

# Database Management Systems

Ivan Heibi

[ivan.heibi2@unibo.it](mailto:ivan.heibi2@unibo.it) – <https://orcid.org/0000-0001-5366-5194> – <https://ivanhb.it>

Computational Management of Data – Part II (A.Y. 2025/2026)  
Second Cycle Degree in Digital Humanities and Digital Knowledge  
Alma Mater Studiorum - Università di Bologna

# Summary of the previous lectures (1/2)

A datum is a declarative statement **subject-predicate-object** that, through the **predicate**, either **attributes** a **literal** (i.e. a value such as a string, a number, etc.) to a **subject entity** or it **relates** such a **subject entity** with **another entity**

Each entity, being used either as **subject** or **object** of a statement, is characterised by a **unique identifier**

The **same entity** can be used as **subject** or **object** in one or more data, while a literal **cannot be used** as **subject** in any datum

An attribute is intrinsically **part of** the **entity** to which it is associated – modifying the value of an attribute affect **only** the **entity** to which it refers to

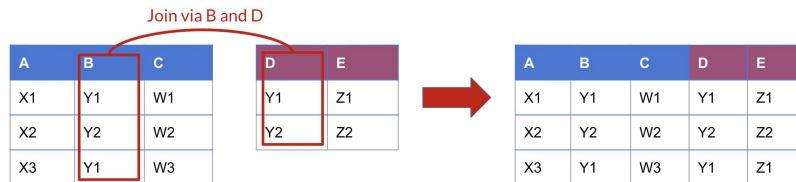
A **data model** is an abstract, simplified and formal representation of some data related to a system or a real domain, and enables us to describe what a data collection is about and to check data correctness

A **data model** permit one to specify classes of entities, their attributes and relations

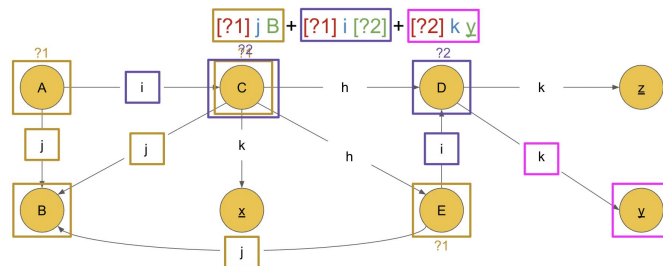
# Summary of the previous lectures (2/2)

Depending on the structure in which data are stored , queries to datasets are approached differently:

- With **tabular data**, often you have to combine tables between them to obtain bigger tables which contain the query requirements and the related answer



- With **graph data**, you explore the graph starting from fixed points (i.e. known entities, values, predicates) to find a pattern that is compliant with the query



We use **descriptive statistics** as methods to summarize and describe data in a quantitative way – (measures of central tendency) Mean; Median; Mode, (measures of variability) minimum, maximum, and standard deviation

# What is a database (DB)

When thinking about the world of computing, we define a **database** as a collection of data which organised, stored and accessed electronically

All data in a database are organised according to database model, that defines the particular structure used to organise the data (e.g. relational and graph-based)

To design a database, one has to pass through a series of steps:

1. **define the data model** to describe a kind of data to handle
2. **create the database** structures (dependent on the particular database model in consideration) to map the data model objects (i.e. classes, attributes and relations)
3. **upload data into the database** according to the related database structures

# What is a database (DB)

The creation of the database structures, the mechanism to upload new data into the database, and all the facilities for querying database data, are enabled by the **database management system (DBMS)**

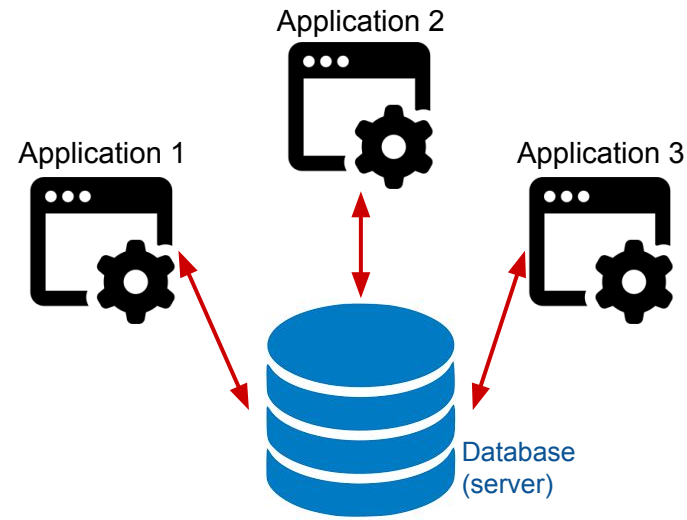
A DBMS is the software which enables actors (users, applications, software agents) to interact with the database itself, and makes available tools to query and manage the database

In particular, a **DBMS** allows one to:

