LECTURER: DR. TAI LE QUY

ANALYTICAL SOFTWARE AND FRAMEWORKS

Who am I?

- Name: Tai Le Quy
- PhD at L3S Research Center – Leibniz University Hannover
  - Research topic: Fairness-aware machine learning in educational data mining
- MSc in Information Technology at National University of Vietnam
- Profile: tailequy.github.io
- Email: tai.le-quy@iu.org
- Materials: https://github.com/tailequy/IU-SoftwareFrameworks

Who are you?

- Name
- Employer
- Position/responsibilities
- Fun Fact
- Previous knowledge? Expectations?

TOPIC OUTLINE

UNIT 1

INTRODUCTION TO ANALYTICAL SOFTWARE AND FRAMEWORKS

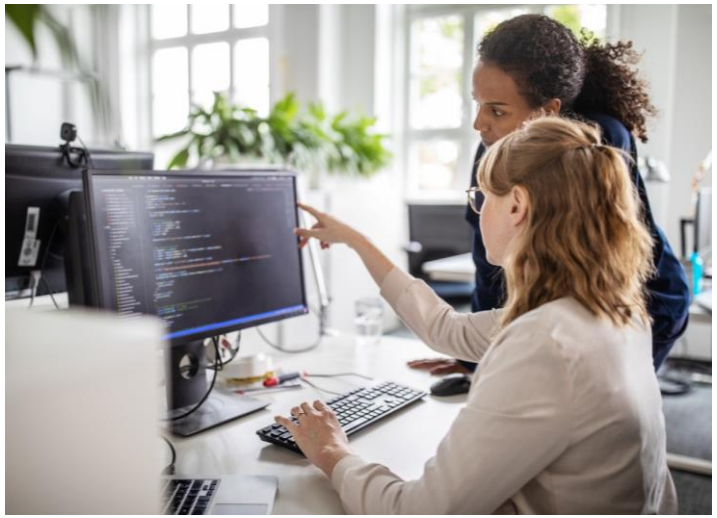
STUDY GOALS

- Understand what is meant by a software system and a framework
- Understand how frameworks support implementing data analytics
- Have an overview of advantages and challenges of distributed computing systems
- Understand how data warehousing differs from transactional database architectures

EXPLAIN SIMPLY

1. What is a software system?
2. How do frameworks and libraries support data analytics?
3. What are benefits and challenges of distributed computing systems?
4. What separates data warehousing from other database architectures?

A Software System is ***"a system made up of software, hardware, and data that provides its primary value by the execution of the software."***



- In data analytics we use AND (may) build software systems, at least if it's not just about one-off tasks/analyses.

The main objectives of choosing a software system:

- achieve higher level of productivity,
- gain greater insight into a specific application, and
- obtain advantage of new opportunities.

Source of the text: Object Management Group (OMG), 2018, p. 5.
Source of the images: Office356 archive.

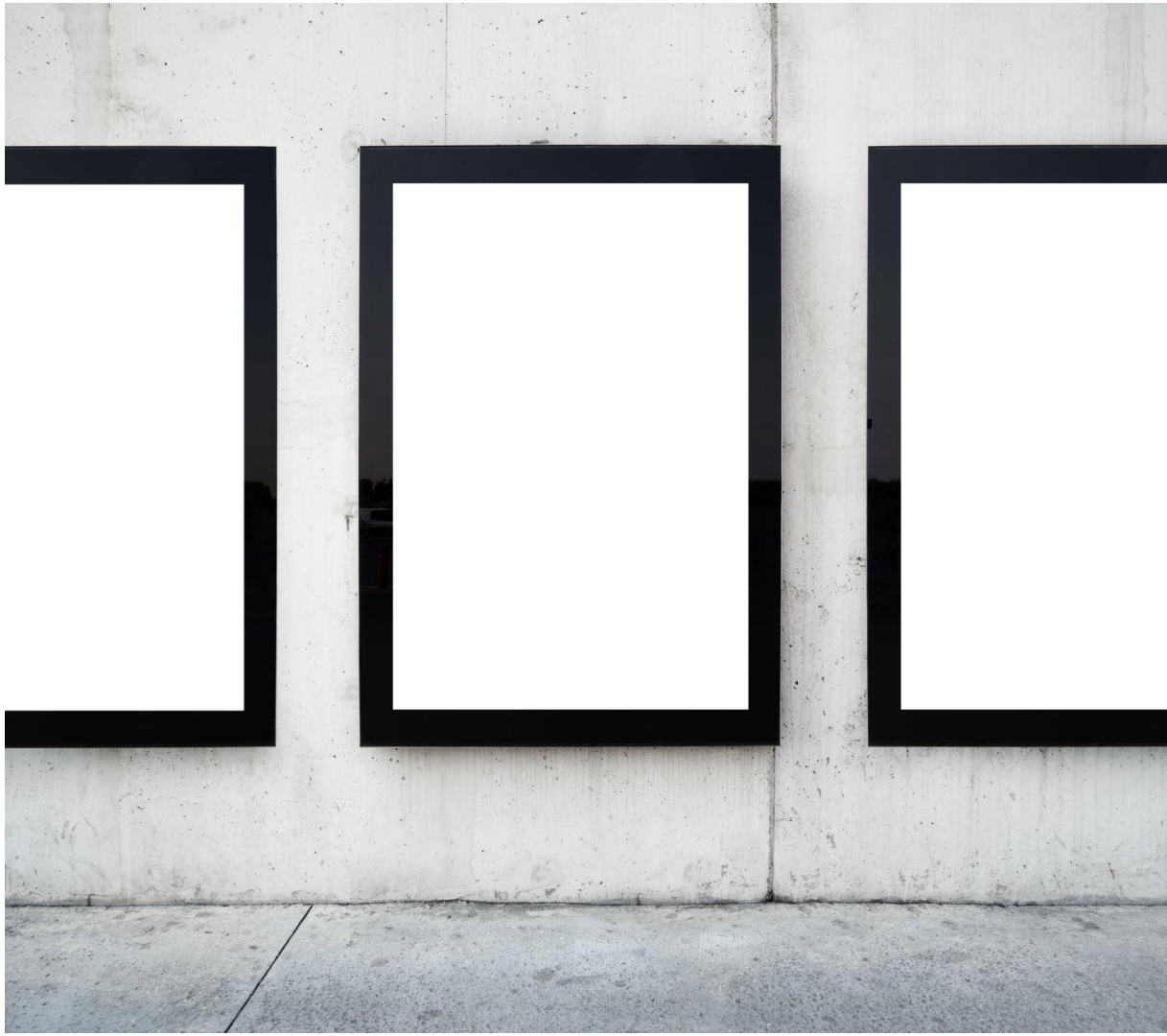
EXAMPLE OF SOFTWARE SYSTEMS

Software system	Description	Examples
Customer relationship management (CRM)	Store and analyze important customers' interactive data and facilitate monitoring of sales processes	Salesforce, HubSpot, Monday sale CRM
Enterprise resource planning (ERP)	Simplify management of dozens of projects and resources, and to ensure an effective, real-time allocation of elements in the processes	Microsoft Dynamics 365, Oracle Netsuite, SAP S/4HANA
Project management	Manage resources and execute related projects: scheduling, cost analysis, etc.	Monday, Microsoft Project, Asana
Business process management (BPM)	Analyze complex data and quick automation of strategic business processes through web-based modeling and a user interface	Oracle BPM, MS Power Automat, IBM Business Automation Workflow
Database management	Allows user interaction to create and modify the associated business's data	Oracle , MySQL, Microsoft SQL Server, MongoDB
Scheduling management	Assign jobs and create machine schedules.	Monday.com, Workday

Source of the text: course book

FRAMEWORK

“A model of a particular domain or an important aspect thereof providing a reusable design and implementation to be extended and customized by clients to produce specialized applications.”



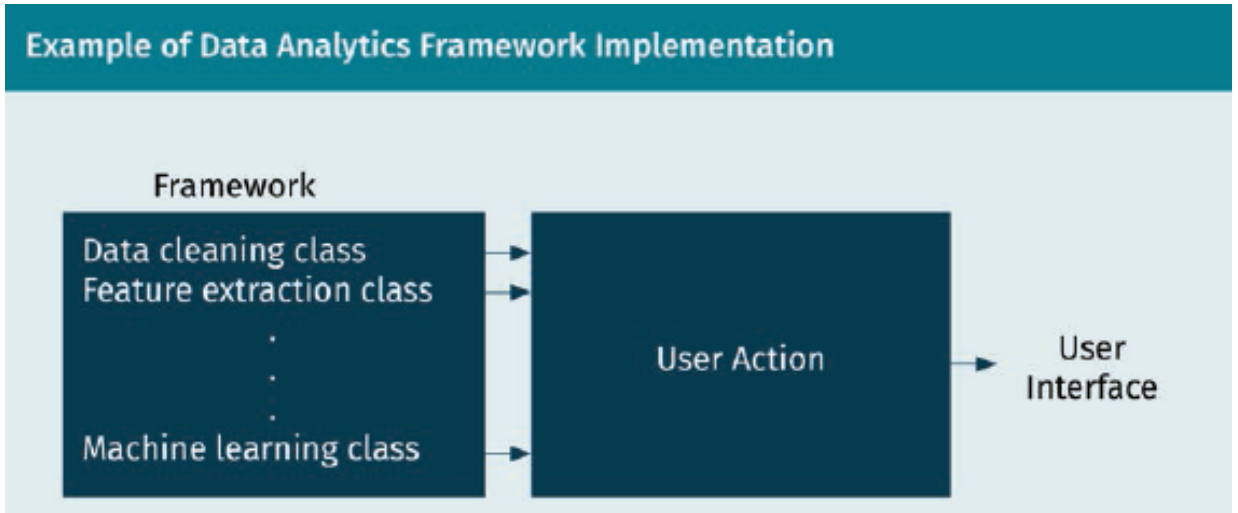
Source of the text: cf. Riehle, 2000, p. 54.
Source of the image: Office365 archive.

