Machine Learning and Data Mining

Last update: 14 October 2023

Academic Year 2023 – 2024
Alma Mater Studiorum · University of Bologna

Contents

Acronyms

BI Business Intelligence

CDC Change Data Capture

DFM Dimensional Fact Model

DM Data Mart

DSS Decision Support System

DWH Data Warehouse

EIS Executive Information System

ERP Enterprise Resource Planning

ETL Extraction, Transformation, Loading

MIS Management Information System

OLAP Online Analysical Processing

OLTP Online Transaction Processing

1 Introduction

1.1 Data

Data Collection of raw values. Data

Information Organized data (e.g. relationships, context, ...). Information

Knowledge Understanding information. Knowledge

1.1.1 Data sources

Transaction Business event that generates or modifies data in an information system (e.g. database). Transaction

Signal Measure produced by a sensor. Signal

External subjects

1.1.2 Software

Online Transaction Processing (OLTP) Class of programs to support transaction oriented applications and data storage. Suitable for real-time applications. Online Transaction Processing

Enterprise Resource Planning (ERP) Integrated system to manage all the processes of a business. Uses a shared database for all applications. Suitable for real-time applications. Enterprise Resource Planning

1.1.3 Insight

Decision can be classified as:

- **Structured** Established and well understood situations. What is needed is known. Structured decision
- **Unstructured** Unplanned and unclear situations. What is needed for the decision is unknown. Unstructured decision

Different levels of insight can be extracted by:

- **Management Information System (MIS)** Standardized reporting system built on existing OLTP. Used for structured decisions. Management Information System
- **Decision Support System (DSS)** Analytical system to provide support for unstructured decisions. Decision Support System
- **Executive Information System (EIS)** Formulate high level decisions that impact the organization. Executive Information System
- **Online Analysical Processing (OLAP)** Grouped analysis of multidimensional data. Involves large amount of data. Online Analysical Processing

Business Intelligence (BI) Applications, infrastructure, tools and best practices to analyze information. Business Intelligence

Big data Large and/or complex and/or fast changing collection of data that traditional DBMSs are unable to process. Big data

Structured e.g. relational tables.

Unstructured e.g. videos.

Semi-structured e.g. JSON.

Anaylitics Structured decision driven by data. Anaylitics

Data mining Discovery process for unstructured decisions. Data mining

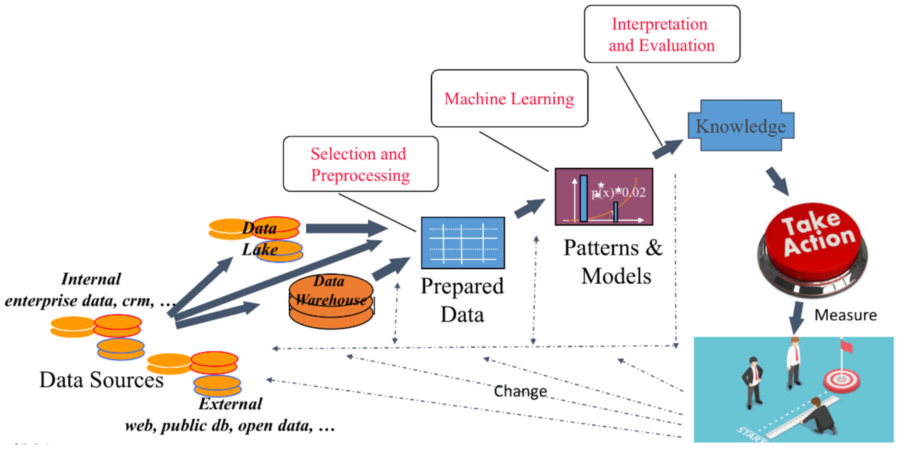


Figure 1.1: Data mining process

Machine learning Learning models and algorithms that allow to extract patterns from data. Machine learning

2 Business Intelligence

Business Intelligence

Business Intelligence Transform raw data into information. Deliver the right information to the right people at the right time through the right channel.

Data Warehouse

Data Warehouse (DWH) Optimized repository that stores information for decision making processes. DWHs are a specific type of DSS.

Features:

- Subject-oriented: focused on enterprise specific concepts.
- Integrates data from different sources and provides an unified view.
- Non-volatile storage with change tracking.