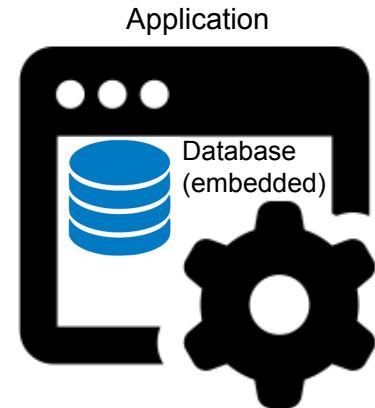
- **Create**, modify, and remove the structures that define the organisation of the data
- **Insert**, modify, and delete data in/from the database
- **Retrieve** the data that can be returned to the user either in the form as they are stored in the database or according to new formats
- **Register** users, monitor and maintain performance and data integrity, and recovering corrupted information

# Standalone vs embedded DBMS

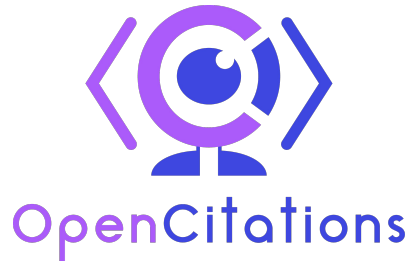
Most of the DBMS available are actually developed as **standalone** applications – thus, acting as servers that store data and execute queries when requested by one or more client applications



**Embedded** DBMS, instead, are particular software providing tools for creating and managing databases that are directly embedded within applications – and, thus, can be considered proper libraries of such applications



# DB and DBMS: some examples



**DB**



**DBMS**

Virtuoso: <https://virtuoso.openlinksw.com/>

MySQL: <https://www.mysql.com>

OpenCitations: <http://opencitations.net>

Wikibase: <https://wikiba.se>

Wikidata: <https://www.wikidata.org>

# Transactions

A **transaction** is a unit of work performed within a DBMS against a database

Usually, it represents any change in a database

Every transaction is composed by one or more operations – e.g. think about a money transfer from a bank account to another (subtract the amount from the source bank adding it to the second bank)

Transactions are one of the main components that enables the integrity of data, being them executed fully or causing a rollback of the database

# ACID properties

Any transaction is compliant with the following four properties:

- **A**tomicity – a transaction is irreducible and indivisible sequence of operations on a database such that either all occurs or nothing occurs
- **C**onsistency – a transaction must change a database accordingly to all defined rules and constraints the database defines
- **I**solation – it enables the correct execution of concurrent transactions, thus leaving the database in the same state that would have been obtained if the transactions were executed sequentially
- **D**urability – a transaction that has been committed will survive permanently even in the case the system crashes

# Relational database

A database is relational when it is based on the relational model of data

A relational data model structures data using tables of columns and rows, where each row (or record) is identified by a unique key

Usually, each table represents one type of entity (e.g. publication or venue), while

- its rows indicates instances of that type of entity (e.g. the article with DOI “10.1162/qss\_a\_00023” or the venue “Quantitative Science Studies”)
- its columns representing values attributed to that instance

A table may specify that certain columns act as keys for entities

- A **primary key** is one or more columns that **uniquely identify** an entity in a table
- A **foreign key** is one or more columns that **refer to** the primary key of the same or another table

# Example of primary and foreign keys

Publication

internalId	doi	title	publicationYear	publicationVenue
Entity 1	<u>"10.1016/s1367-5931(02)00332-0"</u>	<u>"In vitro selection..."</u>	<u>2002</u>	Entity 2

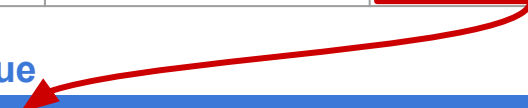
Primary  
key

Foreign  
key

Venue

internalId	id	name
Entity 2	<u>"1367-5931"</u>	<u>"Current Opinion..."</u>

Primary  
key



# List of Relational DBMS

Oracle Database – <https://www.oracle.com/it/database/>

MySQL – <https://www.mysql.com>

Microsoft SQL Server – <https://www.microsoft.com/en-gb/sql-server>

PostgreSQL – <https://www.postgresql.org/>

IBM Db2 – <https://www.ibm.com/uk-en/analytics/db2>

SQLite – <https://www.sqlite.org>

MariaDB – <https://mariadb.org/>

# Graph database

A database is graph-based when it uses a graph structure for representing data

Data are represented as a collection of nodes and edges, where the edges represent the relationships between the nodes (entities)

Graph databases defines data as they are defined conceptually, where

- nodes represent **entities** (publications, venues, etc.)
- edges are the mechanism to connect nodes, and can be **undirected or directed** depending on the particular approach adopted by the database
- properties are **information associated to** nodes

# Resource Description Framework (RDF)

RDF is a data model based on triples subject-predicate-object called statements and used in several graph databases



# List of graph databases

Blazegraph – <https://blazegraph.com/>

Fuseki – <https://jena.apache.org/documentation/fuseki2/>

Neo4j – <https://neo4j.com/>

Ontotext GraphDB – <https://graphdb.ontotext.com/>

OpenLink Virtuoso – <https://virtuoso.openlinksw.com/>

# END

**Ivan Heibi**

[ivan.heibi2@unibo.it](mailto:ivan.heibi2@unibo.it) – <https://orcid.org/0000-0001-5366-5194> – <https://ivanhb.it>

Computational Management of Data – Part II (A.Y. 2025/2026)  
Second Cycle Degree in Digital Humanities and Digital Knowledge  
Alma Mater Studiorum - Università di Bologna