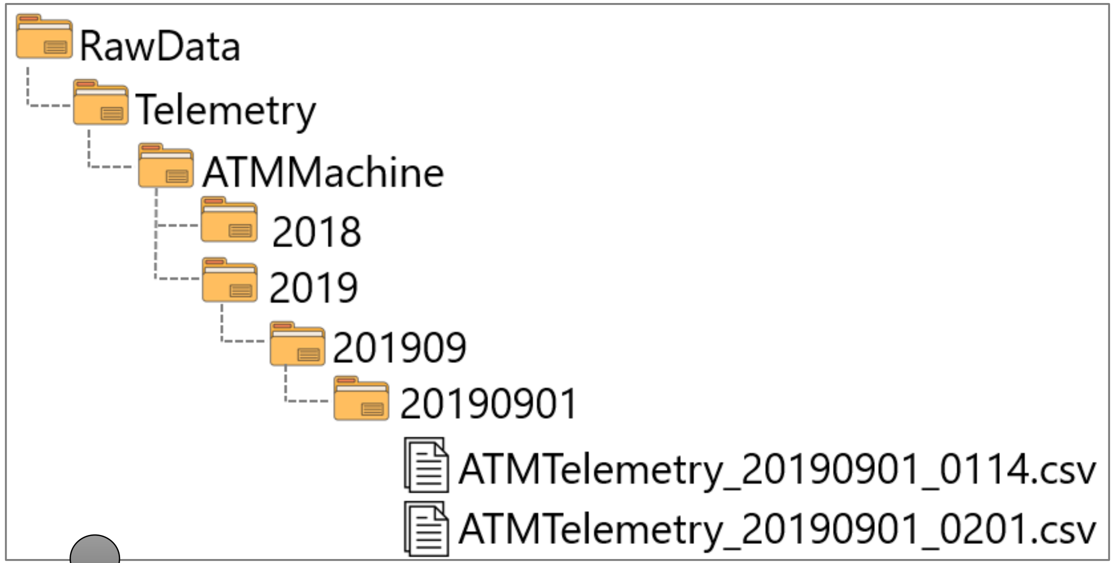
3 Flexibility to use different endpoints for data ingestion vs. data processing



```
RawData
  Telemetry
    Temp
      ATMTelemetry_20190901_0114.csv
      ATMTelemetry_20190901_0201.csv
```

Endpoint:
Object store access (blob)

>



```
RawData
  Telemetry
    ATMMachine
      2018
      2019
        201909
          20190901
            ATMTelemetry_20190901_0114.csv
            ATMTelemetry_20190901_0201.csv
```