Data Mart

Data Mart (DM) Subset of the primary DWH with information relevant to a specific business area.

2.1 Online Analytical Processing (OLAP)

Online Analytical Processing (OLAP)

OLAP analyses Able to interactively navigate the information in a data warehouse. Allows to visualize different levels of aggregation.

OLAP session Navigation path created by the operations that a user applied.

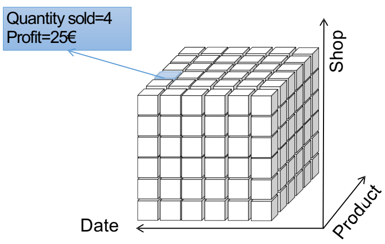
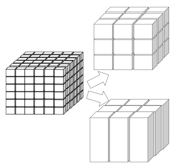


Figure 2.1: OLAP data cube

2.1.1 Operators

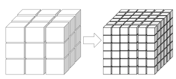
Roll-up

Roll-up Increases the level of aggregation (i.e. `GROUP BY` in SQL). Some details are collapsed together.

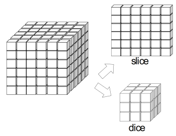


Drill-down

Drill-down Reduces the level of aggregation. Some details are reintroduced.

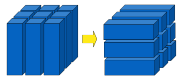


Slide-and-dice The slice operator reduces the number of dimensions (i.e. drops columns).
The dice operator reduces the number of data being analyzed (i.e. `LIMIT` in SQL).



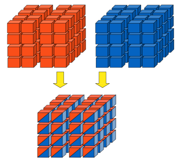
Slide-and-dice

Pivot Changes the layout of the data, to analyze it from a different viewpoint.



Pivot

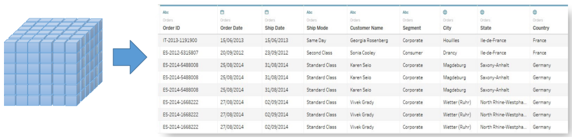
Drill-across Links concepts from different data sources (i.e. `JOIN` in SQL).



Drill-across

Drill-through Switches from multidimensional aggregated data to operational data (e.g. a spreadsheet).

Drill-through



2.2 Extraction, Transformation, Loading (ETL)

The ETL process extracts, integrates and cleans operational data that will be loaded into a data warehouse.

Extraction, Transformation, Loading (ETL)

2.2.1 Extraction

Extracted operational data can be:

Structured with a predefined data model (e.g. relational DB, CSV)

Strucured data

Untructured without a predefined data model (e.g. social media content)

Unstrucured data

Extraction can be of two types:

Static The entirety of the operational data are extracted to populate the data warehouse for the first time.

Static extraction

Incremental Only changes applied since the last extraction are considered. Can be based on a timestamp or a trigger.

Incremental extraction

2.2.2 Cleaning

Operational data may contain:

Duplicate data

Missing data

Improper use of fields (e.g. saving the phone number in the `notes` field)

Wrong values (e.g. 30th of February)

Inconsistency (e.g. use of different abbreviations)

Typos

Methods to clean and increase the quality of the data are:

Dictionary-based techniques Lookup tables to substitute abbreviations, synonyms or typos. Applicable if the domain is known and limited. Dictionary-based cleaning

Approximate merging Merging data that do not have a common key. Approximate merging

Approximate join Use non-key attributes to join two tables (e.g. using the name and surname instead of an unique identifier).

Similarity approach Use similarity functions (e.g. edit distance) to merge multiple instances of the same information (e.g. typo in customer surname).

Ad-hoc algorithms Ad-hoc algorithms

2.2.3 Transformation

Data are transformed to respect the format of the data warehouse:

Conversion Modifications of types and formats (e.g. date format) Conversion

Enrichment Creating new information by using existing attributes (e.g. compute profit from receipts and expenses) Enrichment

Separation and concatenation Denormalization of the data: introduces redundances (i.e. breaks normal form[1]) to speed up operations. Separation and concatenation

2.2.4 Loading

Adding data into a data warehouse:

Refresh The entire DWH is rewritten. Refresh loading

Update Only the changes are added to the DWH. Old data are not modified. Update loading

2.3 Data warehouse architectures

The architecture of a data warehouse should meet the following requirements:

Separation Separate the analytical and transactional workflows.

