Understand Data

What is Data?

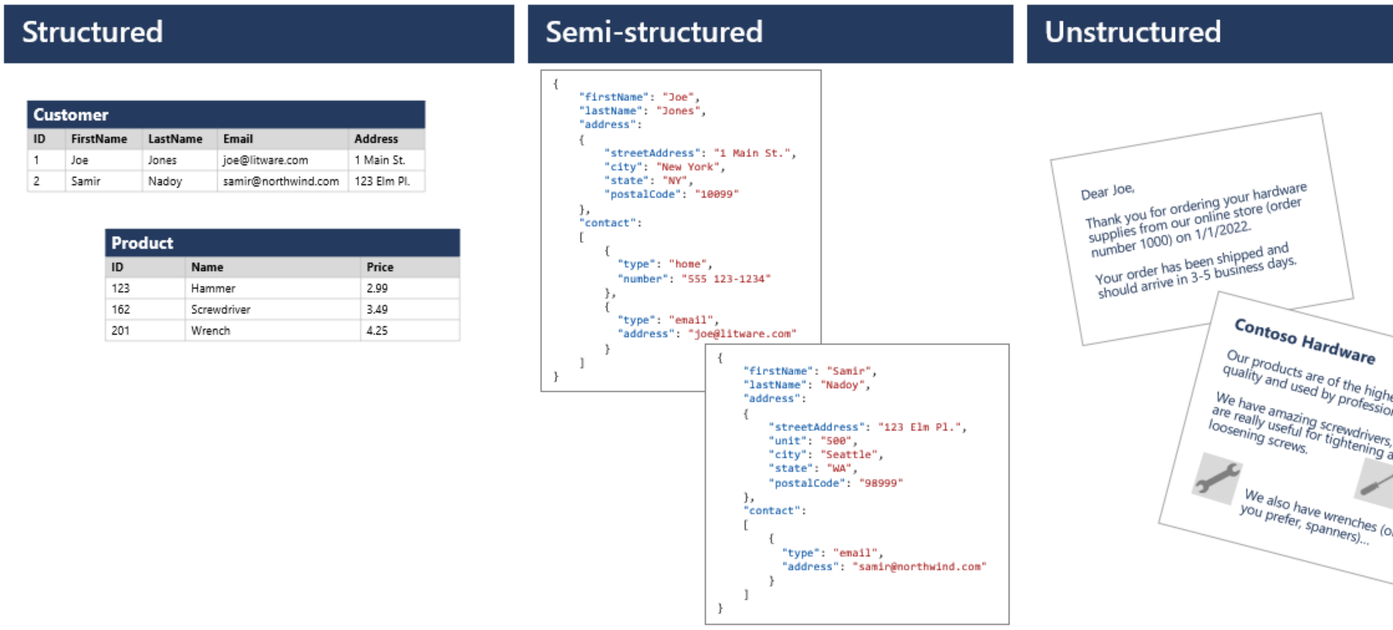
- Data is unit value associated with some Subject.
- It can be number,description and observation used to record the information.
- Often representing ***entities*** that have one or more ***attributes***

Example:

Customer(Name,Email,Address)

Types of Data

- Structured
- Semi-structured
- Unstructured



Structured

Customer

ID	FirstName	LastName	Email	Address
1	Joe	Jones	joe@litware.com	1 Main St.
2	Samir	Nadoy	samir@northwind.com	123 Elm Pl.

Product

ID	Name	Price
123	Hammer	2.99
162	Screwdriver	3.49
201	Wrench	4.25

Semi-structured

```
{
    "firstName": "Joe",
    "lastName": "Jones",
    "address":
    {
        "streetAddress": "1 Main St.",
        "city": "New York",
        "state": "NY",
        "postalCode": "10099"
    },
    "contact":
    [
        {
          "type": "home",
          "number": "555 123-1234"
        },
        {
          "type": "email",
          "address": "joe@litware.com"
        }
    ]
}
```

```
{
    "firstName": "Samir",
    "lastName": "Nadoy",
    "address":
    {
        "streetAddress": "123 Elm Pl.",
        "unit": "500",
        "city": "Seattle",
        "state": "WA",
        "postalCode": "98999"
    },
    "contact":
    [
        {
          "type": "email",
          "address": "samir@northwind.com"
        }
    ]
}
```

Unstructured

Dear Joe,

Thank you for ordering your hardware supplies from our online store (order number 1000) on 1/1/2022.