SOFTWARE FRAMEWORK

- A software framework is a **reusable**, **semi-complete** application that can be customized to produce specialized applications.
- It contains a collection of classes, which are object-oriented code modules written in a specific programming language. There are a set of actions spread over these classes, which may require further subclasses to be performed.
- A developer will normally **customize** a framework to a specific application by “subclassing” and composing instances of framework classes
- In conventional subroutine calls, we write the main procedure and call the code we want to reuse. However, in frameworks, we reuse the main procedure and write the code it calls.

Examples in data analytics

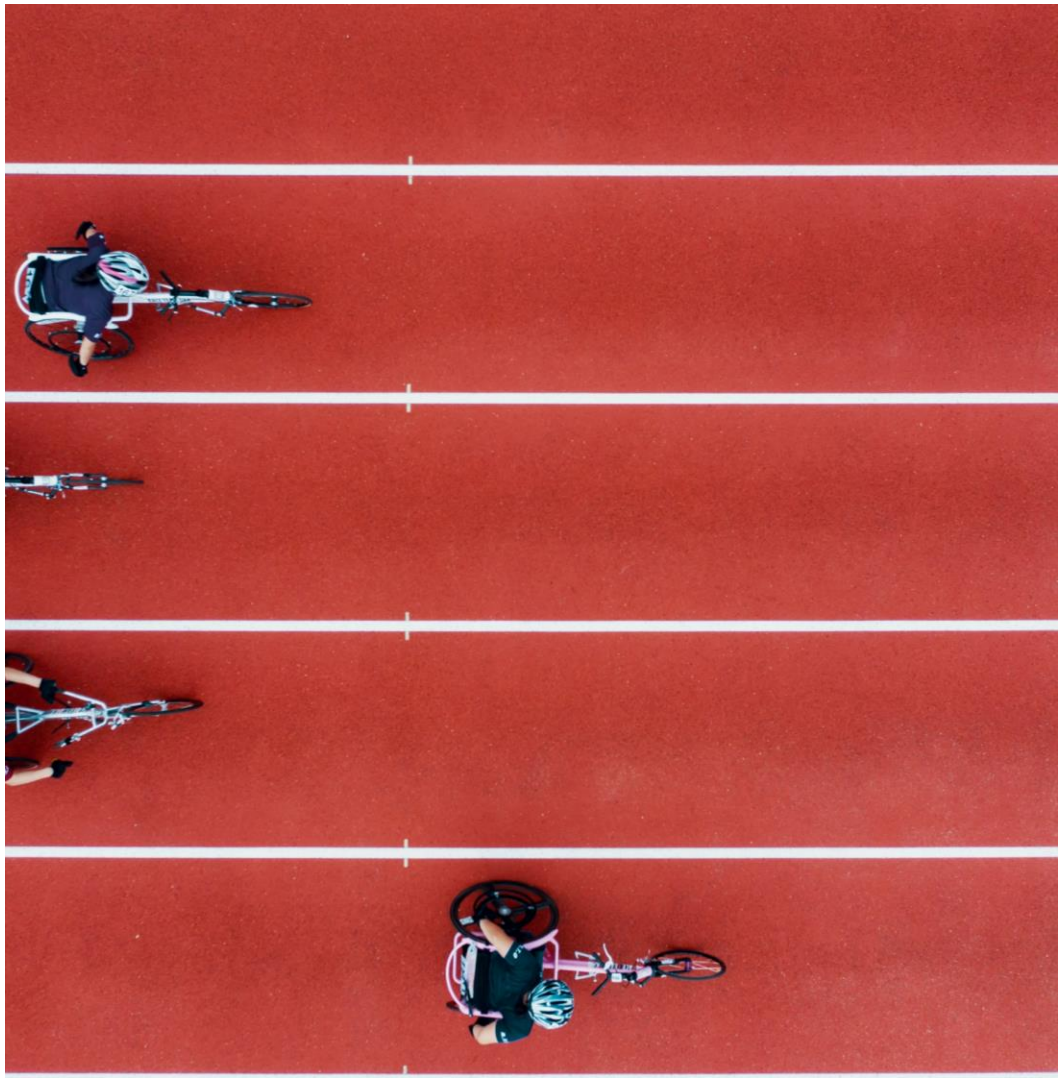
- Function of a framework
  - manage information
  - manipulate data
  - construct visuals
  - coordinate resources
  - calculate statistics
  - produce learning models



Source of the image: Course book

Significantly **reduce effort and accelerate development** of high-quality software systems

- Built to be ***reused***, to be adapted and extended where needed
- ***Simplify*** development by defining conventions, restrictions, and guidelines
- ***Reduce*** *code writing* focused on higher level functionality
- ***Embed knowledge and best practices***



Source of the text: Course book DLMBDSA02, p. 15.
Source of the image: Office365 archive.

FRAMEWORK PROPERTIES

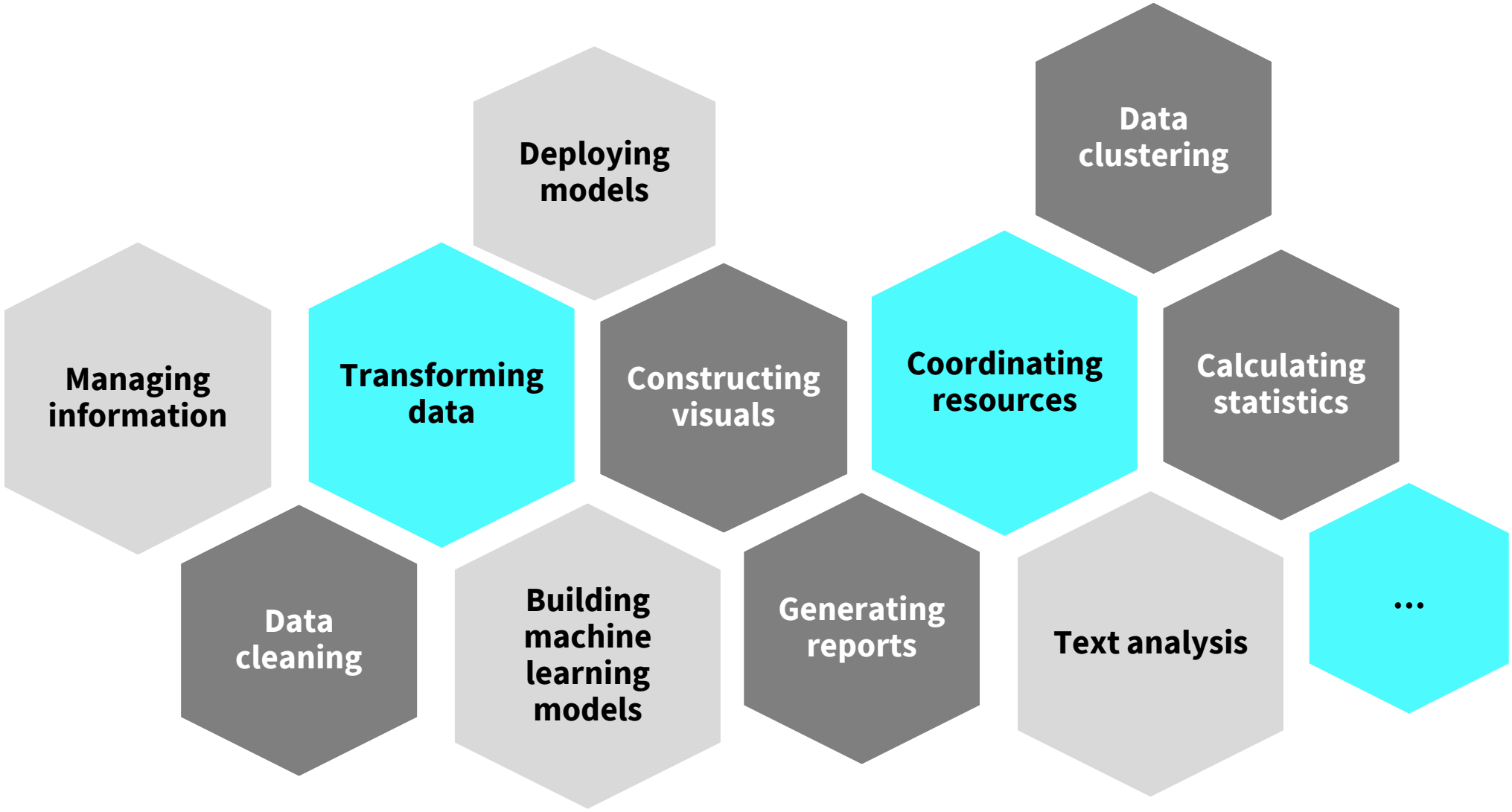
Property	Description
Modularity	– Stable interfaces that encapsulate volatile implementation details – **Points of planned variability** to extended behavior (hotspots) – Design and implementation **changes limited to these points** to **reduce the effort** required to understand/sustain the framework
Reusability	– **Stable interfaces** define **generic components that can be extended** – Reuse of framework components **improves developer efficiency**
Extensibility	– Providing **explicit** hook **methods for planned variability**, essential to ensure **rapid customization** of new application features
Inversion of control	– The **flow of control** not **dictated** by users, but **by the framework** itself – Enables canonical application processing steps to be customized by hotspots
Non-modifiable code	– The framework source code **supposed to be extended but not modified**

Source of the table: Course book DLMBDSA02, p. 15f.

