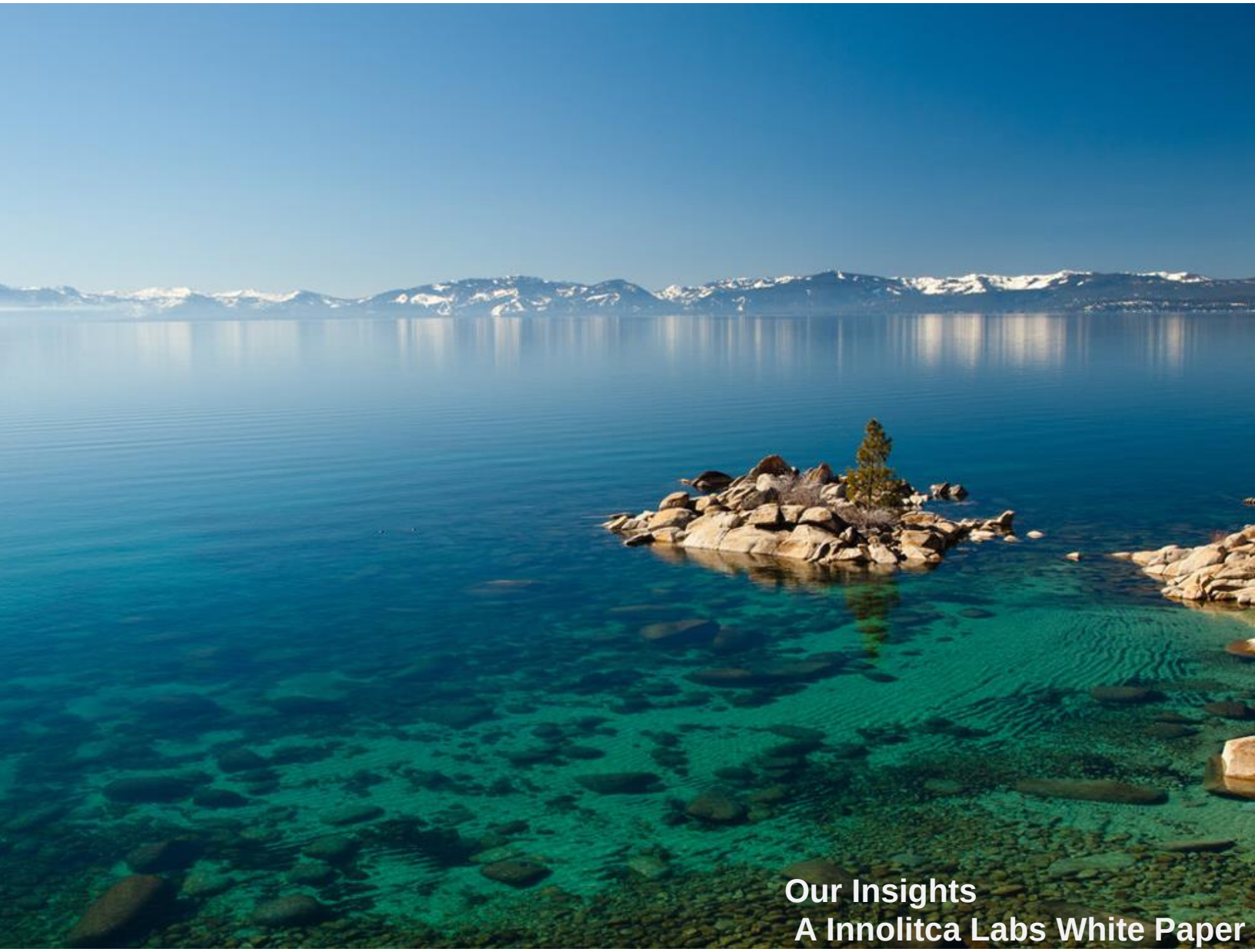
Why Enterprise Should Consider Azure Data Lake?



About the Author

Sanjay Kumar is founder of boutique Big Data and Advanced Analytics consulting firm, Innolitica Labs. He has over twenty years of experience in Big Data, Data Science, and IoT. As a Big Data and Data Science expert, he has consulted many F500 companies and enabled them to harness power of data and analytics.