Your order has been shipped and should arrive in 3-5 business days.

Contoso Hardware

Our products are of the highe quality and used by profession

We have amazing screwdrivers, are really useful for tightening a loosening screws.

We also have wrenches (o you prefer, spanners)...

Structured Data:

- Fixed Schema
- Table Based flat structure
- Do not support Hierarchical storage
- Normalization process is used for Relational database design.
- Here we get large number of narrow, well-defined **tables** (a *narrow* table is a table with few columns), with references from one table to another.
- Normalization enables fast throughput of transactions.

- Querying the data often requires reassembling information from multiple tables by joining the data back together at run-time. These types of queries can be expensive.
- Vertically Scalable

Semi-Structured Data:

- **Semi-structured** data is information that does not reside in a relational database but still has some structure to it.
- Data structure is defined within the actual data by fields.
- Dynamic , flexible Schema.
- Key-value pairs, document-based, graph databases or wide-column stores
- Querying does not require join.

Example: Retrieving the details of a customer, including the address, is a matter of reading a single document

- Horizontally Scalable

What is NoSQL?

- This is the term used to describe non-relational Data Management System, that does not require a fixed schema.
- Non-relational collection can have Multiple entries in the same collection or container with different fields.

Example:

- Documents held in *JavaScript Object Notation* (JSON) format
- Key-Value store
- Graph Database
- Column family databases

Document:

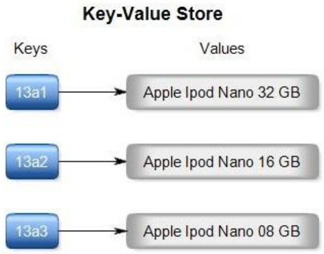
Document databases typically store data in JSON format.

## Document 1 ##	## Document 2 ##
{ "customerID": "103248", "name": { "first": "AAA", "last": "BBB" }, "address":	{ "customerID": "103249", "name": { "title": "Mr", "forename": "AAA", "lastname": "BBB" },

{
"street": "Main Street",
"number": "101",
"city": "Acity",
"state": "NY"
},
"ccOnFile": "yes",
"firstOrder": "02/28/2003"
}
	"address":
	{
	"street": "Another Street",
	"number": "202",
	"city": "Bcity",
	"county": "Gloucestershire",
	"country-region": "UK"
	},
	"ccOnFile": "yes"
	}

Key Value:

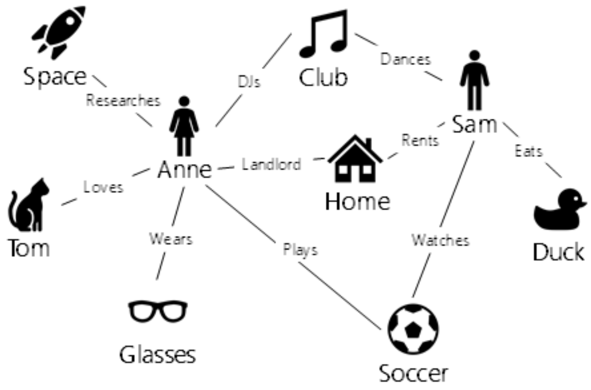
- A key-value store is like a relational table, except that each row can have any number of columns.
- A key value store uses a **hash table** in which there exists a **unique key** and a **pointer** to a particular item of data.

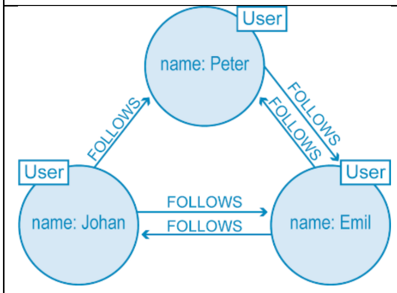


Graph:

- Graph databases enable you to store entities, but the main focus is on the relationships that these entities have with each other.
- **It can be used to** store and query information about complex relationships. A graph contains nodes (information about objects), and edges (information about the relationships between objects).