FRAMEWORKS CATEGORIES

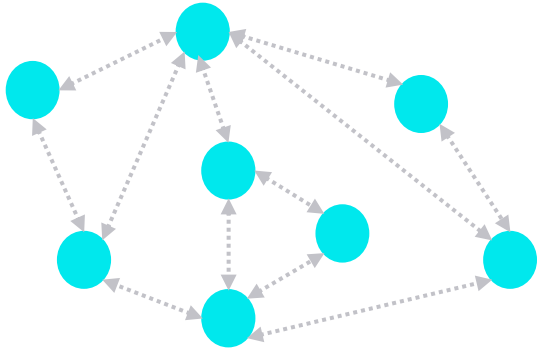
Category	Description
Technology frameworks	- provide a standard and generic software foundation - examples: COM, CORBA, and Java
Application frameworks	- implement the standard structure of an application - examples: ModelView Controller, Microsoft Foundation Classes, Mac-App/ACS
Business frameworks	- domain-specific, business solution that can be extended into an organization - examples: Enterprise Resource Planning (ERP), San Francisco Business Objects (Taligent/IBM), Oracle Enterprise Architecture Framework
Web-based frameworks	- designed to support development of dynamic websites, web applications, web services, and web resources - examples: Zope (Zope Corporation), Apache Struts, Django, Ruby on Rails, Symfon

FRAMEWORKS IN DATA ANALYTICS SUPPORT AT

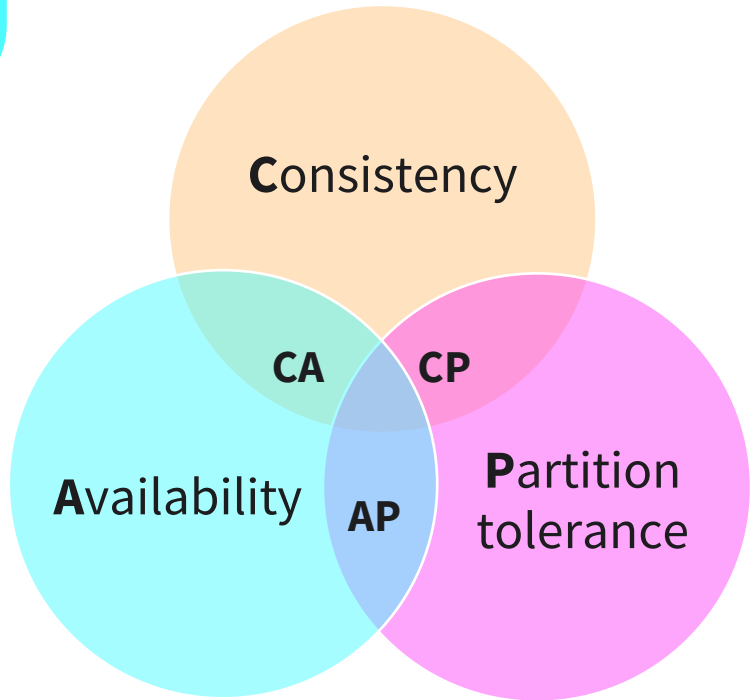


Source of the text: Course book DLMBDSA02, p. 14.

*"A distributed system is a **collection of autonomous computing elements** [or **'nodes'**] **that appears to its users as a single coherent system**."*

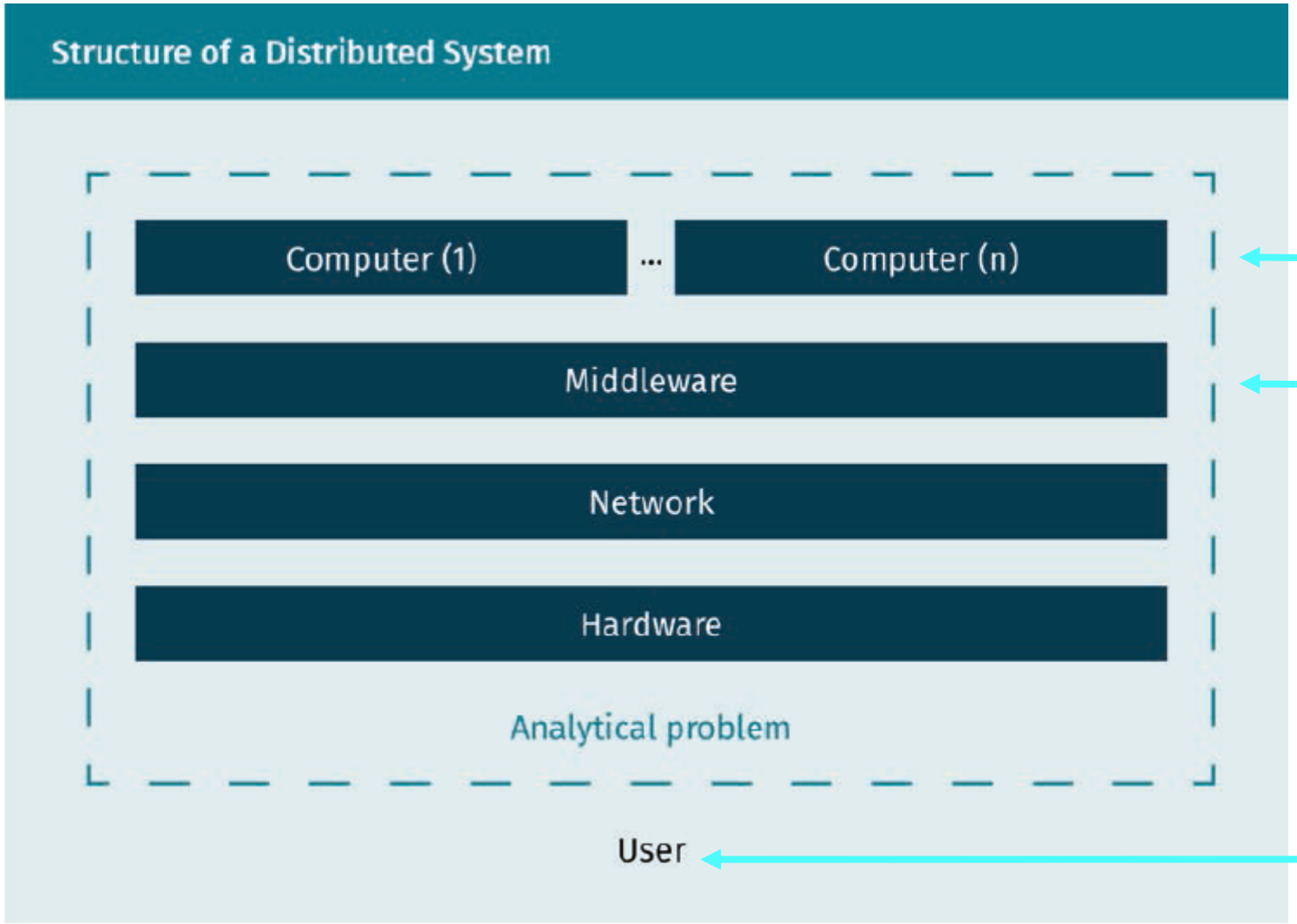


- Nodes are able to **behave independently** of each other.
- Nodes **collaborate** to serve a common goal.
- Nodes are inter**connected** to collaborate.



Source of the text: van Steen & Tanenbaum, 2017, p. 16.

DISTRIBUTED COMPUTING AND SYSTEMS

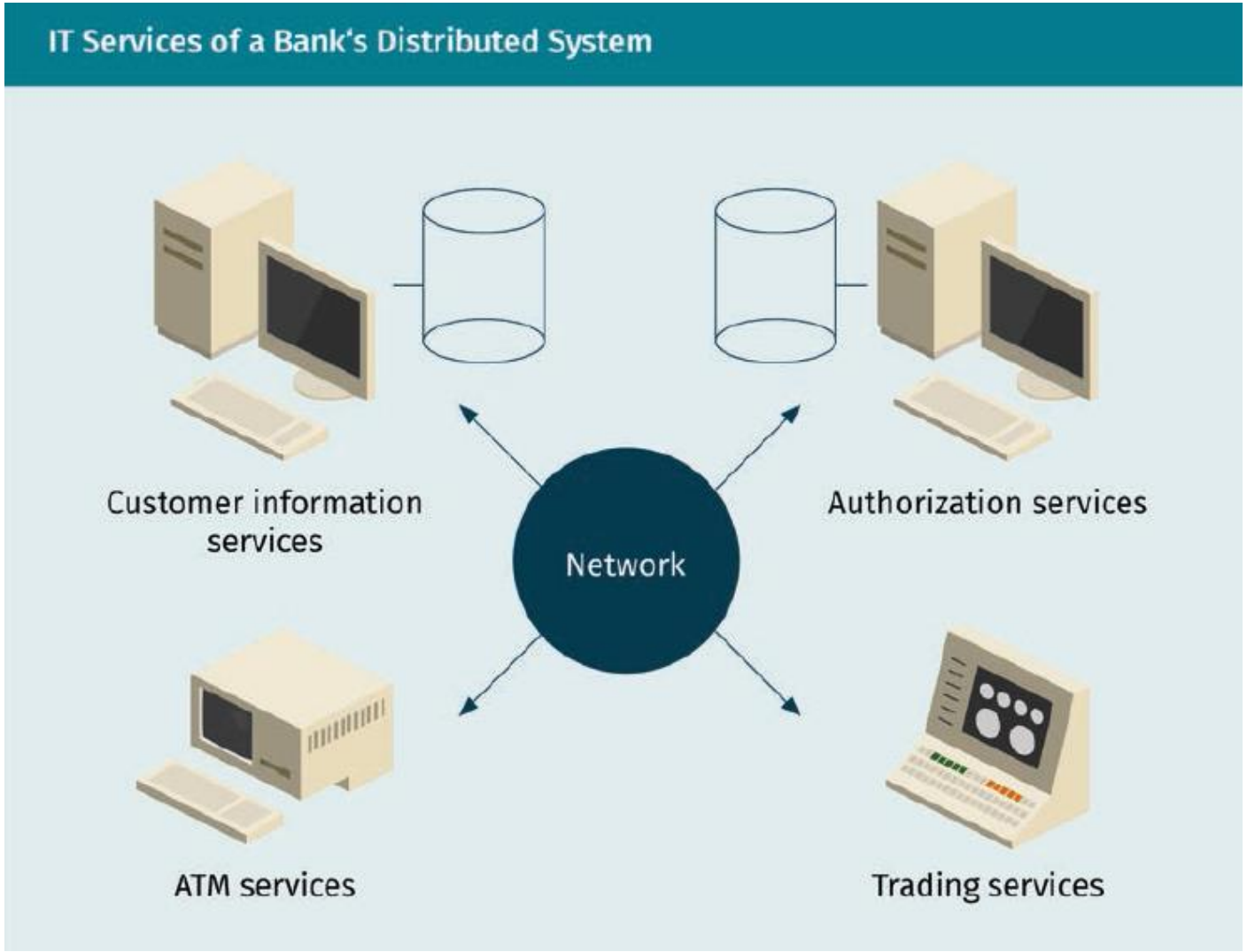


performs a specific procedure of the requested analytical task

facilitates the computers involved to synchronize their connected tasks

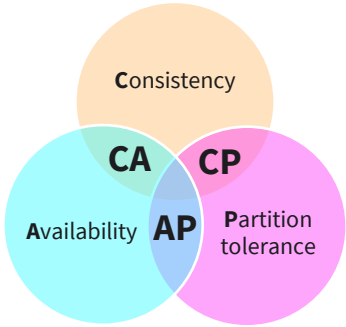
sees the whole distributed system as one computing unit

