

Explore fundamentals of large-scale analytics

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/examine-components-of-modern-data-warehouse/1-introduction>

Introduction

Completed

Large-scale data analytics solutions combine conventional data warehousing used to support business intelligence (BI) with techniques used for so-called "big data" analytics. A conventional data warehousing solution typically involves copying data from transactional data stores into a relational database with a schema that's optimized for querying and building multidimensional models. Big Data processing solutions, however, are used with large volumes of data in multiple formats, which is batch loaded or captured in real-time streams and stored in a *data lake* from which distributed processing engines like Apache Spark are used to process it. The combination of flexible data lake storage and data warehouse SQL analytics has led to the emergence of a large-scale analytics design often called a *data lakehouse*.

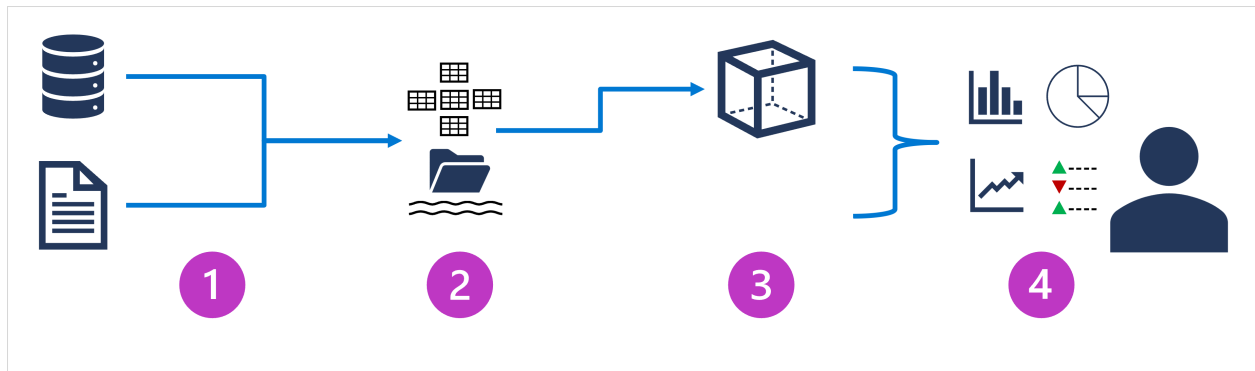
2. Describe data warehousing architecture

<https://learn.microsoft.com/en-us/training/modules/examine-components-of-modern-data-warehouse/2-describe-warehousing>

Describe data warehousing architecture

Completed

Large-scale data analytics architecture can vary, as can the specific technologies used to implement it; but in general, the following elements are included:



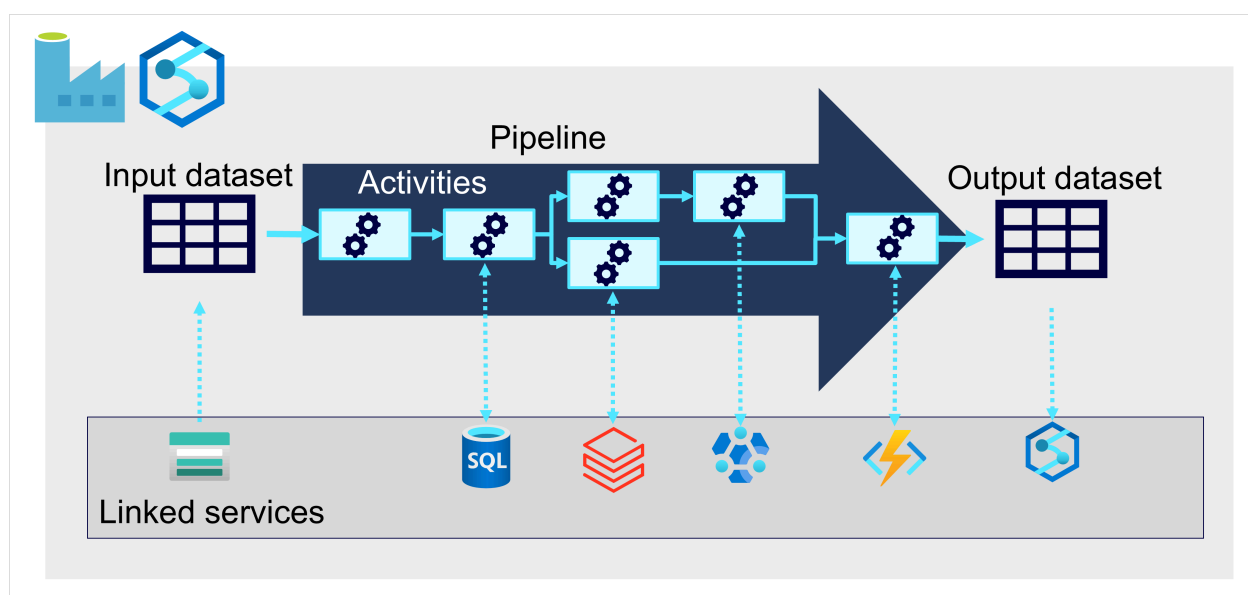
1. **Data ingestion and processing** – data from one or more transactional data stores, files, real-time streams, or other sources is loaded into a data lake or a relational data warehouse. The load operation usually involves an *extract, transform, and load* (ETL) or *extract, load, and transform* (ELT) process in which the data is cleaned, filtered, and restructured for analysis. In ETL processes, the data is transformed before being loaded into an analytical store, while in an ELT process the data is copied to the store and then transformed. Either way, the resulting data structure is optimized for analytical queries. The data processing is often performed by distributed systems that can process high volumes of data in parallel using multi-node clusters. Data ingestion includes both batch processing of static data and real-time processing of streaming data.
2. **Analytical data store** – data stores for large scale analytics include relational *data warehouses*, file-system based *data lakes*, and hybrid architectures that combine features of data warehouses and data lakes (sometimes called *data lakehouses* or *lake databases*). We'll discuss these in more depth later.
3. **Analytical data model** – while data analysts and data scientists can work with the data directly in the analytical data store, it's common to create one or more data models that pre-aggregate the data to make it easier to produce reports, dashboards, and interactive visualizations. Often these data models are described as *cubes*, in which numeric data values are aggregated across one or more dimensions (for example, to determine total sales by product and region). The model encapsulates the relationships between data values and dimensional entities to support "drill-up/drill-down" analysis.
4. **Data visualization** – data analysts consume data from analytical models, and directly from analytical stores to create reports, dashboards, and other visualizations. Additionally, users in an organization who may not be technology professionals might perform self-service data analysis and reporting. The visualizations from the data show trends, comparisons, and key performance indicators (KPIs) for a business or other organization, and can take the form of printed reports, graphs and charts in documents or PowerPoint presentations, web-based dashboards, and interactive environments in which users can explore data visually.