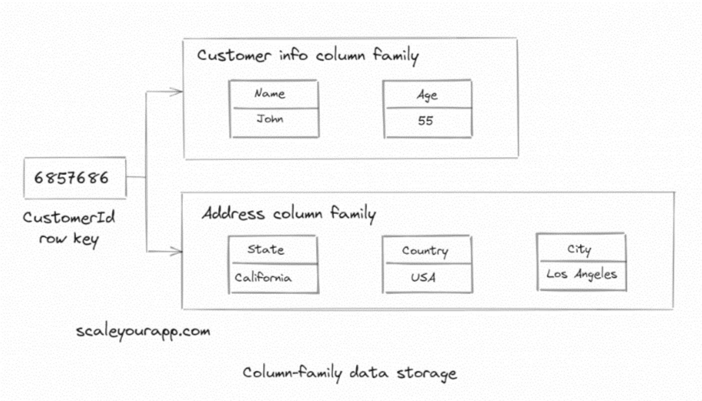
Example:





Column family Databases:

- A column family database organizes data into rows and columns.
- Colum family database data, is organized into column families, where each column family represents a group of related data.
- It is a tuple (pair) that consists of a key–value pair, where the key is mapped to a value that is a set of columns.
- Columnar database is optimized for fast retrieval of columns of data, typically in analytical applications



Unstructured Data:

- Audio and video files, and binary data files might not have a specific structure. They are referred to as *unstructured* data.
- The sources of unstructured data can be sensors, documents, digital stream, Logs, Social media etc.

Unstructured data is Frequently used in combination with Machine Learning or Cognitive Services capabilities to "extract data" by using:

- Text Analytics
- Sentiment Analysis with Cognitive APIs
- Vision API

Data Stores:

There are two broad categories of data store in common use:

- File stores
- Databases

File storage

- Files can be stored in local file systems.
- most organizations, important data files are stored centrally in some kind of shared file storage system
- Increasingly, that central storage location is hosted in the cloud, enabling cost-effective, secure, and reliable storage for large volumes of data.

File Formats:

The specific file format used to store data depends on a number of factors, including:

- The type of data being stored (structured, semi-structured, or unstructured).
- The applications and services that will need to read, write, and process the data.
- The need for the data files to be readable by humans, or optimized for efficient storage and processing.

Delimited text files:

- Data is often stored in plain text format with specific field delimiters and row terminators.
- **Delimited text is a good choice for structured data that needs to be accessed by a wide range of applications and services in a human-readable format**

Example:CSV

```
FirstName,LastName,Email
Joe,Jones,joe@litware.com
```

```
Samir,Nadoy,samir@northwind.com
```

JSON:

- JavaScript Object Notation is a schema-less, text-based representation of structured data that is based on key-value pairs and ordered lists.
- It is used for storing and exchanging the data between computers, more commonly used in network communication
- JSON is language independent used by any programming language

JSON allows you to create a hierarchical structure of your data. used to define data entities (objects) that have multiple attributes. Each attribute might be an object (or a collection of objects); making JSON a flexible format that's good for both structured and semi-structured data.

Hierarchical document schema is

```
{
 "customers":
 [
  {
   "firstName": "Joe",
   "lastName": "Jones",
   "contact":
   [
    {
     "type": "home",
     "number": "555 123-1234"
    },
    {
     "type": "email",
     "address": "joe@litware.com"
    }
   ]
  },
  {
   "firstName": "Samir",
   "lastName": "Nadoy",
   "contact":
   [
    {
     "type": "email",
     "address": "samir@northwind.com"
    }
   ]
  }
 ]
}
```

XML:

- XML is a markup language, To define a syntax for encoding documents which are both humans and machines readable.
- It is standards file format for exchange between applications.

```
<students>
```

```
 <student>
   <name>Rick Grimes</name>
   <age>35</age>
   <subject>Maths</subject>
   <gender>Male</gender>
 </student>
 <student>
   <name>Daryl Dixon </name>
   <age>33</age>
   <subject>Science</subject>
   <gender>Male</gender>
 </student>
 <student>
   <name>Maggie</name>
   <age>36</age>
   <subject>Arts</subject>
   <gender>Female</gender>
 </student>
</students>
```

- The redundant nature of the syntax causes higher storage and transportation cost when the volume of data is large.
- Not splitable since XML has an opening tag at the beginning and a closing tag at the end. You cannot start processing at any point in the middle of those tags.
- It's largely been superseded by the less verbose JSON format

BLOB:

- All files are stored as binary data (1's and 0's), but in the human-readable formats discussed above, the bytes of binary data are mapped to printable characters through encoding(ASCII ,Unicode)
- Some file formats however, particularly for unstructured data, store the data as raw binary that must be interpreted by applications and rendered.
- Common types of data stored as binary include images, video, audio, and application-specific documents and commonly referred as BLOB