Source of the figure: course book



Source of the image: course book

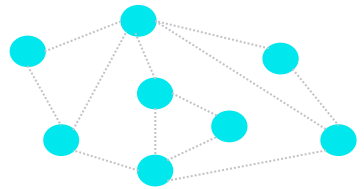
DISTRIBUTED SYSTEMS VS. CENTRALIZED SYSTEMS



	Centralized System	Distributed System
Autonomy	One **central system** with non-autonomous components	Multiple **autonomous computers (nodes)**
Homogeneity	Built using **homogenous** technology	(May be) built using **heterogenous** technology
Point(s) of control and failure	**Single point of control/failure**	**Multiple points of control/failure**, failing node(s) may not cause a failing system at whole
Availability	All resources are **consistently available or not at all (CA)**	Some resources **may not be available at all times (CP)**
Consistency		Some resources **may not be consistent at all times (AP)**

Source of the table: Course book DLMBDSA02, p. 20.

WITH DISTRIBUTED COMPUTING AND SYSTEMS COME ADVANTAGES AND CHALLENGES



Advantages

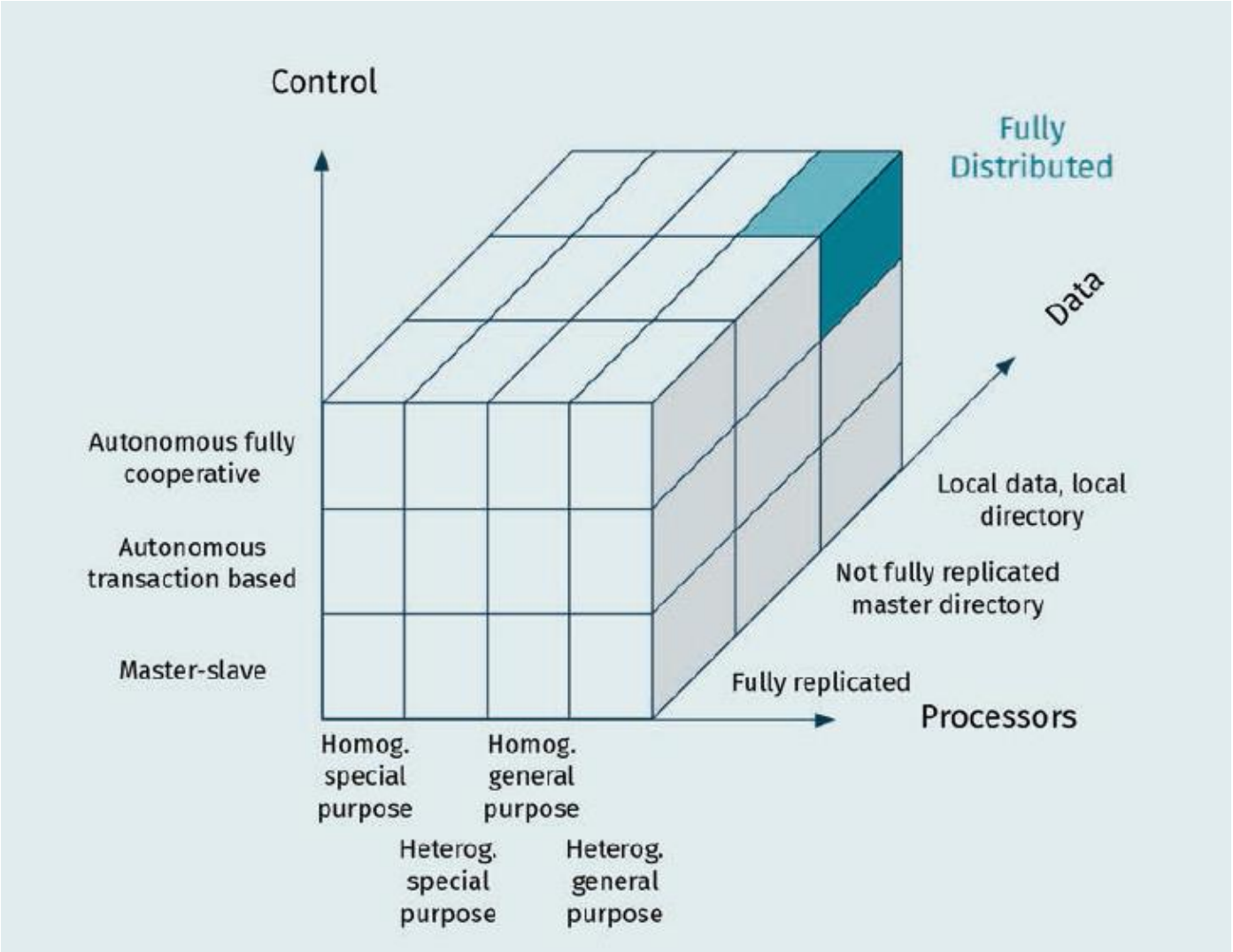
- Potential for **improved reliability, resilience, availability, performance** by avoiding single point of control and failure
- **Expandability** by adding new nodes to the system (horizontal scaling/scaling out)
- Potential for **cost reductions**

Challenges

- **Network reliance** and **latency**
- **Added complexity** with **overhead for coordination and communication** of nodes,
- **Added security threats**
- **Multiple points of failure** (some potentially out of your control)
- Harder to monitor, develop, test, maintain

Source of the text: Course book DLMBDSA02, p. 20.

THREE DIMENSIONS OF A DISTRIBUTED SYSTEMS



TRADITIONAL FILE APPROACH

Example of the Traditional File Approach

Index	1	2	3	4	5	6	7	8	9	10	...	...	...	...	n
Record#1	S	M	I	T	H				12	APR					
Record#2	M	A	T	T					03	JAN					
Record#m	A	N	D	E	R	S	O	N	17	OCT					

Family name | Birth date

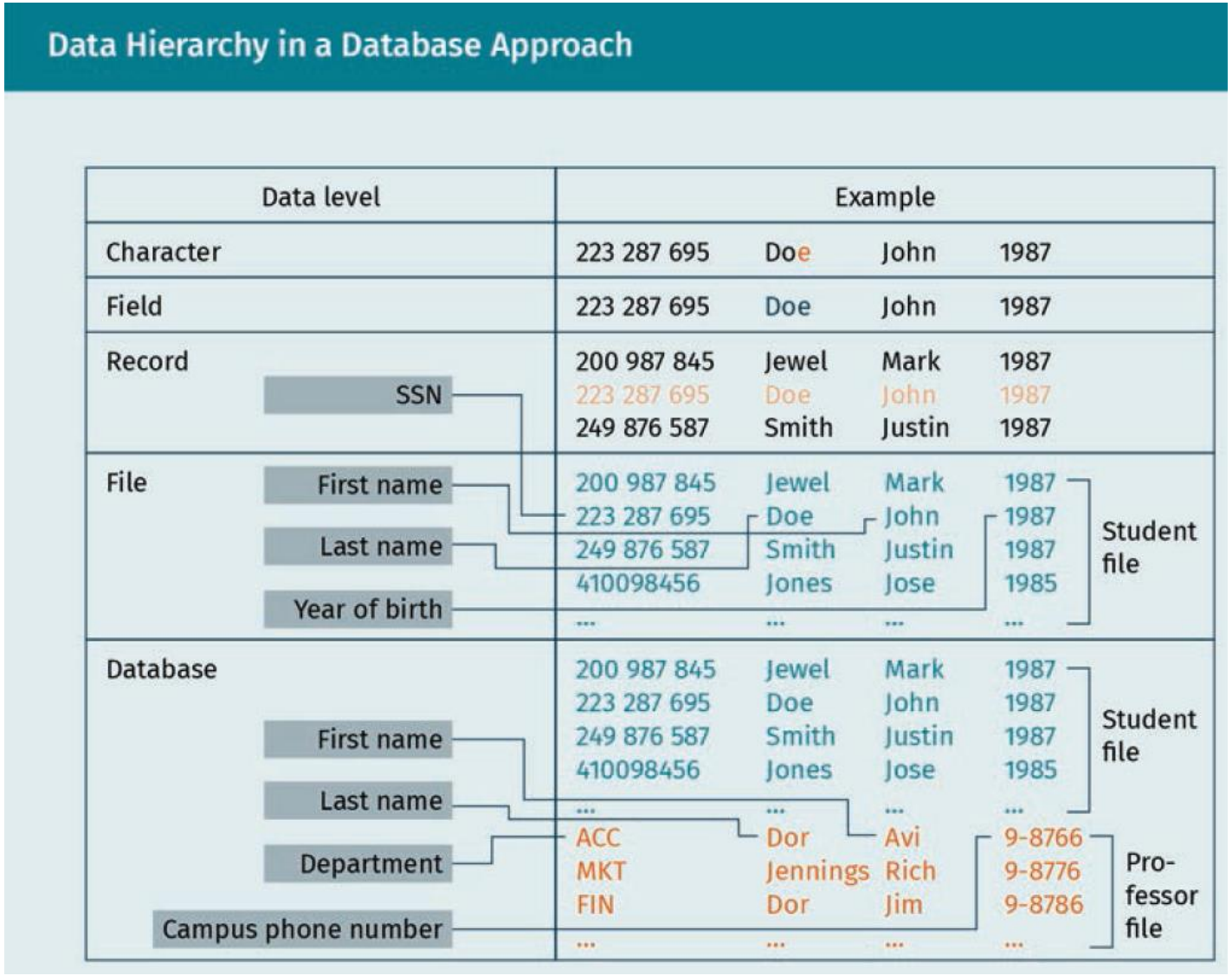
Source of the image: course book

*"[A **database** is] any collection of data, or information, that is specially organized for rapid search and retrieval by a computer. Databases are structured to facilitate the storage, retrieval, modification, and deletion of data […]"*