Scalability Hardware and software should be easily upgradable.

Extensibility Capability to host new applications and technologies without the need to redesign the system.

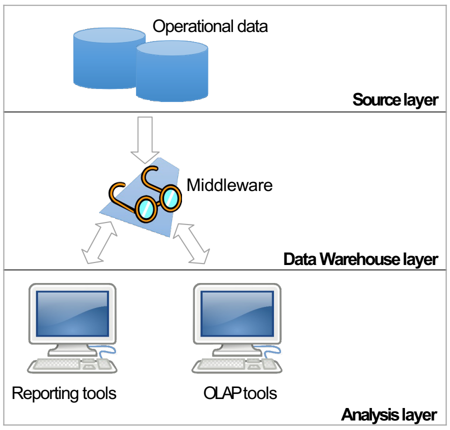
Security Access control.

Administrability Easily manageable.

[1] `https://en.wikipedia.org/wiki/Database_normalization`

2.3.1 Single-layer architecture

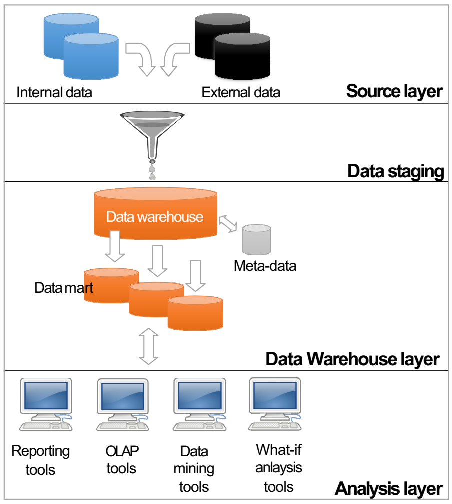
- Minimizes the amount of data stored (i.e. no redundancies).
- The source layer is the only physical layer (i.e. no separation).
- A middleware provides the DWH features.



Single-layer architecture

2.3.2 Two-layer architecture

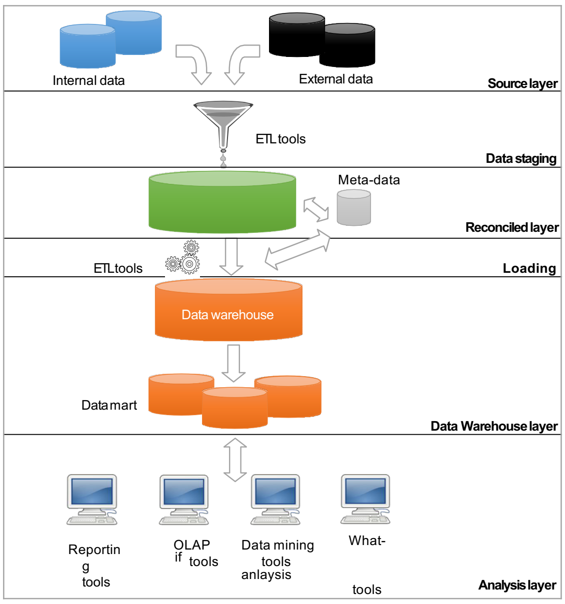
- Source data (source layer) are physically separated from the DWH (data warehouse layer).
- A staging layer applies ETL procedures before populating the DWH.
- The DWH is a centralized repository from which data marts can be created. Metadata repositories store information on sources, staging and data marts schematics.



Two-layer architecture

2.3.3 Three-layer architecture

- A reconciled layer enhances the cleaned data coming from the staging step by adding enterprise-level details (i.e. adds more redundancy before populating the DWH).



Three-layer architecture

2.4 Conceptual modeling

Dimensional Fact Model (DFM) Conceptual model to support the design of data marts. The main concepts are:

Dimensional Fact Model (DFM)

- **Fact** Concept relevant to decision-making processes (e.g. sales).
- **Measure** Numerical property to describe a fact (e.g. profit).
- **Dimension** Property of a fact with a finite domain (e.g. date).
- **Dimensional attribute** Property of a dimension (e.g. month).
- **Hierarchy** A tree where the root is a dimension and nodes are dimensional attributes (e.g. date → month).
- **Primary event** Occurrence of a fact. It is described by a tuple with a value for each dimension and each measure.
- **Secondary event** Aggregation of primary events. Measures of primary events are aggregated if they have the same (preselected) dimensional attributes.