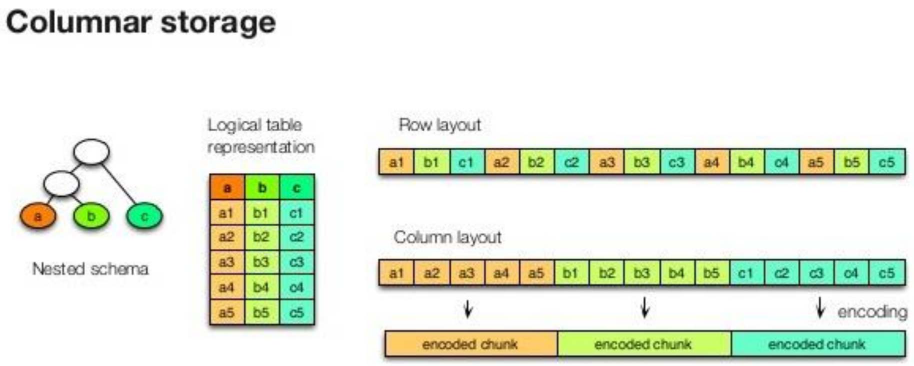
Optimized file formats:

- **Avro** is an row based, open-***source* schema** specification for data serialization that provides serialization and data exchange services for Apache Hadoop.
- **Each record contains header** that describes the structure of the data in the record in JSON

- The Data is stored in a binary format making it compact and efficient.
- An application uses the information in the header to parse the binary data and extract the fields it contains
- Avro is a good format for compressing data and minimizing storage and network bandwidth requirements.

Parquet

- Columanar storage, used by Hadoop systems, such as Pig, Spark, and Hive.
- Supports efficient data compression and encoding schemes that can lower data storage costs.
- It is cross platform and language independent format.
- It stores Metadata that describes the set of rows found in each chunk
- An application can use this metadata to quickly locate the correct chunk for a given set of rows, and retrieve the data in the specified columns for these rows
- Services such as Azure Synapse Analytics, Azure Databricks and Azure Data Factory have native functionality that take advantage of Parquet file formats.



Performance:

- Unlike row-based file formats like CSV, Parquet is optimized for performance.
- You can focus only on the relevant data very quickly. Moreover, the amount of data scanned will be way smaller and will result in less I/O usage

Note: *Consider Parquet when the I/O patterns are more read heavy or when the query patterns are focused on a subset of columns in the records.*

ORC

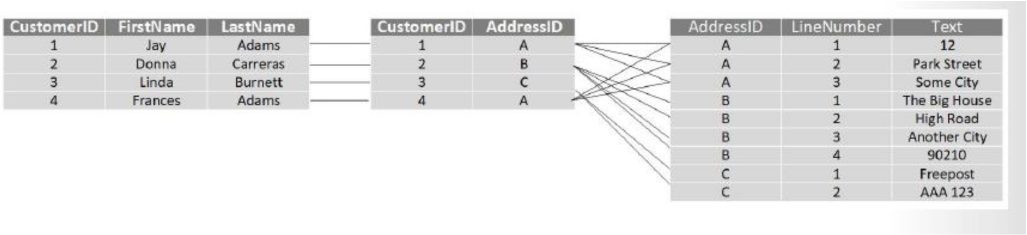
- *ORC* (Optimized Row Columnar format) organizes data into columns rather than rows. It was developed by HortonWorks for optimizing read and write operations in Apache Hive(Hive is a data warehouse system that supports fast data summarization and querying over large datasets)

- An ORC file contains *stripes* of data. Each stripe holds the data for a column or set of columns. A stripe contains an index into the rows in the stripe, the data for each row, and a footer that holds statistical information (count, sum, max, min, and so on) for each column.

Databases

Relational databases:

- **Relational databases** are commonly used to store and query structured data.
- The data is stored in tables
- The tables are managed and queried using Structured Query Language (SQL), which is based on an ANSII standard, so it's similar across multiple database systems.