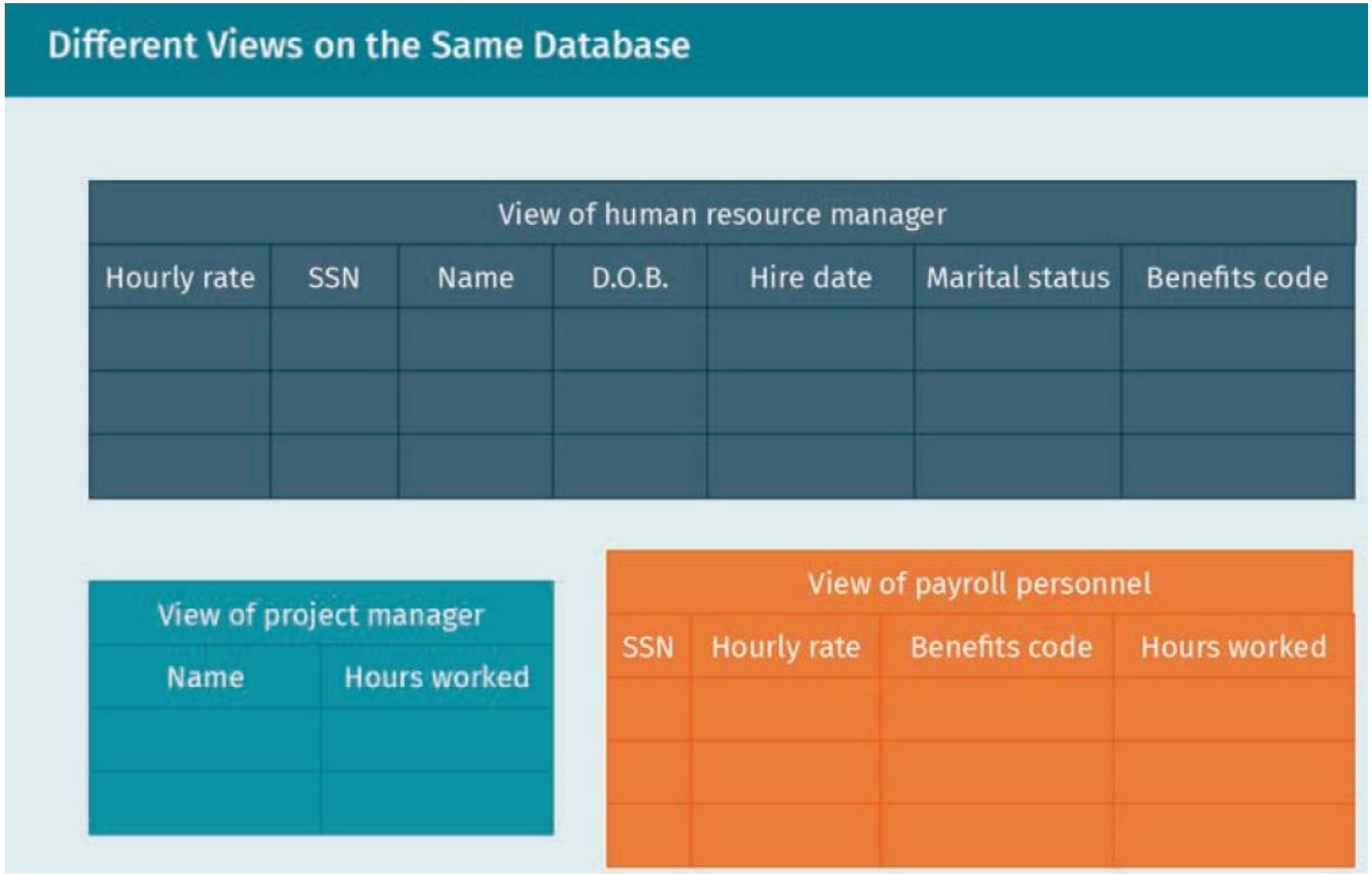
- **Database Management Systems (DBMS)** provide operations for reliable, secure, flexible, and comfortable access and management of the data stored in a database.

Source of the text: Britannica, 2021.

DATABASE HIERARCHY



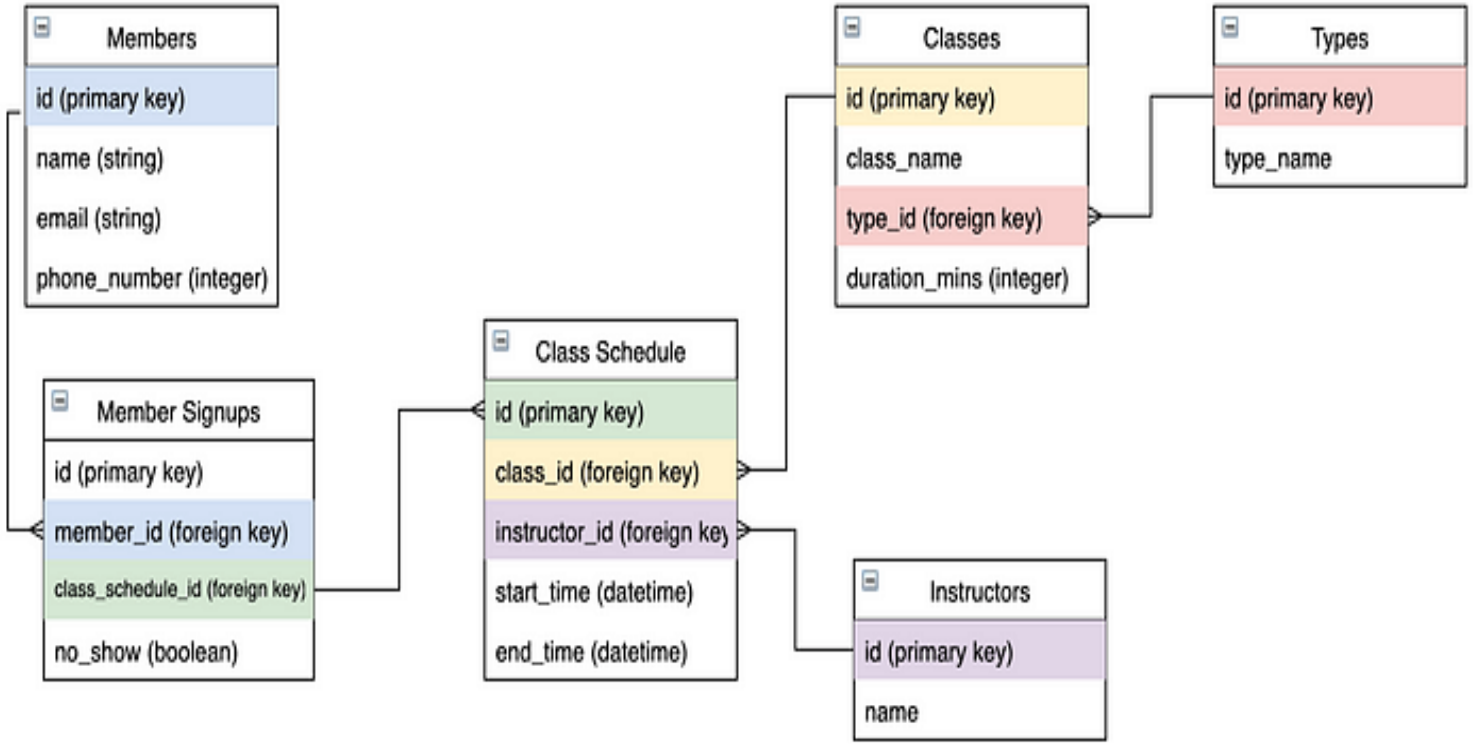
DIFFERENT VIEWS



Source of the image: course book

RATIONAL DATABASE MODELS

- Assigns the data into tables with rows to represent the records and columns to represent the fields.
- There are “primary keys” that are considered as unique identifiers for each record and can be used to link fields among several tables
- “Foreign keys” with relationships to the other table