Sanjay has MS in computer science from IIT Chicago and MBA from Northwestern's Kellogg School of Management. You can reach him at sanjay.kumar@innolitica.com

1. Introduction

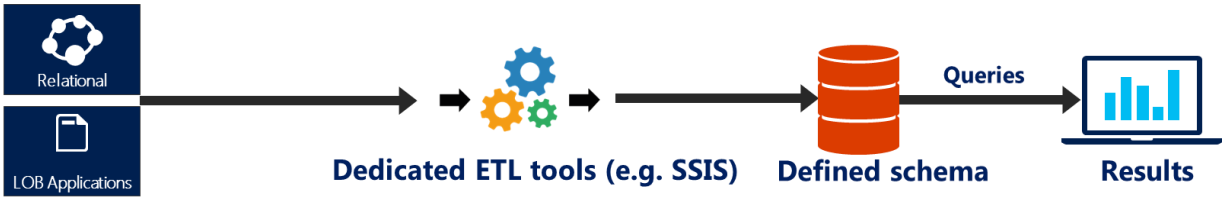
Data is gold and every enterprise wants to monetize it. However, how to harness the power of data is still the biggest challenge. Some of the challenges are building infrastructure to store the data, identifying use cases, acquiring data sources, hiring skilled professional who can mine the data, and domain expertise to drive insights. Microsoft has recently released Azure Data Lake service on Azure Cloud Platform which addressed data storage and management problem by offering Big Data as Service. In this paper, we will discuss how Azure Data Lake can enable enterprise to accelerate Big Data adoption.

2. Why Data Lake

Enterprise needs to store, analyze, and drive insights from data. However this is easier said than done as enterprises are facing many challenges. Some of them are

- Data in silo
- Enabling any application to access any data
- Scalability and performance
- Security
- Acquiring new skills

To solve the data in silo problem, enterprises need to store data at a single repository. There are two approaches to do it - top down approach and bottom up approach. In top down approach, schema is predefined base on some pre-determined use cases, and data is transformed and stored based on pre-defined schema. This approach is called schema on write and is suitable for getting information about **PAST** – what has happened.



. **Figure 1 – Traditional Business Intelligence Data Flow**

On the other hand, in bottom up approach, data need to be stored in a central repository called “Data Lake” and there is no need to define schema at the time of storage. This approach is called schema on read and is suitable to determine **PREDICTIVE** or **PRESCRITIVE** patterns – what will happen in future.