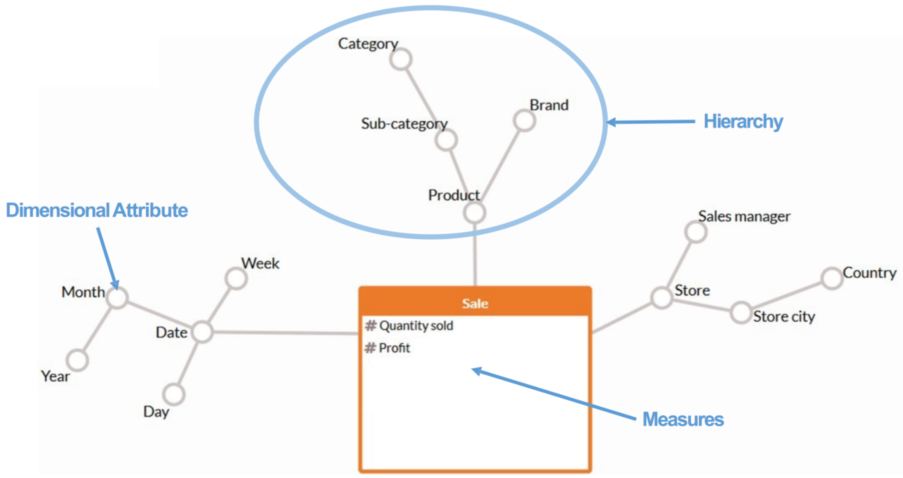


Figure 2.2: Example of DFM

Primary events

Date	Store	Product	Qty sold	Profit
01/03/15	Central store	Milk	20	60
01/03/15	Central store	Coke	25	50
02/03/15	Central store	Bread	40	70
10/03/15	Central store	Wine	15	150

SUM SUM

Secondary event

Month	Store	Category	Qty sold	Profit
March 2015	Central store	Food and Beverages	100	330

Figure 2.3: Example of primary and secondary events

2.4.1 Aggregation operators

Measures can be classified as:

- **Flow measures** Evaluated cumulatively with respect to a time interval (e.g. quantity sold).

  Flow measures

- **Level measures** Evaluated at a particular time (e.g. number of products in inventory).

  Level measures

- **Unit measures** Evaluated at a particular time but expressed in relative terms (e.g. unit price).

  Unit measures

Aggregation operators can be classified as:

- **Distributive** Able to calculate aggregates from partial aggregates (e.g. `SUM`, `MIN`, `MAX`).

  Distributive operators

- **Algebraic** Requires a finite number of support measures to compute the result (e.g. `AVG`).

  Algebraic operators

- **Holistic** Requires an infinite number of support measures to compute the result (e.g. `RANK`).

  Holistic operators

Additivity A measure is additive along a dimension if an aggregation operator can be applied.

Additive measure

	Temporal hierarchies	Non-temporal hierarchies
Flow measures	SUM, AVG, MIN, MAX	SUM, AVG, MIN, MAX
Level measures	AVG, MIN, MAX	SUM, AVG, MIN, MAX
Unit measures	AVG, MIN, MAX	AVG, MIN, MAX

Table 2.1: Allowed operators for each measure type

2.4.2 Logical design

Defining the data structures (e.g. tables and relationships) according to a conceptual model. There are mainly two strategies:

Logical design

- **Star schema** A fact table that contains all the measures and linked to dimensional tables.