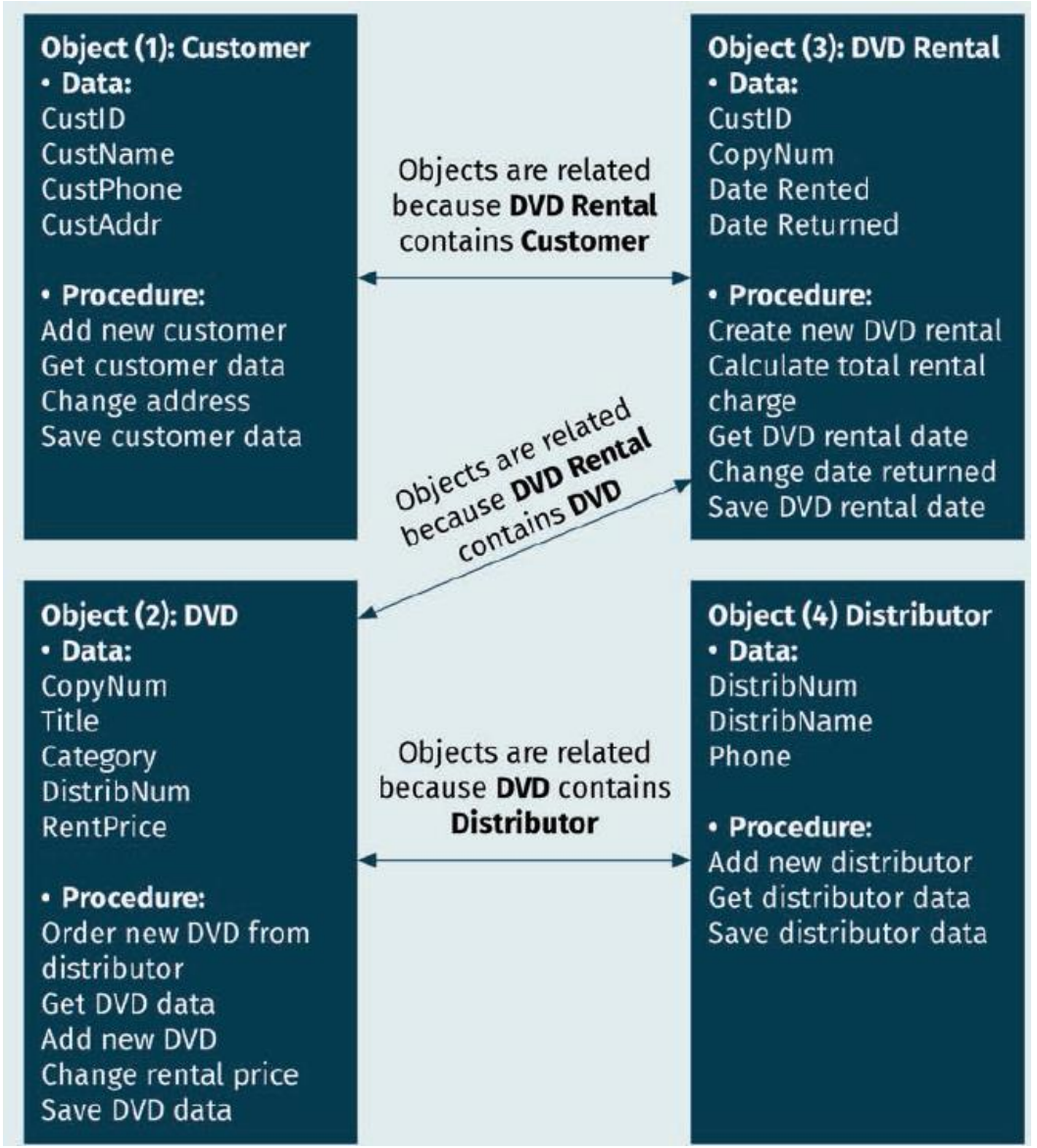


Workout Studio database

Source of the image: https://medium.com/@claire_logan/3-relational-data-model-examples-c9f70c61588c

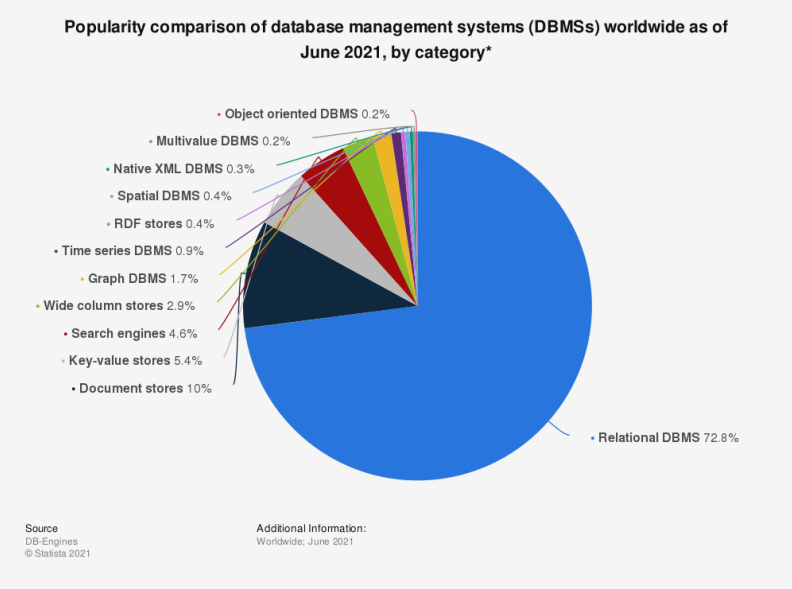
OBJECT-ORIENTED DATABASE (ODB) MODEL

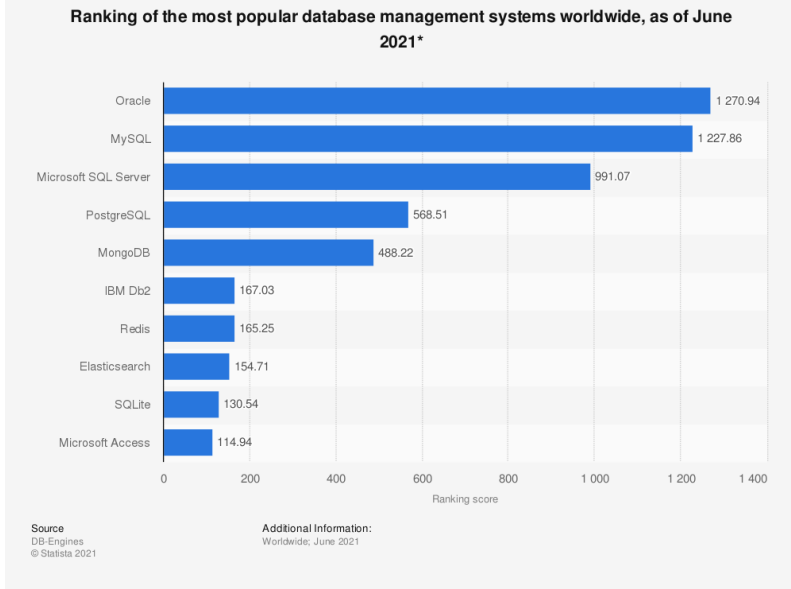
- Combines the storage of data and their associated procedures to access/retrieve them
- ODBs store data objects, not records.

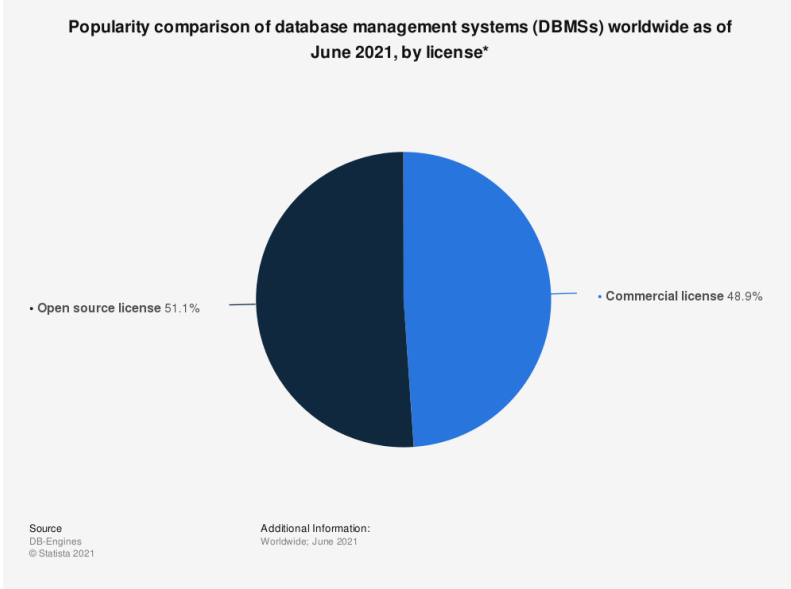


Souce of the image: course book

TODAY, WE MAY CHOOSE FROM A WIDE VARIETY OF HIGH-QUALITY DATABASE SYSTEMS

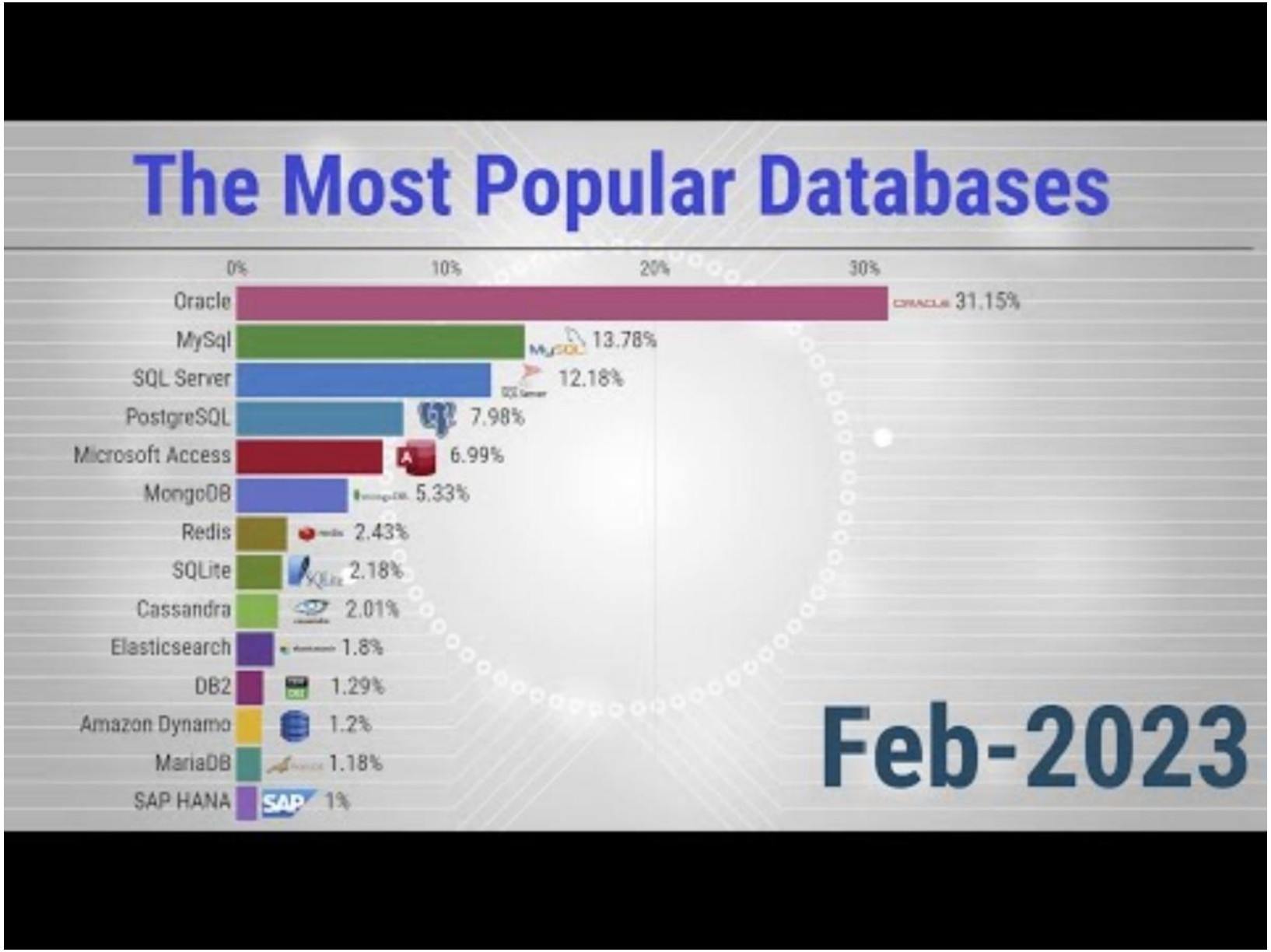






Source of the images: Statista, 2021a; Statista, 2021b; Statista, 2021c.