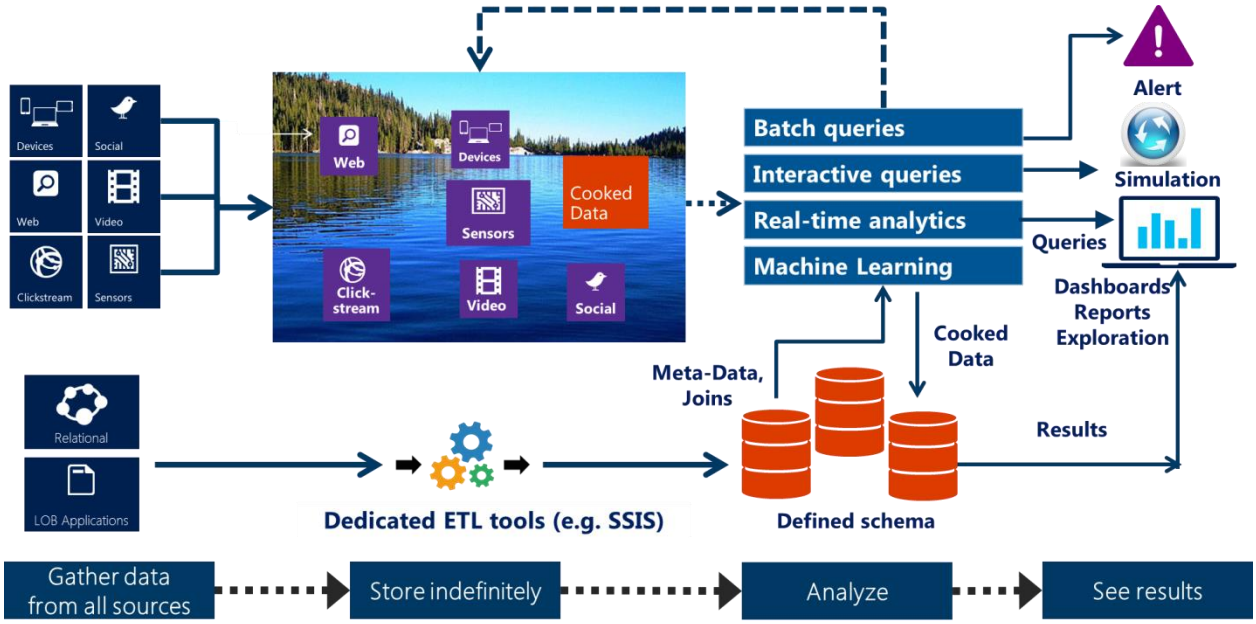


Figure 2 – Modern Data Lake Architecture

3. How to build a Data Lake

There are three ways to build data lake

- Build Hadoop Platform
- Hadoop as a service
- Data Lake as a Service

For Building Hadoop platform, enterprise need to

- Select open source stack (HDP/CDH/MAPR)
- Identify several components based on workload
- Size hardware requirements
- Procure hardware, racks space, networking equipment etc.
- Setup an installation and management services such as Ambari
- Install core and optional Hadoop Components

- Secure Hadoop cluster
- Configure and tune cluster
- Hire skilled professionals to manage the cluster which is very expensive and not readily available
- Manage the cluster and debug problems which is very time consuming

Once Hadoop platform is built, scaling the platform is another major issue. Hadoop platform can be scaled on two dimensions

a. Storage
b. Compute

In most cases, enterprise either needs to scale on store dimension or on compute dimension. There are very few scenarios where scalability is required on both dimensions. Currently if an enterprise needs to scale up Hadoop cluster, it needs to add additional nodes – i.e. pay for storage as well as compute. Inherently, enterprise is paying for something that they are not using most of the times and it cost millions of dollars.

Azure Data Lake is designed to solve these problems. First, Azure data lake is offered as Big Data as a Service (Pass). It means clusters are managed and monitored by Microsoft. Enterprise does not need to hire skilled professionals to upgrade, add patches, and manage it.

Second, Azure Data is designed to solve scalability issue to ensure that enterprise should pay for what they are using. Compute and storage have been separated in Azure Data Lake to ensure that enterprise should pay what they are using for.

Azure Data Lake has two components

- Azure Data Lake Store (ADLS)
- Azure Data Lake Analytics (ADLA)

It enables customers to scale on either of dimension without paying for other. For example, if storage requirement is growing, customers can buy storage only without paying for compute and vice-versa.

4. Key features of Azure Data Lake Store (ADLS)

a. Storage Without Limitations

- ADLS can store unstructured, semi-structured, and structured data. It does not enforce any schema at the time of storing data.
- ADLS can store any size of data, any number of files and there is no limitation on file size. Each ADLS file store is sliced into blocks and blocks are distributed across multiple data nodes. There is no limitations of number of blocks and data nodes. These blocks runs on Azure cloud which has virtually unlimited resources.
- Azure Data Lake Store is seamlessly integrated with Azure Stream Analytics, Azure Event Hub, and Azure Managed Cluster Services such as Storm. It enables Azure Data Lake store to acts as persistence storage for streams.