  Star schema

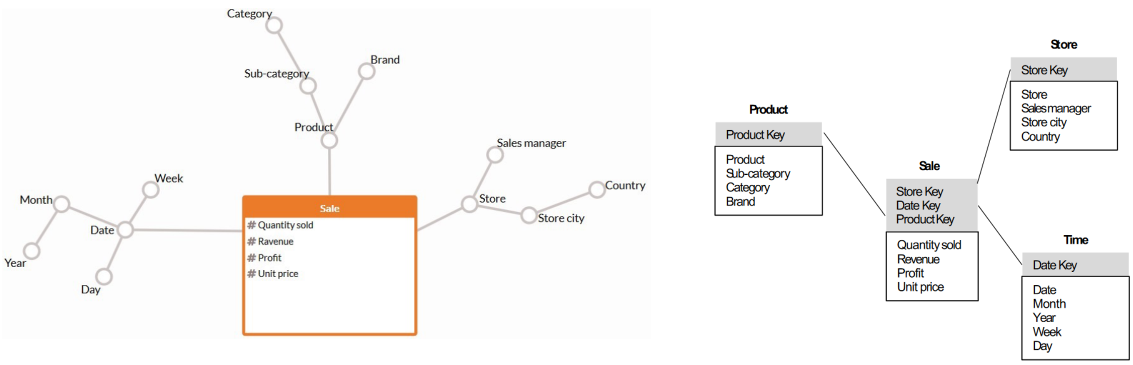


Figure 2.4: Example of star schema

- **Snowflake schema** A star schema variant with partially normalized dimension tables.

  Snowflake schema

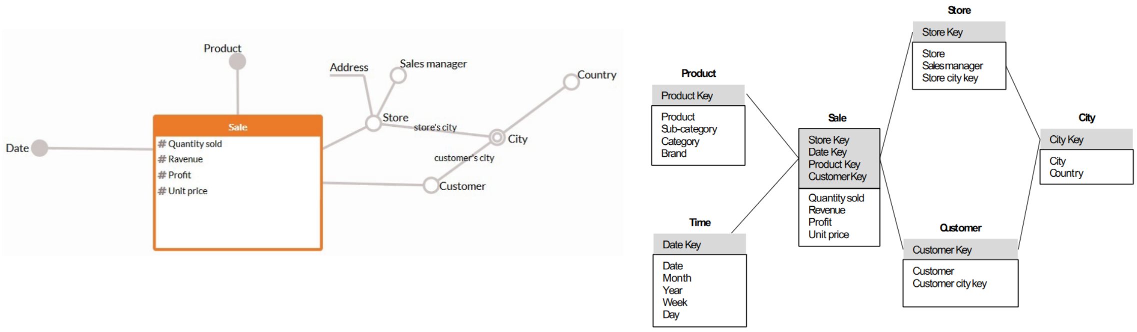


Figure 2.5: Example of snowflake schema

2.5 Data lake

Dark data Acquired and stored data that are never used for decision-making processes.

Dark data

Data lake Repository to store raw (unstructured) data. It has the following features:

Data lake

- Does not enforce a schema on write.
- Allows flexible access and applies schemas on read.
- Single source of truth.
- Low cost and scalable.

Storage Stored data can be classified as:

Hot A low volume of highly requested data that require low latency. More expensive HW/SW.

Hot storage

Cold A large amount of data that does not have latency requirements. Less expensive.

Cold storage

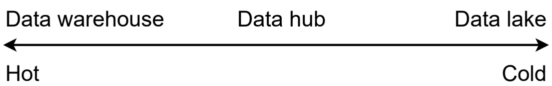


Figure 2.6: Data storage technologies

2.5.1 Traditional vs insight-driven data systems

	Traditional (data warehouse)	Insight-driven (data lake)
Sources	Structured data	Structured, semi-structured and unstructured data
Storage	Limited ingestion and storage capability	Virtually unlimited ingestion and storage capability
Schema	Schema designed upfront	Schema not fixed
Transformations	ETL upfront	Transformations on query
Analytics	SQL, BI tools, full-text search	Traditional methods, self-service BI, big data, machine learning, ...
Price **Performance**	High storage cost Fast queries	Low storage cost Scalability/speed/cost tradeoffs
Quality	High data quality	Depends on the use case

2.5.2 Data architecture evolution

Traditional data warehouse (i.e. in-house data warehouse) Traditional data warehouse

- Structured data with predefined schemas.
- High setup and maintenance cost. Not scalable.
- Relational high-quality data.
- Slow data ingestion.

Modern cloud data warehouse Modern cloud data warehouse

- Structured and semi-structured data.
- Low setup and maintenance cost. Scalable and easier disaster recovery.
- Relational high-quality data and mixed data.
- Fast data ingestion if supported.