CustomerID	FirstName	LastName
1	Jay	Adams
2	Donna	Carreras
3	Linda	Burnett
4	Frances	Adams

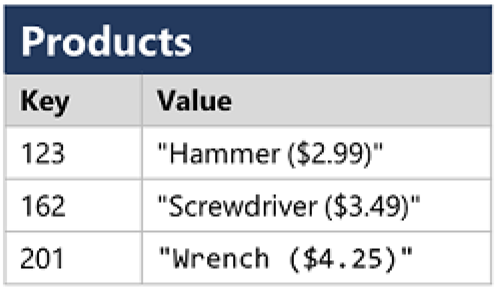
CustomerID	AddressID
1	A
2	B
3	C
4	A

AddressID	LineNumber	Text
A	1	12
A	2	Park Street
A	3	Some City
B	1	The Big House
B	2	High Road
B	3	Another City
B	4	90210
C	1	Freepost
C	2	AAA 123

Non-relational databases

- Non-relational databases are data management systems that don’t apply a relational schema to the data.
- Non-relational databases are often referred to as NoSQL database with some support a variant of the SQL language.
- There are four common types of Non-relational database commonly in use.

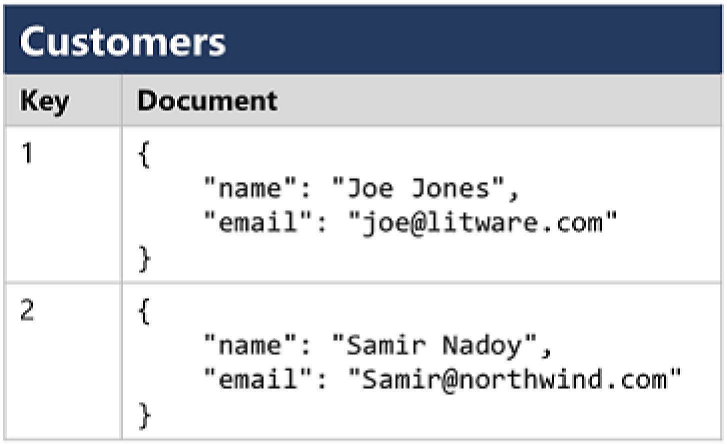
1. **Key-value databases** in which each record consists of a unique key and an associated value, which can be in any format.



Products

Key	Value
123	"Hammer ($2.99)"
162	"Screwdriver ($3.49)"
201	"Wrench ($4.25)"

2. **Document databases**, which are a specific form of key-value database in which the value is a JSON document (which the system is optimized to parse and query)



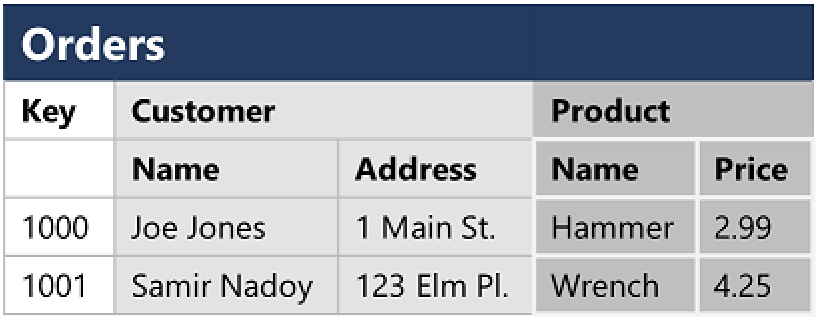
Customers

Key	Document

1	`{`
	`"name": "Joe Jones",`
	`"email": "joe@litware.com"`
	`}`

2	`{`
	`"name": "Samir Nadoy",`
	`"email": "Samir@northwind.com"`
	`}`

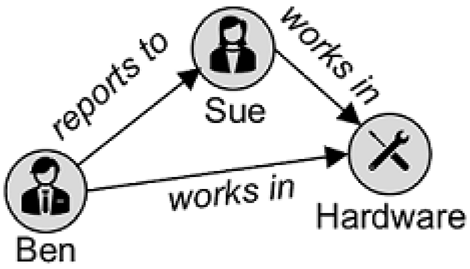
3. **Column family databases**, which store tabular data comprising rows and columns, but you can divide the columns into groups known as column-families. Each column family holds a set of columns that are logically related together.



Orders

Key	Customer		Product	
	Name	Address	Name	Price
1000	Joe Jones	1 Main St.	Hammer	2.99
1001	Samir Nadoy	123 Elm Pl.	Wrench	4.25

4. **Graph databases**, which store entities as nodes with links to define relationships between them.