b. HDFS file system in cloud

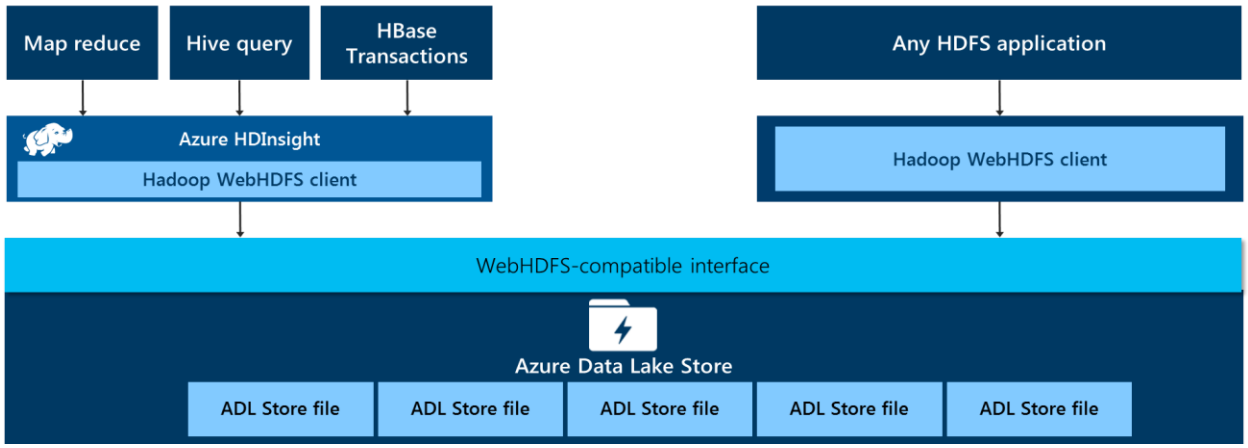


Figure 3 – ADLS and HDFS Interface

Azure Data Lake file system is compatible with open source HDFS file system. It supports WebHDFS API that can be used to integrate current and future Hadoop ecosystem's open source tools with ADLS. It means enterprise can leverage state of the art tools from open source community or Hadoop ecosystem and integrate with Azure Data Lake store. They do not need to depend on Microsoft to integrate state of the art tools and technology to ADLS. Furthermore, data in Azure Data Lake Store can be easily analyzed using open source tools such as spark, hive or pig. Also, Microsoft HDInsights cluster can be configured and provisioned to directly access data from Azure Data Lake Store.

c. Built to support Analytics workload

Azure Data Lake Store is built to support massive throughput and latency to query and analyze large amount of data. To process petabyte size of file in parallel, it splits the files in small chunks and sends to multiple nodes that improves throughput and latency. Internally, thousands of mappers and reducers run in parallel to support massive parallel processing. It supports processing of batch, streaming, and interactive workloads.

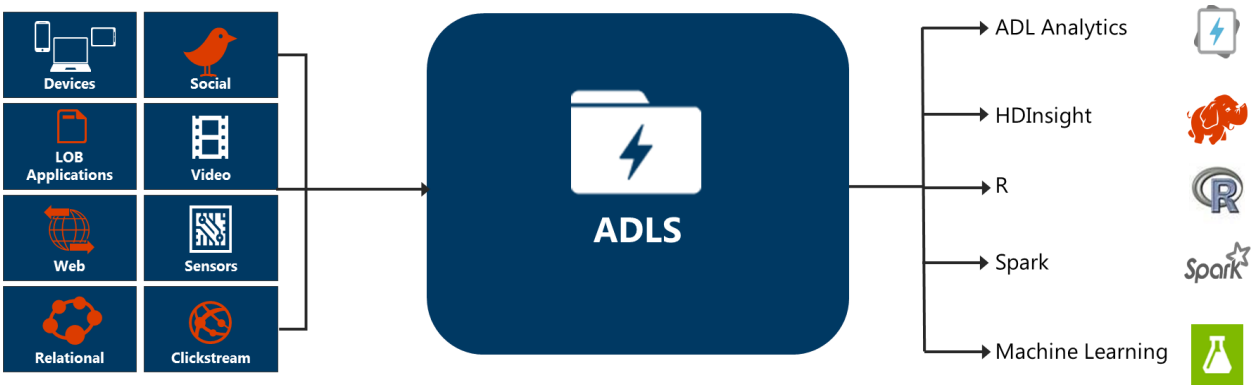


Figure 4 – ADLS Interface with Analytics Tools

d. Enterprise Ready

Security

Azure Data Lake provides enterprise grade security. All data are first encrypted then stored in ADLS. When data is being transferred or in motion, transfer link is also being encrypted. ADLS is tightly integrated with Windows Active Directory and POSIX-style permissions are applied to files and directories. It also provides mechanism to enforce regulatory compliance. Administrators can easily perform audit on all operations.