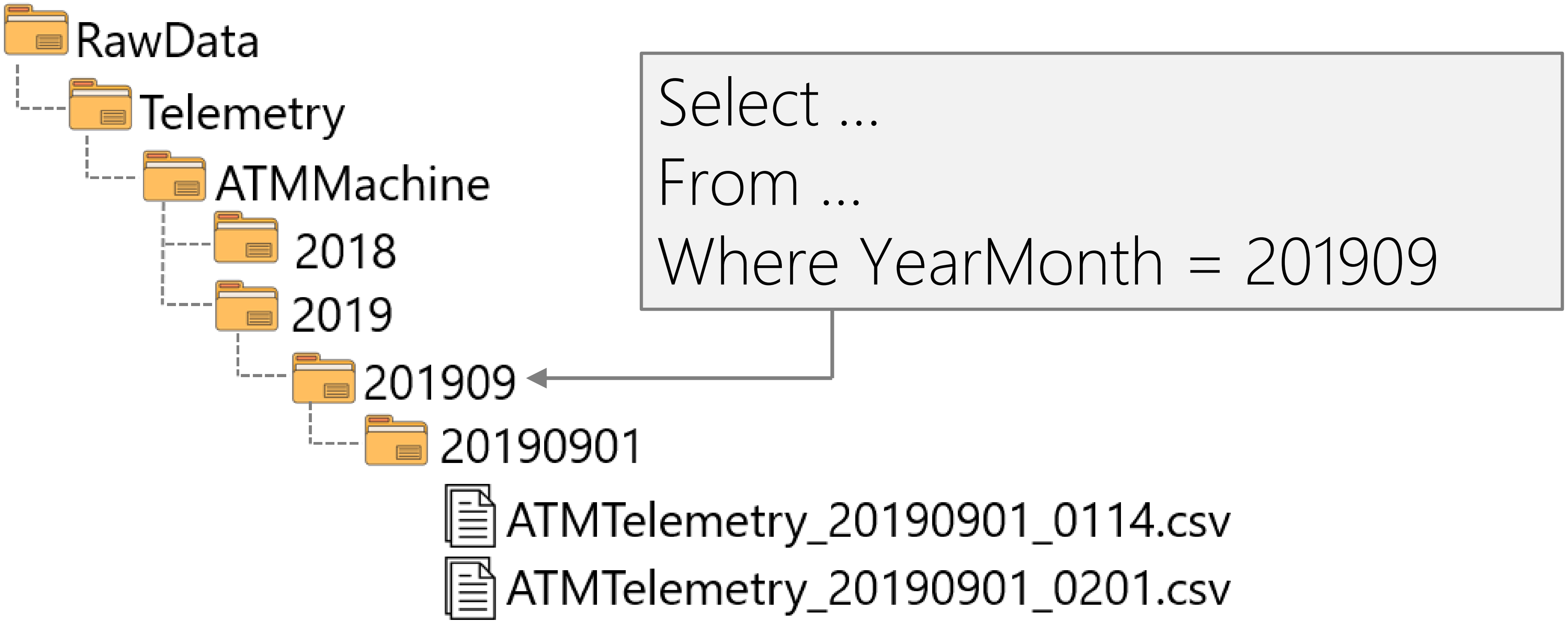
Endpoint:
File system access (dfs)

Initial data ingestion to a Temp directory > Data validations & processing

Which endpoint should we use whenever possible?