3. Explore data ingestion pipelines

Explore data ingestion pipelines

Completed

Now that you understand a little about the architecture of a large-scale data warehousing solution, and some of the distributed processing technologies that can be used to handle large volumes of data, it's time to explore how data is ingested into an analytical data store from one or more sources.



On Azure, large-scale data ingestion is best implemented by creating *pipelines* that orchestrate ETL processes. You can create and run pipelines using [Azure Data Factory](#), or you can use the pipeline capability in [Microsoft Fabric](#) if you want to manage all of the components of your data warehousing solution in a unified workspace.

In either case, pipelines consist of one or more *activities* that operate on data. An input dataset provides the source data, and activities can be defined as a data flow that incrementally manipulates the data until an output dataset is produced. Pipelines use *linked services* to load and process data – enabling you to use the right technology for each step of the workflow. For example, you might use an Azure Blob Store linked service to ingest the input dataset, and then use services such as Azure SQL Database to run a stored procedure that looks up related data values, before running a data processing task on Azure Databricks, or apply custom logic using an Azure Function. Finally, you can save the output dataset to a destination such as a Microsoft Fabric Lakehouse or Data Warehouse. Pipelines can also include some built-in activities, which don't require a linked service.

4. Explore analytical data stores

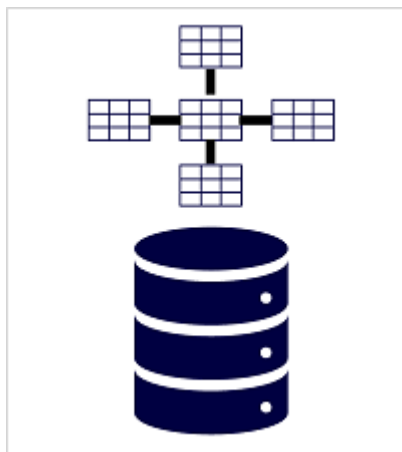
<https://learn.microsoft.com/en-us/training/modules/examine-components-of-modern-data-warehouse/4-analytical-data-stores>

Explore analytical data stores

Completed

There are two common types of analytical data store.

Data warehouses



A *data warehouse* is a relational database in which the data is stored in a schema that is optimized for data analytics rather than transactional workloads. Commonly, the data from a transactional store is transformed into a schema in which numeric values are stored in central *fact* tables, which are related to one or more *dimension* tables that represent entities by which the data can be aggregated. For example a fact table might contain sales order data, which can be aggregated by customer, product, store, and time dimensions (enabling you, for example, to easily find monthly total sales revenue by product for each store). This kind of fact and dimension table schema is called a *star schema*; though it's often extended into a *snowflake schema* by adding additional tables related to the dimension tables to represent dimensional hierarchies (for example, product might be related to product categories). A data warehouse is a great choice when you have transactional data that can be organized into a structured schema of tables, and you want to use SQL to query them.