DATABASE SYSTEMS



Source of the video: https://www.youtube.com/watch?v=hfKnFvv9vRU

	STANDARD DATABASE MODEL	DATA WAREHOUSE
	Online Transaction Processing (OLTP)	**Online Analytical Processing (OLAP)**
Focus	**Operational** support of business processes	**Strategical (analytical)** support of business processes
Primary data access	Creates, reads, updates, or deletes data records (**CRUD**)	**Mostly reads** data records
Transaction size	Many **small transactions** with megabytes to gigabytes of data	**Long queries** with gigabytes to terabytes of data
Time focus	Focus on **current data**	Focus on **historical and aggregated data**
User#	Available for **thousands of users** (e.g., operating users)	Available for **hundreds of users** (e.g., decision makers and analysts)
Data dimensions	Data in **two-dimensional** tables of columns and rows (in RDBMS)	Data in **multi-dimensional** layers

Source of the table: Course book DLMBDSA02, p. 26.

DATA WAREHOUSE (DWH)

"A data warehouse is a […] ***data architecture*** *that* ***tracks integrated, consistent, and detailed data over time, establishing relationships*** *between them* ***using metadata and schema****."*

SUBJECT-ORIENTED
Reflecting business entities and processes of the organization

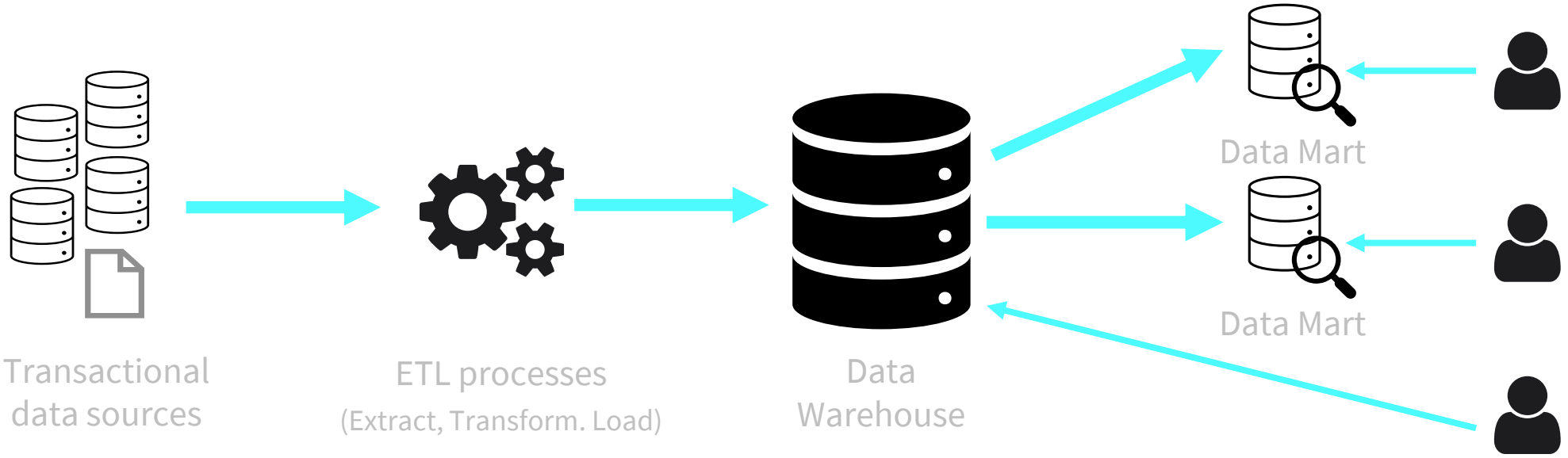
INTEGRATED AND CONSISTENT
Standardized formats and values, complete, accurate, and integer

TIME VARIANT AND NON-VOLATILE
Captures and tracks data changes over time

METADATA, SCHEMA, AND THE DATA DICTIONARY
Describing context of data and their structure

Source of the text: Teradata, 2021.

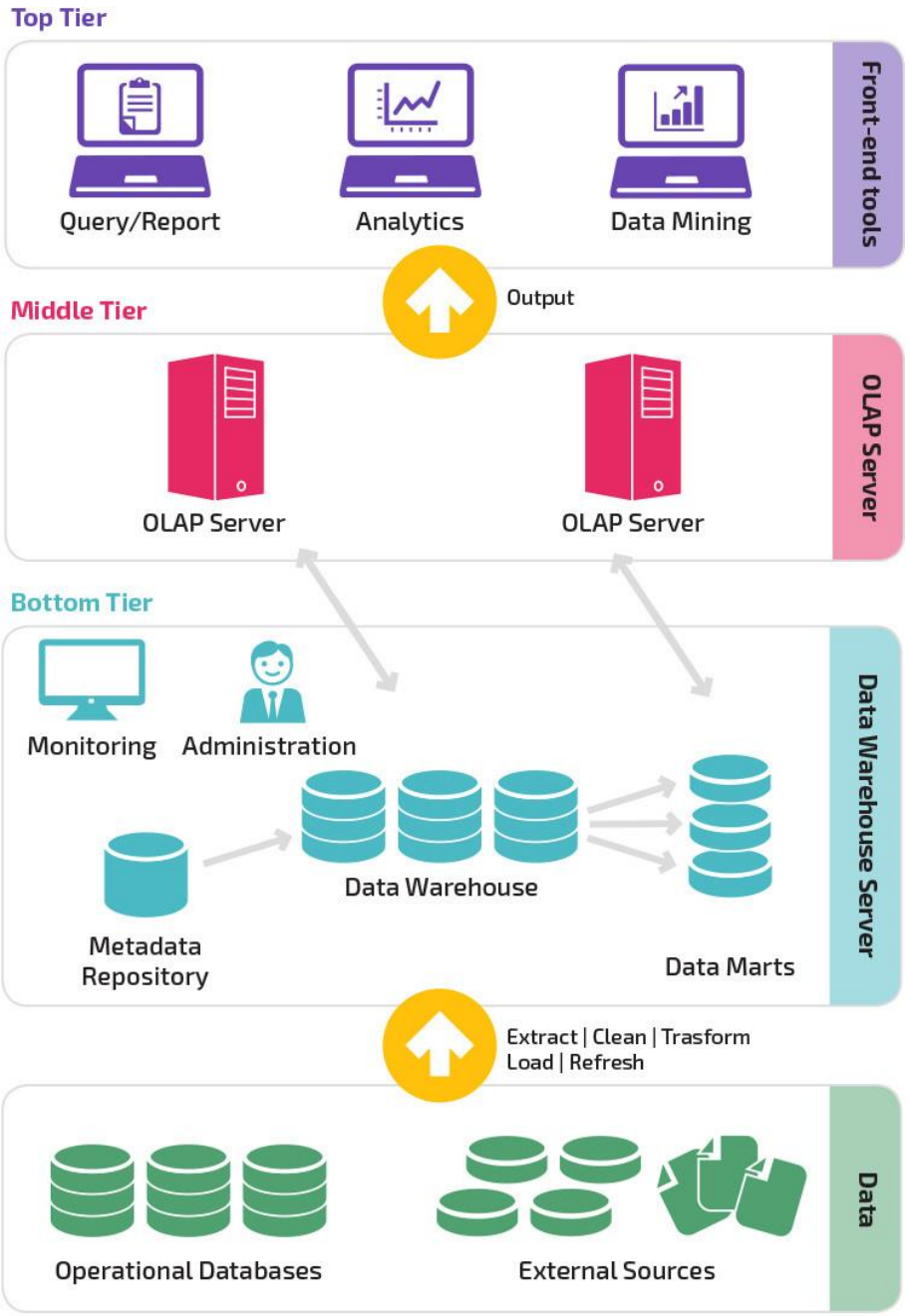
- **Extract**ed from transactional databases
- **Transform**ed and cleansed to follow unified formats using common set of enterprise definitions
- **Load**ed and transferred into the DWH



Source of the text: Course book DLMBDSA02, p. 26.