Advantages of ABFS Driver + DFS Endpoint

1 Improved query performance with partition scans & partition pruning



RawData

Telemetry

ATMMachine

2018

2019

201909

20190901

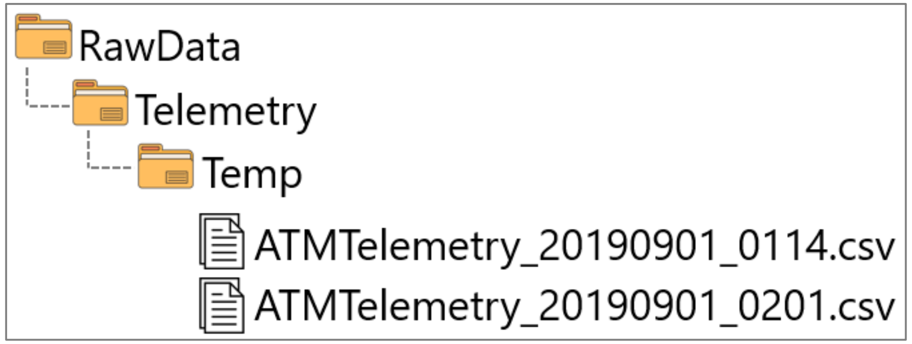
ATMTelemetry_20190901_0114.csv

ATMTelemetry_20190901_0201.csv

```
Select …
From …
Where YearMonth = 201909
```

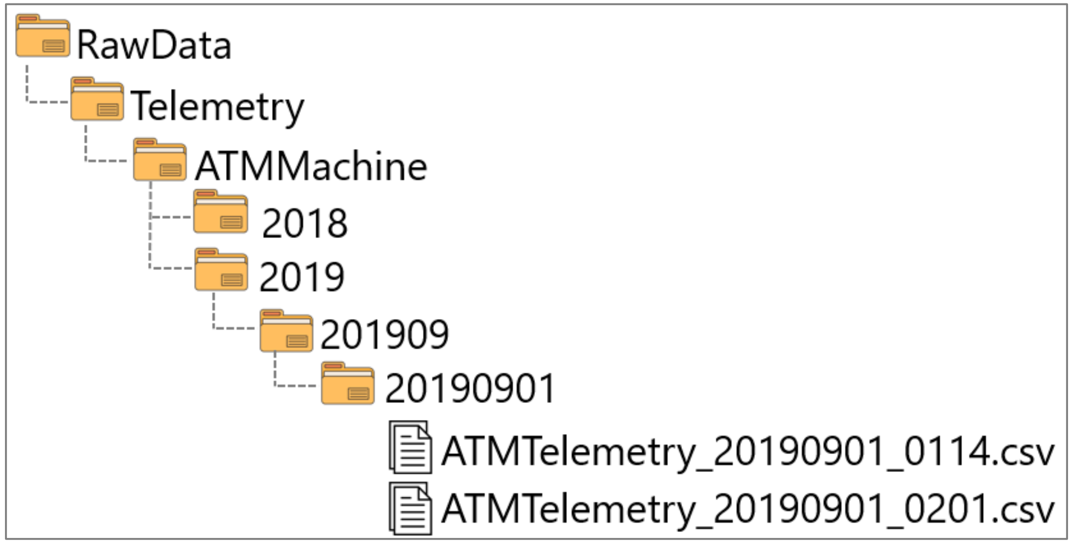
Advantages of ABFS Driver + DFS Endpoint

2 The file system endpoint can perform metadata-only changes which results in significantly better performance (whereas object store which does copies & deletes)



- RawData
  - Telemetry
    - Temp
      - ATMTelemetry_20190901_0114.csv
      - ATMTelemetry_20190901_0201.csv

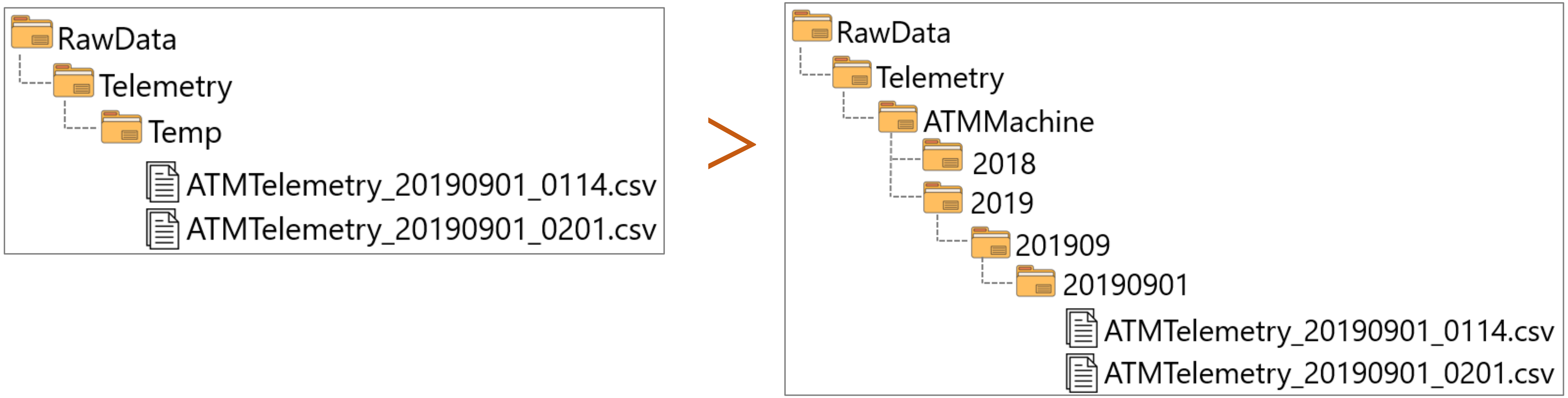
>



- RawData
  - Telemetry
    - ATMMachine
      - 2018
      - 2019
        - 201909
          - 20190901
            - ATMTelemetry_20190901_0114.csv
            - ATMTelemetry_20190901_0201.csv

Advantages of ABFS Driver + DFS Endpoint

3 Improved data consistency with atomic operations (because the entire data operation succeeds or fails as a unit)



- RawData
  - Telemetry
    - Temp
      - ATMTelemetry_20190901_0114.csv
      - ATMTelemetry_20190901_0201.csv