Data lakes



A *data lake* is a file store, usually on a distributed file system for high performance data access. Technologies like Spark or Hadoop are often used to process queries on the stored files and return data for reporting and analytics. These systems often apply a *schema-on-read* approach to define tabular schemas on semi-structured data files at the point where the data is read for analysis, without applying constraints when it's stored. Data lakes are great for supporting a mix of structured, semi-structured, and even unstructured data that you want to analyze without the need for schema enforcement when the data is written to the store.

Hybrid approaches

You can use a hybrid approach that combines features of data lakes and data warehouses in a *data lakehouse*. The raw data is stored as files in a data lake, and Microsoft Fabric SQL analytics endpoints expose them as tables, which can be queried using SQL. When you create a Lakehouse with Microsoft Fabric, a SQL analytics endpoint is automatically created. Data lakehouses are a relatively new approach in Spark-based systems, and are enabled through technologies like *Delta Lake*; which adds relational storage capabilities to Spark, so you can define tables that enforce schemas and transactional consistency, support batch-loaded and streaming data sources, and provide a SQL API for querying.

Azure services for analytical stores

On Azure, there are several services that you can use to implement a large-scale analytical store including:



Microsoft Fabric is a unified, end-to-end solution for large scale data analytics. It brings together multiple technologies and capabilities, enabling you to combine the data integrity and reliability of a scalable, high-performance SQL Server based relational data warehouse with the flexibility of a data lake and open-source Apache Spark. It also includes native support for log and telemetry analytics with Microsoft Fabric Real-Time Intelligence, as well as built in data pipelines for data ingestion and transformation. Each Microsoft Fabric product experience has its own home, for

example the Data Factory Home. Each Fabric Home displays the items that you create and have permission to use from all the workspace that you access. Microsoft Fabric is a great choice when you want to create a single, unified analytics solution.



Azure Databricks is an Azure implementation of the popular Databricks platform.

Databricks is a comprehensive data analytics solution built on Apache Spark, and offers native SQL capabilities as well as workload-optimized Spark clusters for data analytics and data science.

Databricks provides an interactive user interface through which the system can be managed and data can be explored in interactive notebooks. Due to its common use on multiple cloud platforms, you might want to consider using Azure Databricks as your analytical store if you want to use existing expertise with the platform or if you need to operate in a multicloud environment or support a cloud-portable solution.

Note

Each of these services can be thought of as an analytical data *store*, in the sense that they provide a schema and interface through which the data can be queried. In many cases however, the data is actually stored in a data lake and the service is used to *process* the data and run queries. Some solutions might even combine the use of these services. An *extract, load, and transform* (ELT) ingestion process might copy data into the data lake, and then use one of these services to transform the data, and another to query it. For example, a pipeline might use a notebook running in Azure Databricks to process a large volume of data in the data lake, and then load it into tables in a Microsoft Fabric Warehouse.

5. Exercise: Explore data analytics in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/examine-components-of-modern-data-warehouse/5-exercise-data-analytics-fabric>

Exercise: Explore data analytics in Microsoft Fabric

Completed

In this exercise, you create a Microsoft Fabric data lakehouse and use it to ingest and analyze some data.

The exercise is designed to familiarize you with some key elements of a large-scale data analytics solution, not as a comprehensive guide to performing advanced data analysis with Microsoft Fabric. The exercise should take around 30 minutes to complete.

Note

To complete this lab, you will need a Microsoft Fabric account, or if you don't have an account yet, sign up for a [free trial](#).

Go to [Getting started with Fabric](#) for more information.

Launch the exercise and follow the instructions.

Launch Exercise

6. Knowledge check

<https://learn.microsoft.com/en-us/training/modules/examine-components-of-modern-data-warehouse/6-knowledge-check>

Knowledge check

Completed

7. Summary

<https://learn.microsoft.com/en-us/training/modules/examine-components-of-modern-data-warehouse/7-summary>

Summary

Completed

Large-scale data warehousing is a complex workload that can involve many different technologies. This module provided a high-level overview of the key features of a modern data warehousing solution, and explored some of the services in Azure that you can use to implement one.

In this module, you learned how to:

- Identify common elements of a large-scale data warehousing solution
- Describe key features for data ingestion pipelines
- Identify common types of analytical data store and related Azure services
- Provision Microsoft Fabric and use it to ingest, process, and query data

Next steps

Now that you learned about large-scale data warehousing, consider learning more about data-related workloads on Azure by pursuing a Microsoft certification in [Azure Data Fundamentals](#).