Relational	Non-Relational
Pre-defined Schema Table Based flat structure Do not support Hierarchical storage	Dynamic , flexible Schema. key-value pairs, document-based, graph databases or wide-column stores
Normalization process is used for Relational database design. Normalization enables fast throughput of transactions. Here we get large number of narrow, well-defined tables (a *narrow* table is a table with few columns), with references from one table to another.	Non-relational databases enable you to store data in a format that more closely matches the original structure. Multiple entities in the same collection or container with different fields

Querying the data often requires reassembling information from multiple tables by joining the data back together at run-time. These types of queries can be expensive.	Querying does not require join. Example: Retrieving the details of a customer, including the address, is a matter of reading a single document
Vertically Scalable	Horizontally Scalable

Storing Data in Azure Cloud:

Depending on the type of data such as structured, semi-structured, or unstructured, data will be stored differently.

- Structured Data Store: Azure SQL Database, Azure SQL Datawarehouse (SQLPool)
- Unstructured Data Store: Azure Blob Storage
- Semi-Structured Data Store: Table Storage for Key-Value, CosmosDB for all different types of semi-structured data.

Note:You need to provision respective services and provide access to users based on their roles.

OLTP/OLAP

- Data processing solutions often fall into one of two broad categories: analytical systems, and transaction processing systems.
  - OLTP: Online is a transactional processing.
  - OLAP: Online analytical processing system

Transactional data workloads

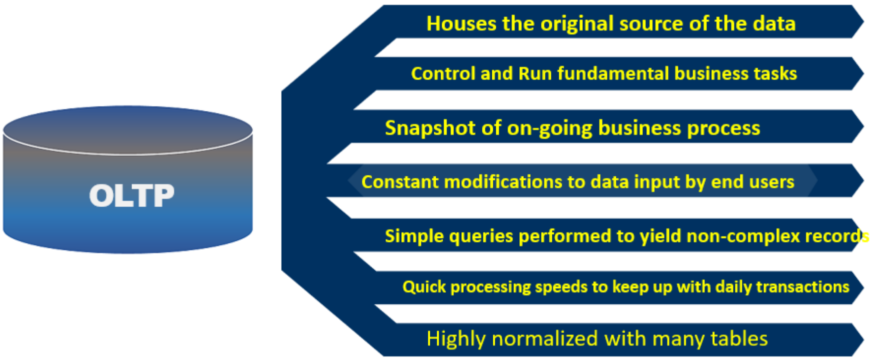
Data is stored in a database that is optimized for *online transactional processing* (OLTP) operations that support applications A mix of *read* and *write* activity

For example:

- Read the *Product* table to display a catalog
- Write to the *Order* table to record a purchase

OLTP:

- OLTP's are operational systems which help execute and record the day-to-day operations of a business



OLTP

Houses the original source of the data

Control and Run fundamental business tasks

Snapshot of on-going business process

Constant modifications to data input by end users

Simple queries performed to yield non-complex records

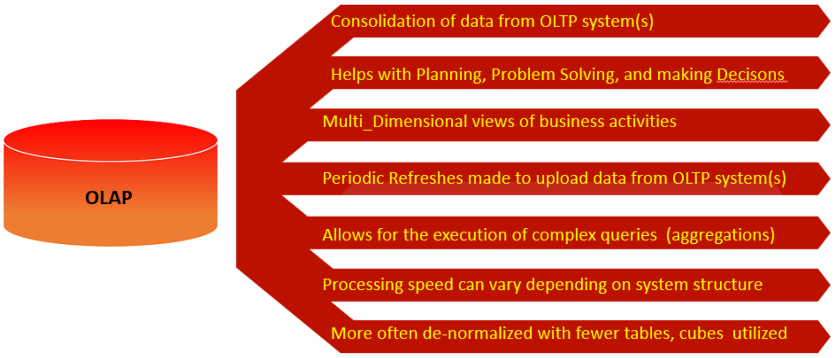
Quick processing speeds to keep up with daily transactions

Highly normalized with many tables

- In these systems, normalization enables fast throughput for transactions but it can make querying more complex as t involves joins on multiple tables.

OLAP:

- Analytical system is designed to support business users who need to query data and gain a *big picture* view of the information held in a database.