Availability and Reliability

- Azure maintains three replicas of each data object per region across three fault and upgrade domains. Each create or append operation on a replica is replicated to other two
- Writes are committed to application only after all replicas are successfully updated
- Read operations can go against any replica
- Provides 'read-after-write' consistency

5. Data Ingestion Tools for Azure Data Lake Store

Data can be ingested into Azure Data Lake Store (ADLS) using following tools

Bulk Ingestion

- Azure Data Factory
- AdlCopy
- Distcp
- Sqoop
- Custom Programs (PowerShell/CLI/.Net)

Event Ingestion

- Azure Event Hub
- HDI Storm

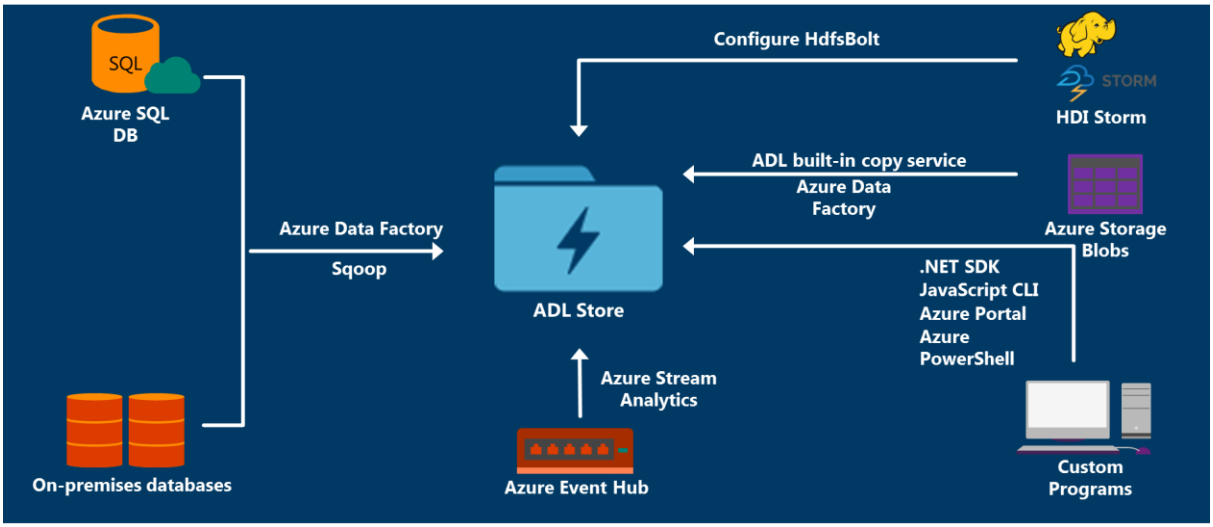


Figure 5 – Data Ingestion Tools in ADLS

a. Azure Data Factory

Azure data lake factory can be used to move data from anywhere to data lake. Connectors are built within Azure Data Factory that enables data to move data from relational databases, apps, and unstructured data sources to azure data lake.

b. Azure Streaming Analytics

Azure streaming analytics service is tightly integrated with Azure Data Lake Store. Streaming data can be ingested into ADLS using Azure Stream Analytics service.

c. AdlCopy

AdlCopy is a command line utility tool that can be used to copy data from Azure Blob Storage to Data Lake Store or between two Data Lake Store accounts.

d. DistCP

DistCP command can be used to copy data from HDInsight clusters.

e. Sqoop

Sqoop can be used to copy data from relational databases and Data Lake Store.