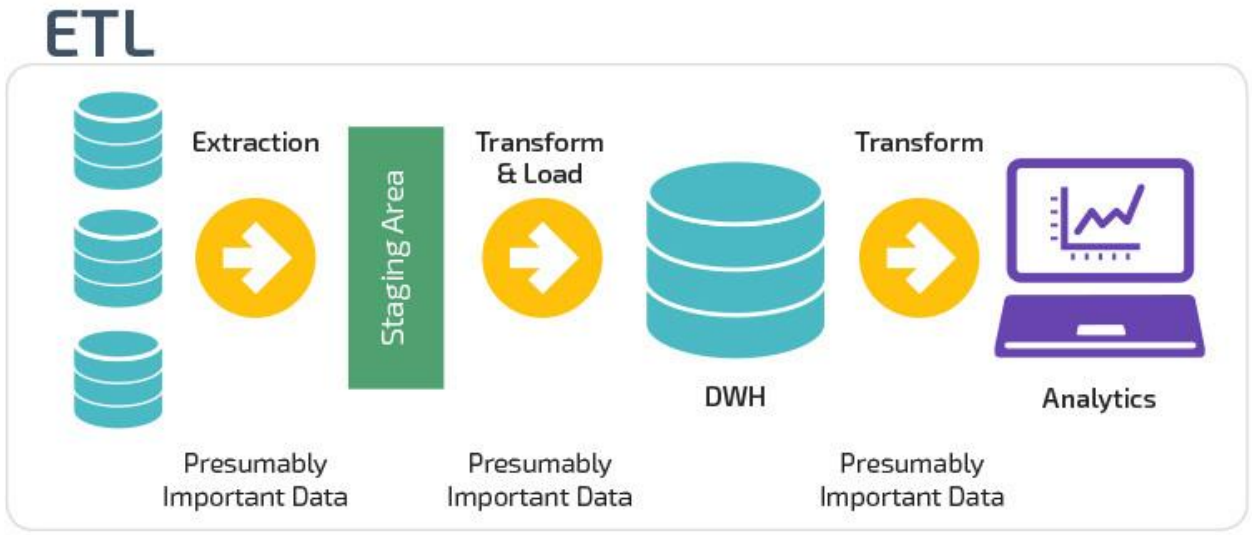
THREE-TIER ARCHITECTURE

- Bottom tier: data from many different sources
- Middle tier: OLAP server - transforms the data into a structure better suited for analysis and complex querying
- Top tier: client layer - high-level data analysis, querying reporting, and data mining

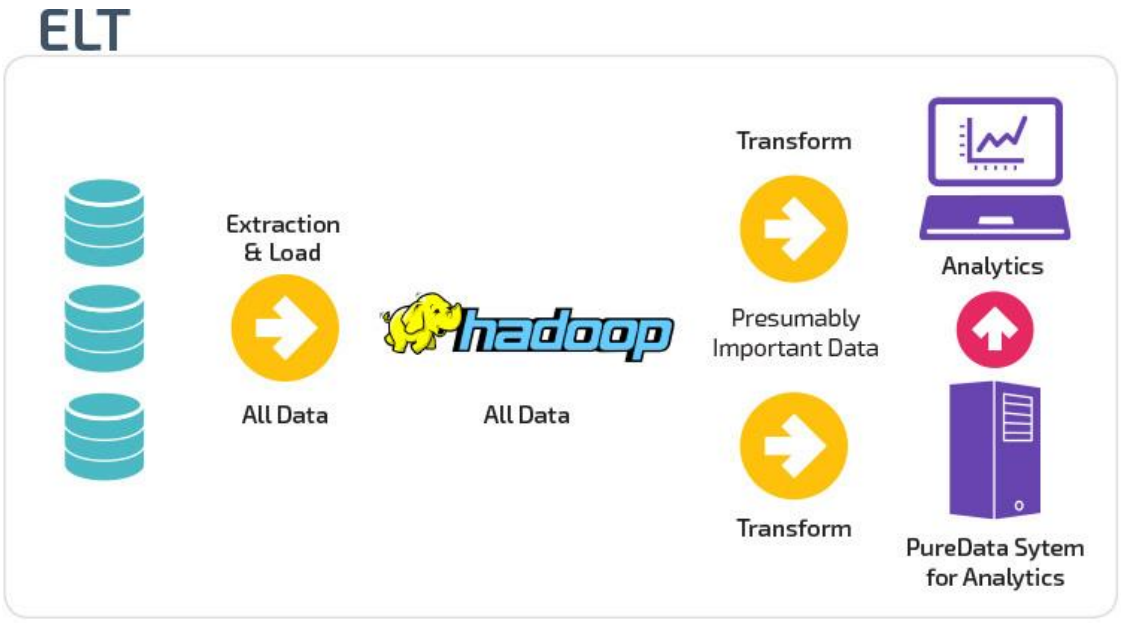


Source of the image: https://panoply.io/data-warehouse-guide/data-warehouse-architecture-traditional-vs-cloud/

ETL VS. ELT



Extract, Transform, Load (ETL)



Extract Load Transform (ELT)

Source of the images: https://panoply.io/data-warehouse-guide/data-warehouse-architecture-traditional-vs-cloud/

Source of the video: https://www.youtube.com/watch?v=u6GTMs7ZCFg

REVIEW STUDY GOALS

- Understand what is meant by a software system and a framework
- Understand how frameworks support implementing data analytics
- Have an overview of advantages and challenges of distributed computing systems
- Understand how data warehousing differs from transactional database architectures

SESSION 1

TRANSFER TASK

TRANSFER TASK

Gather in groups of 2—3 to work on the following question.

What could a transactional database (OLTP) for companies in the following sectors store to support their core business?

Banking, Airlines, Universities, Telecommunication, Finance, Sales & Production, Manufacturing, HR Management, Social Network

TRANSFER TASK

Gather in groups of 2—3 to work on the following question.

What could a Data Warehouse for companies in the following sectors store to support their core business?

Banking, Airlines, Universities, Telecommunication, Finance, Sales & Production, Manufacturing, HR Management, Social Network

TRANSFER TASK

Gather in groups of 2—3 to work on the following task.

Please discuss and decide if the following artefacts illustrate a (transactional) database or a Data Warehouse! What are the reasons supporting your decision?

TRANSFER TASK

Sales Reps

Sales Rep ID (PK)	Sales Rep Name	District

Customers

Customer ID (PK)	Customer Name	Sales Rep ID (FK)

Invoices

Invoice#	Customer ID	Date	...

TRANSFER TASK

Sales Reps

Sales Rep ID (PK)	Sales Rep Name	District

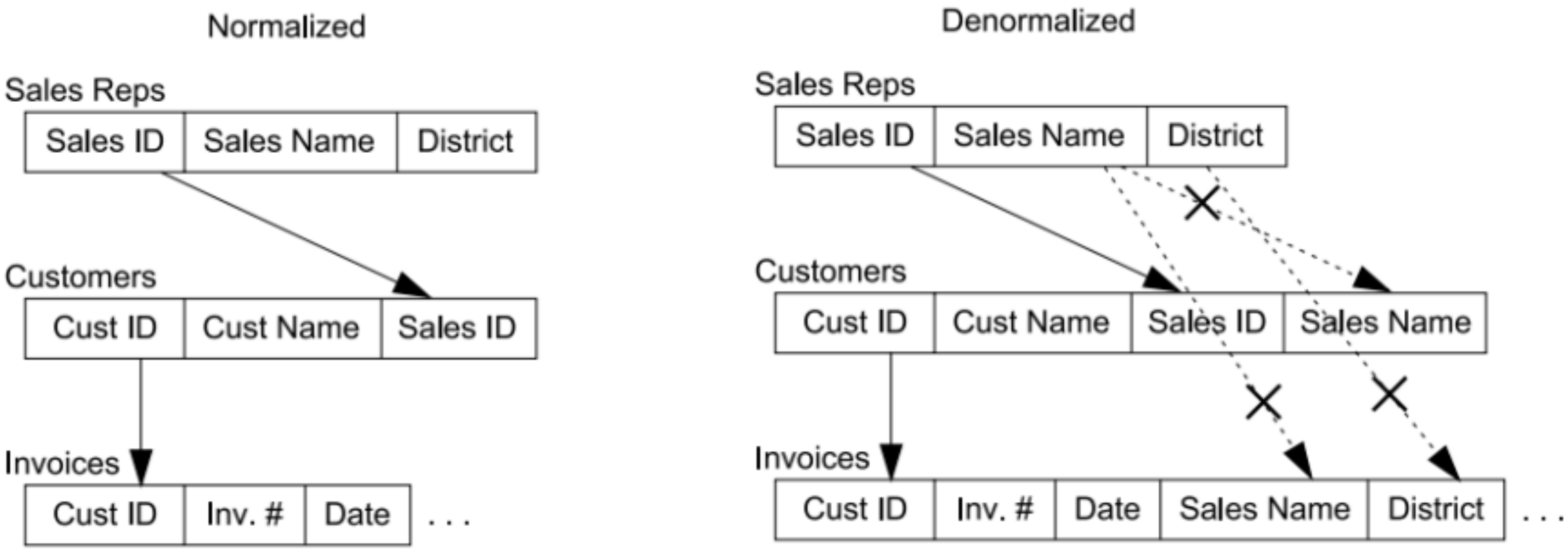
Customers

Customer ID (PK)	Customer Name	Sales Rep ID (FK)	Sales Rep Name

Invoices

Invoice#	Customer ID	Date	Sales Rep Name	District	...

TRANSFER TASK



TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your results.

The results will be discussed in plenary.

1. The software system used for analyzing complex data and quickly automating strategic business processes through web-based modeling and user interface is called ...

a) project management.
b) business process management.
c) customer relation management.
d) database management.

2. Which property of a software framework ensures the flow of control is dictated by the framework itself (not the users)?

a) Non-modifiable code
b) Modularity
c) Inversion of control
d) Extensibility

3. Which of the following is not an issue with a distributed computing system?

a) Performance improvement
b) Network reliability
c) Security
d) Multiple points of failure

LIST OF SOURCES

Britannica Encyclopaedia (2021). *database*. Encyclopedia Britannica. https://www.britannica.com/technology/database

Object Management Group (2018). *Kernel and Language for Software Engineering Methods.* http://semat.org/essence-1.2

Riehle, D. (2000). *Framework Design: A Role Modeling Approach.* Ph.D. Thesis, No. 13509. Zürich, Switzerland, ETH Zürich.

Statista (2021a). *DB-Engines. (June 14, 2021). Popularity comparison of database management systems (DBMSs) worldwide as of June 2021, by category*. https://www.statista.com/statistics/1131595/worldwide-popularity-database-management-systems-category/

Statista (2021b). *DB-Engines. (June 14, 2021). Ranking of the most popular database management systems worldwide, as of June 2021.* https://www.statista.com/statistics/809750/worldwide-popularity-ranking-database-management-systems/

Statista (2021c). *DB-Engines. (June 14, 2021). Popularity comparison of database management systems (DBMSs) worldwide as of June 2021, by license.* https://www.statista.com/statistics/1131575/worldwide-popularity-database-management-systems-license/

Teradata (2021). *What is Data Warehousing?* https://www.teradata.de/Glossary/What-is-Data-Warehousing

Van Steen, M., & Tanenbaum, A.S. (2017). *Distributed Systems* (3rd ed.). https://distributed-systems.net