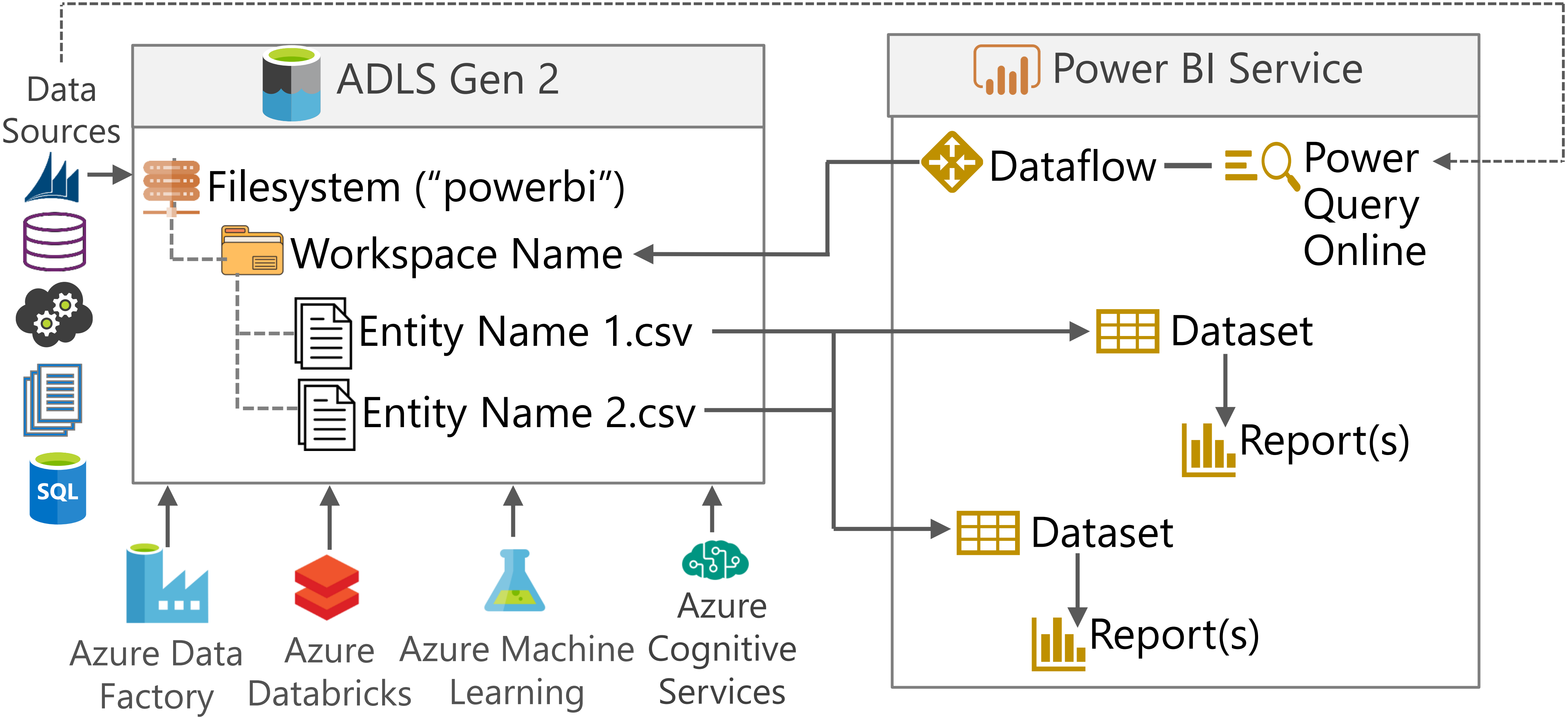
>

- RawData
  - Telemetry
    - ATMMachine
      - 2018
      - 2019
        - 201909
          - 20190901
            - ATMTelemetry_20190901_0114.csv
            - ATMTelemetry_20190901_0201.csv