OLAP

Consolidation of data from OLTP system(s)

Helps with Planning, Problem Solving, and making Decisons

Multi_Dimensional views of business activities

Periodic Refreshes made to upload data from OLTP system(s)

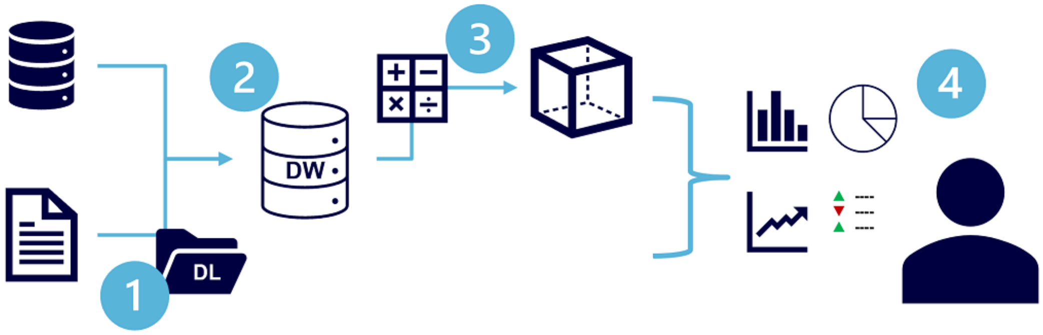
Allows for the execution of complex queries (aggregations)

Processing speed can vary depending on system structure

More often de-normalized with fewer tables, cubes utilized

- Most analytical data processing systems need to perform similar tasks: data ingestion, data transformation, data querying, and data visualization.

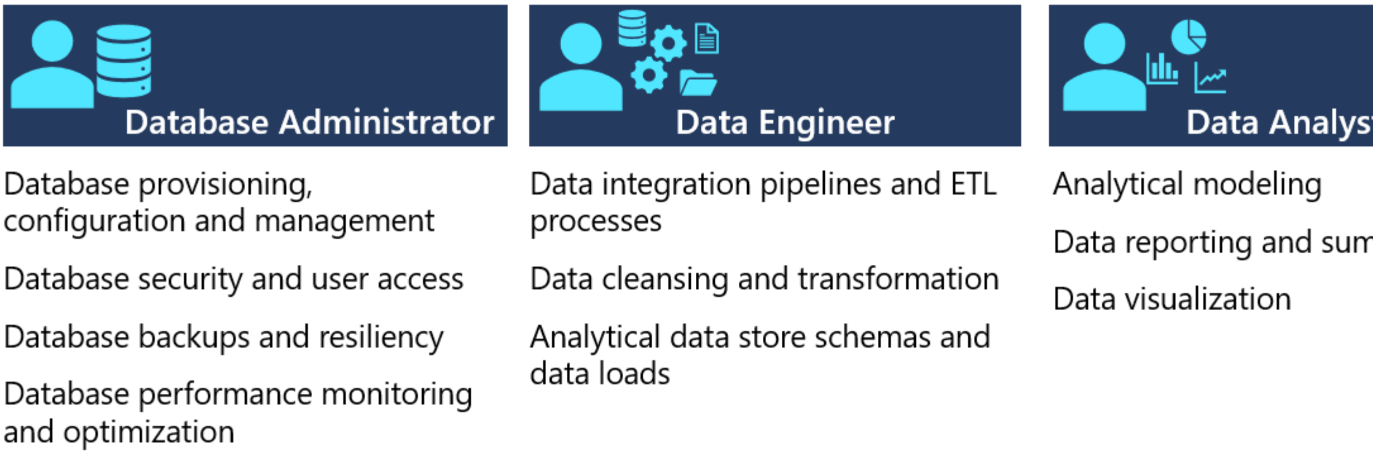
Analytical Data Workloads:



1. Data files may be stored in a central *data lake* for analysis

2. An extract, transform, and load (ETL) process copies data from files and OLTP databases into a *data warehouse* that is optimized for *read* activity
3. Data in the data warehouse may be aggregated and loaded into an online analytical processing (OLAP) model, or *cube*
4. The data in the data lake, data warehouse, and analytical model can be queried to produce reports and dashboards

Data Roles



Database Administrator

Database provisioning, configuration and management

Database security and user access

Database backups and resiliency

Database performance monitoring and optimization

Data Engineer

Data integration pipelines and ETL processes

Data cleansing and transformation

Analytical data store schemas and data loads

Data Analys

Analytical modeling

Data reporting and sum

Data visualization

Data Services in Azure

Azure SQL

Azure SQL is the collective name for a family of relational database solutions based on the Microsoft SQL Server database engine. Specific Azure SQL services include:

- **Azure SQL Database** – a fully managed platform-as-a-service (PaaS) database hosted in Azure
- **Azure SQL Managed Instance** – a hosted instance of SQL Server with automated maintenance, which allows more flexible configuration than Azure SQL DB but with more administrative responsibility for the owner.
- **Azure SQL VM** – a virtual machine with an installation of SQL Server, allowing maximum configurability with full management responsibility.