6. How to Ingest, Store, and Process different type of data in ADLS

a. Events/Streams

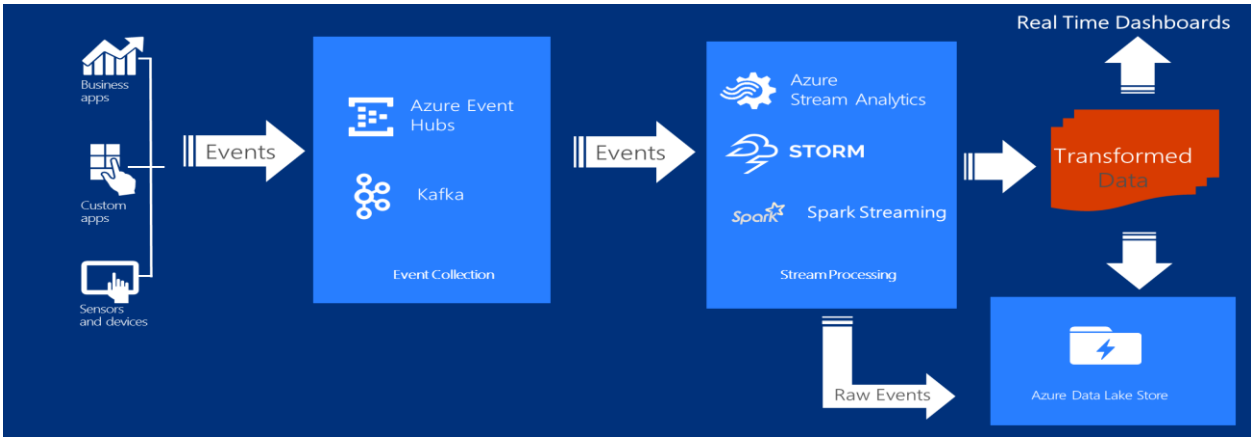


Figure 6 – Event Ingestion in ADLS

The above figure provides solution architecture of ingesting, processing, and storing real-time events. An event can be generated by multiple data sources. These events are can be collected by event collectors Kafka or Event Hub. These events can be processed by Azure stream analytics or Storm or Spark Streaming. After processing, events can be stored into Azure Data Lake Store and at the same time can be transformed and sent to real-time dashboard.

b. Bulk Load

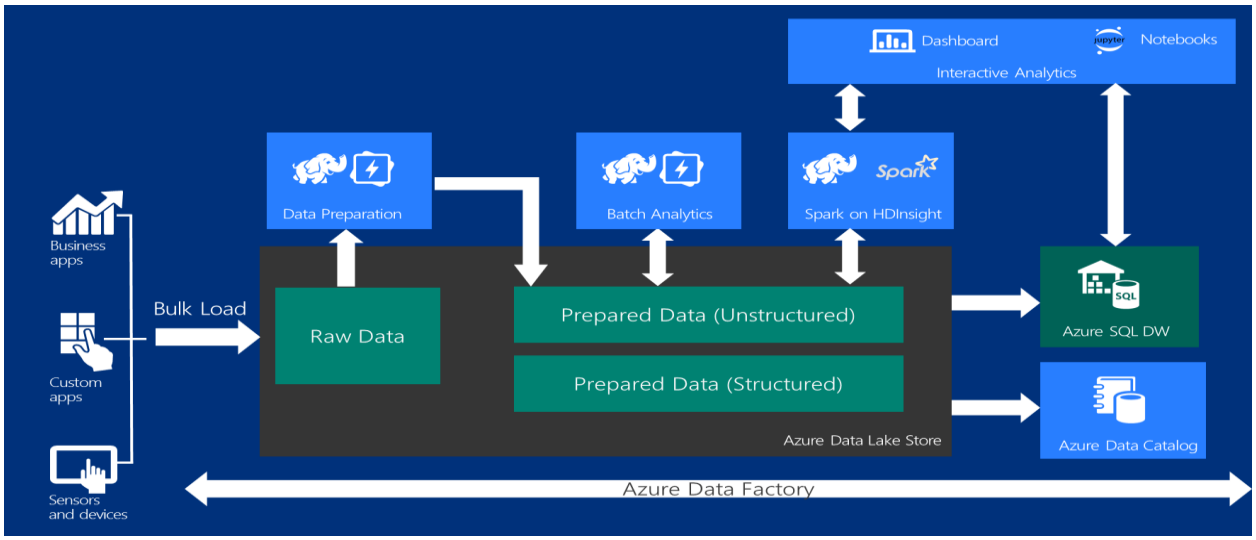


Figure 7 – Bulk Data Ingestion in ADLS

For data at rest or bulk load, first data is uploaded into Azure Data Lake Store as Raw Data. After preparation, data is ready for analytics. Either Azure Data Lake Analytics or HDInsights can be used to preform analytics. Processed data can be also stored into dataware house for reporting purpose. Data can be also registered with Azure Data Catalog for management purpose.

7. Conclusion

Azure Data Lake service enables enterprise to accelerates adoption of data lake and become a data driven organization. It significantly reduces cost to build and manage Data Lake without worrying about scalability, reliability, and security. With robust interface with multiple machine learning tools such as R, Python, and Azure ML, a data scientist can easily leverage data stored in Azure Data Lake Store to build intelligent solutions.