Azure Data Lake Storage Gen 2: Integration with Azure Services

Integration w/ Power BI Dataflows

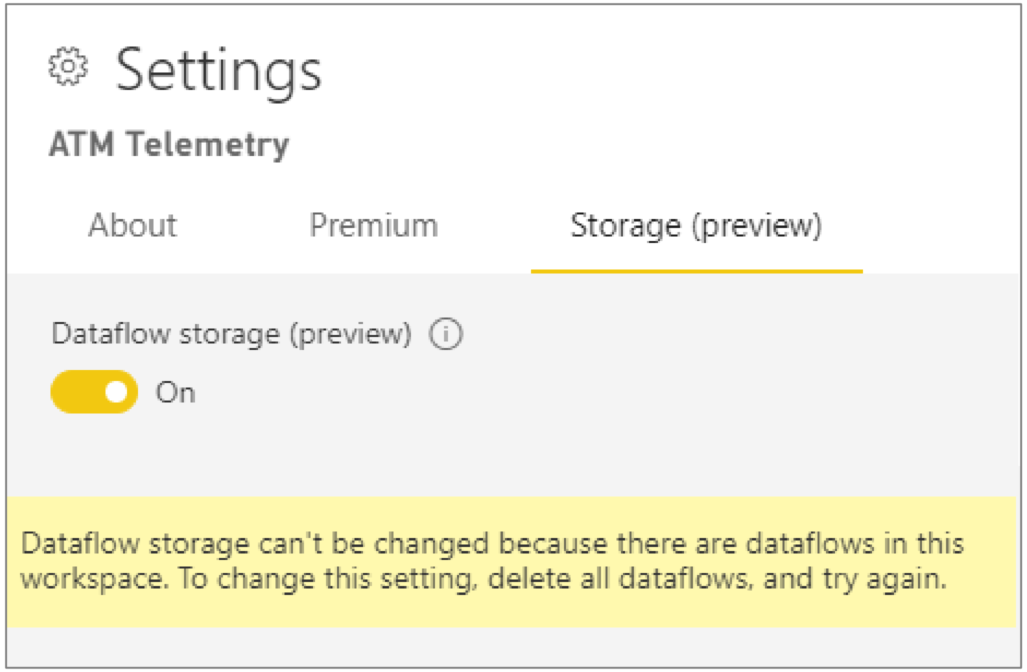
Common Data Model-compliant folders stored in ADLS Gen 2:



Integration w/ Power BI Dataflows

Tenant settings: Asssociate the ADLS Gen 2 account w/ the Power BI tenant.

Workspace settings: Every Power BI workspace which contains dataflows needs to be enabled to store data in ADLS Gen 2:



Settings

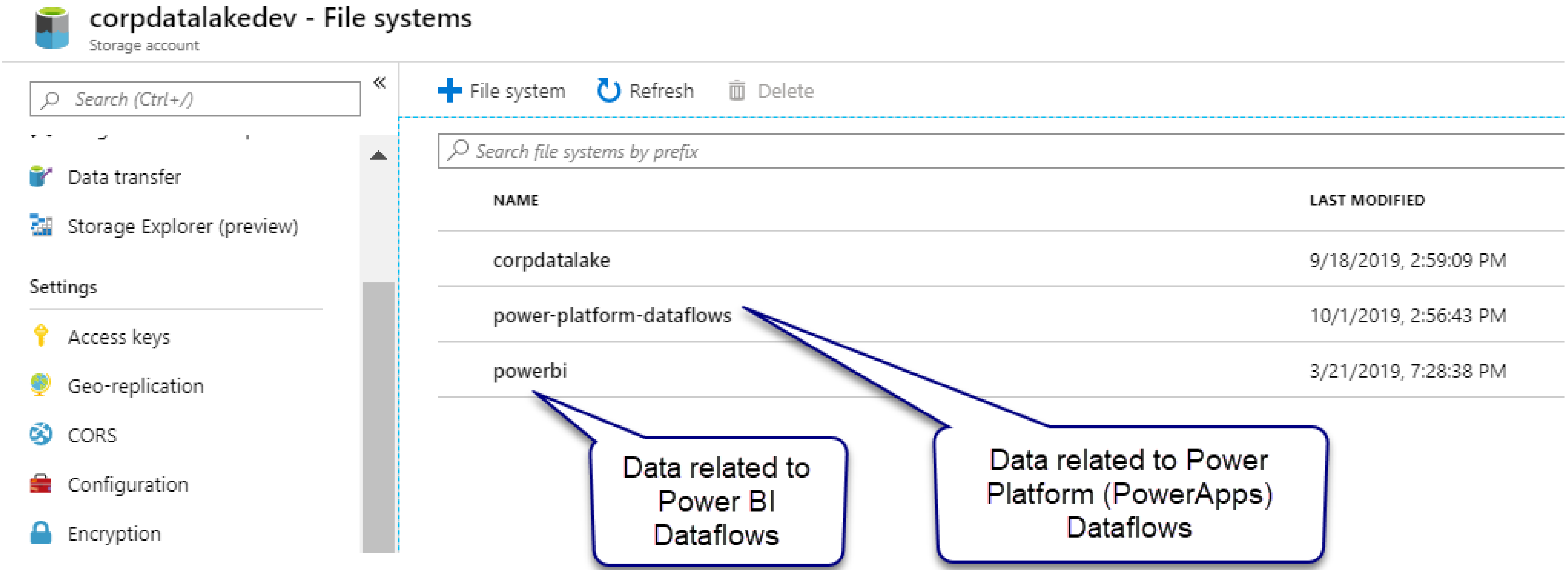
ATM Telemetry

Dataflow storage (preview): On

Dataflow storage can't be changed because there are dataflows in this workspace. To change this setting, delete all dataflows, and try again.

Integration w/ Power BI & Power Platform Dataflows

Data is segregated into its own filesystem in ADLS Gen 2:



corpdatalakedev - File systems

Storage account

NAME	LAST MODIFIED
corpdatalake	9/18/2019, 2:59:09 PM
power-platform-dataflows	10/1/2019, 2:56:43 PM
powerbi	3/21/2019, 7:28:38 PM

Data related to Power BI Dataflows

Data related to Power Platform (PowerApps) Dataflows

Azure Integration Options

Utilities

Azure Storage Explorer
AzCopy
DistCp
PowerShell
CLI

Data & Analytics

Azure Data Factory
Azure Databricks
Azure SQL Data Warehouse
SQL Server 2019 Big Data Clusters--HDFS Tiering

Azure Machine Learning
Azure Cognitive Services
Azure HDInsight

Power Platform

Power BI Dataflows
Power Platform Dataflows

Other

Azure Data Share
Azure Event Grid

Data Ingestion

Azure Stream Analytics
Azure Event Hubs
Azure IoT Hub
Azure Data Explorer
Azure Feature Pack for SSIS