Azure Database for open-source relational databases

Azure includes managed services for popular open-source relational database systems, including:

- **Azure Database for MySQL** - a simple-to-use open-source database management system that is commonly used in *Linux*, *Apache*, *MySQL*, and *PHP* (LAMP) stack apps.
- **Azure Database for MariaDB** - a newer database management system, created by the original developers of MySQL. The database engine has since been rewritten and optimized to improve performance. MariaDB offers compatibility with Oracle Database (another popular commercial database management system).
- **Azure Database for PostgreSQL** - a hybrid relational-object database. You can store data in relational tables, but a PostgreSQL database also enables you to store custom data types, with their own non-relational properties.

Azure Cosmos DB

- Azure Cosmos DB is a global-scale non-relational (*NoSQL*) database system that supports multiple application programming interfaces (APIs), enabling you to store and manage data as JSON documents, key-value pairs, column-families, and graphs.

Azure Storage

Azure Storage is a core Azure service that enables you to store data in:

- **Blob containers** - scalable, cost-effective storage for binary files.
- **File shares** - network file shares such as you typically find in corporate networks.
- **Tables** - key-value storage for applications that need to read and write data values quickly.

Data engineers use Azure Storage to host *data lakes* - blob storage with a hierarchical namespace that enables files to be organized in folders in a distributed file system.

Azure Data Factory

- Azure Data Factory is an Azure service that enables you to define and schedule data pipelines to transfer and transform data. You can integrate your pipelines with other Azure services, enabling you to ingest data from cloud data stores, process the data using cloud-based compute, and persist the results in another data store.
- Azure Data Factory is used by data engineers to build *extract*, *transform*, and *load* (ETL) solutions that populate analytical data stores with data from transactional systems across the organization.

Azure Synapse Analytics

Azure Synapse Analytics is a comprehensive, unified data analytics solution that provides a single service interface for multiple analytical capabilities, including:

- **Pipelines** - based on the same technology as Azure Data Factory.
- **SQL** - a highly scalable SQL database engine, optimized for data warehouse workloads.
- **Apache Spark** - an open-source distributed data processing system that supports multiple programming languages and APIs, including Java, Scala, Python, and SQL.
- **Azure Synapse Data Explorer** - a high-performance data analytics solution that is optimized for real-time querying of log and telemetry data using Kusto Query Language (KQL).

Azure Databricks

Azure Databricks is an Azure-integrated version of the popular Databricks platform, which combines the Apache Spark data processing platform with SQL database semantics and an integrated management interface to enable large-scale data analytics.

Azure HDInsight

Azure HDInsight is an Azure service that provides Azure-hosted clusters for popular Apache open-source big data processing technologies, including:

- **Apache Spark** - a distributed data processing system that supports multiple programming languages and APIs, including Java, Scala, Python, and SQL.
- **Apache Hadoop** - a distributed system that uses *MapReduce* jobs to process large volumes of data efficiently across multiple cluster nodes. MapReduce jobs can be written in Java or abstracted by interfaces such as Apache Hive - a SQL-based API that runs on Hadoop.
- **Apache HBase** - an open-source system for large-scale NoSQL data storage and querying.
- **Apache Kafka** - a message broker for data stream processing.
- **Apache Storm** - an open-source system for real-time data processing.

Azure Stream Analytics

- Azure Stream Analytics is a real-time stream processing engine that captures a stream of data from an input, applies a query to extract and manipulate data from the input stream, and writes the results to an output for analysis or further processing.

Azure Data Explorer

- Azure Data Explorer is a standalone service that offers the same high-performance querying of log and telemetry data as the Azure Synapse Data Explorer runtime in Azure Synapse Analytics.
- Data analysts can use Azure Data Explorer to query and analyze data that includes a timestamp attribute, such as is typically found in log files and *Internet-of-things* (IoT) telemetry data.

Microsoft Power BI

- Microsoft Power BI is a platform for analytical data modeling and reporting that data analysts can use to create and share interactive data visualizations.