On-premise big data (i.e. in-house data lake) On-premise big data

- Any type of data with schemas on read.
- High setup and maintenance cost.
- Fast data ingestion.

Cloud data lake Cloud data lake

- Any type of data with schemas on read.
- Low setup and maintenance cost. Scalable and easier disaster recovery.
- Fast data ingestion.

2.5.3 Components

Data ingestion Data ingestion

Workload migration Inserting all the data from an existing source.

Incremental ingestion Inserting changes since the last ingestion.

Streaming ingestion Continuously inserting data.

Change Data Capture (CDC) Mechanism to detect changes and insert the new data into the data lake (possibly in real-time). Change Data Capture (CDC)

Storage

Raw Immutable data useful for disaster recovery. Raw storage

Optimized Optimized raw data for faster query. Optimized storage

Analytics Ready to use data. Analytics storage

Columnar storage

- Homogenous data are stores contiguously.
- Speeds up methods that process entire columns (i.e. all the values of a feature).
- Insertion becomes slower.

Data catalog Methods to add descriptive metadata to a data lake. This is useful to prevent an unorganized data lake (data swamp).

Processing and analytics

Processing and analytics

Interactive analytics Interactive queries to large volumes of data. The results are stored back in the data lake.

Big data analytics Data aggregations and transformations.

Real-time analytics Streaming analysis.

2.5.4 Architectures

Lambda lake

Lambda lake

Batch layer Receives and stores the data. Prepares the batch views for the serving layer.

Serving layer Indexes batch views for faster queries.

Speed layer Receives the data and prepares real-time views. The views are also stored in the serving layer.

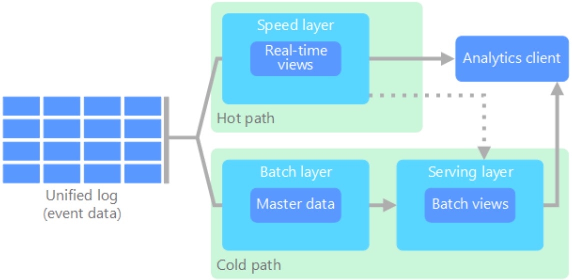


Figure 2.7: Lambda lake architecture

Kappa lake The data are stored in a long-term store. Computations only happen in the speed layer (avoids lambda lake redundancy between batch layer and speed layer).

Kappa lake

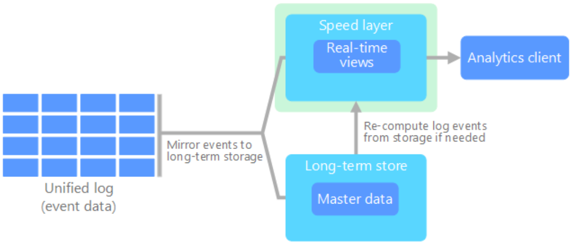


Figure 2.8: Kappa lake architecture

Delta lake Framework that adds features on top of an existing data lake.

Delta lake

- ACID transactions
- Scalable metadata handling
- Data versioning
- Unified batch and streaming
- Schema enforcement

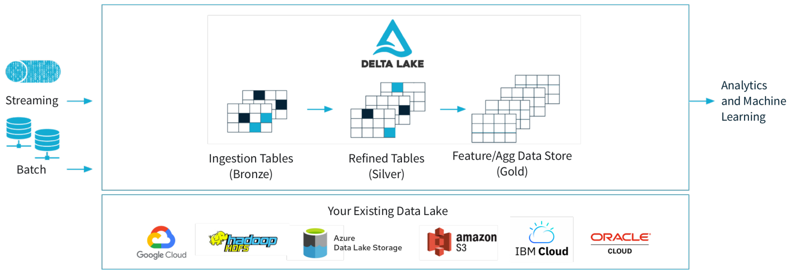


Figure 2.9: Delta lake architecture

2.5.5 Metadata

Metadata are used to organize a data lake. Useful metadata are: Metadata

Source Origin of the data.

Schema Structure of the data.

Format File format or encoding.

Quality metrics (e.g. percentage of missing values).

Lifecycle Retention policies and archiving rules.

Ownership

Lineage History of applied transformations or dependencies.

Access control

Classification Sensitivity level of the data.

Usage information Record of who accessed the data and how it is used.