Custom/Dev

REST APIs
.NET SDK
Python SDK
Java SDK
Node.js SDK

Common Azure Data Lake Implementation

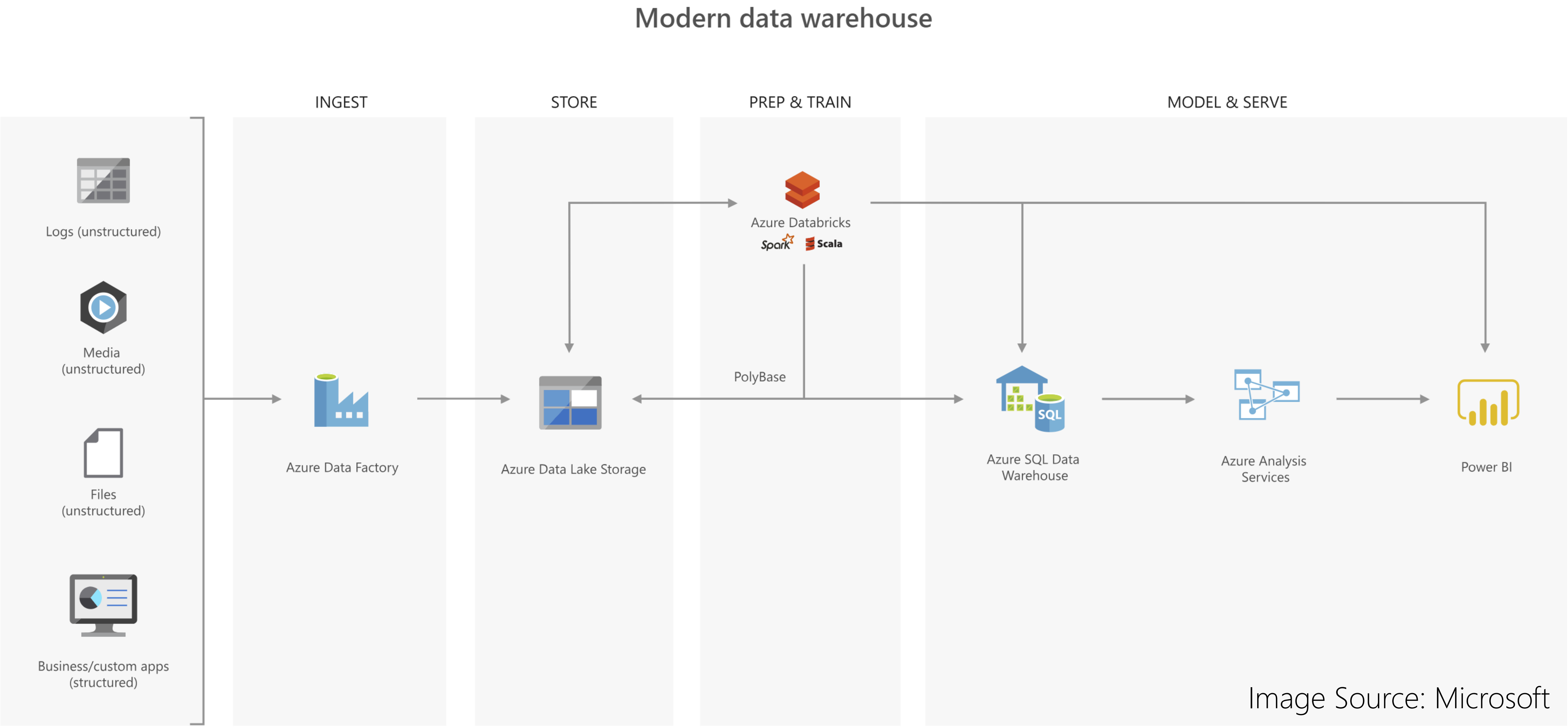


Image Source: Microsoft

Azure Data Lake Storage Gen 2: Current State & Roadmap

Migrating from ADLS Gen 1 to Gen 2

There is not a migration tool, or in-place upgrade, at this time.

Can use tools like Azure Data Factory or AzCopy to help.

Watch out for:

- Any adls:// URIs from Gen 1 need to change to abfss://.
- There was no file size limit in ADLS Gen 1. In Azure Storage, file size is limited to 5 TB.
- Security for Gen 1 was root of account; Gen 2 is root folder.
- File naming rules are a bit different.

Features are Still Evolving

Upgrading, feature support, and roadmap:

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-upgrade

Known issues:

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-known-issues

Multi-Protocol Access (MPA): Public Preview

Multi-protocol access info:

https://azure.microsoft.com/en-us/blog/silo-busting-2-0-multi-protocol-access-for-azure-data-lake-storage/

Register for multi-protocol access preview (subscription whitelisting):

https://forms.office.com/Pages/ResponsePage.aspx?id=v4j5cvGGr0GRqy180BHbR2EUNXd_ZNJCq_eDwZGaF5VURjFLTDRGS0Q4VVZCRFY5MUVaTVJDTkROMi4u

Other Helpful Links

10 Things to Know About Azure Data Lake Storage Gen 2:

https://www.blue-granite.com/blog/10-things-to-know-about-azure-data-lake-storage-gen2

Thank You!

Please visit the Community Resources area at CoatesDataStrategies.com/Presentations to download these slides.