DATABASES INTRODUCTION
DataBase Foundations

Author: Eng. Carlos Andrés Sierra, M.Sc.

cavirguezs@udistrital.edu.co

Lecturer
Computer Engineer
School of Engineering
Universidad Distrital Francisco José de Caldas

2024-III

Outline

Outline

Modular Software Components

- **Software Components** are the building *blocks* of software systems.

  → use more technologies

- Modular Software is a software design technique that emphasizes separating the functionality of a program into independent, interchangeable **modules**.

- **Software Applications** are the final product of software development.

- **Software Development** is the process of creating software applications.

Modular Software Components

- **Software Components** are the building blocks of software systems.
- **Modular Software** is a software design technique that emphasizes separating the functionality of a program into independent, interchangeable **modules**.
- Software Applications are the final product of software development.
- Software Development is the process of creating software applications.

Modular Software Components

- **Software Components** are the building blocks of software systems.
- **Modular Software** is a software `design` `technique` that emphasizes separating the functionality of a program into independent, interchangeable **modules**..
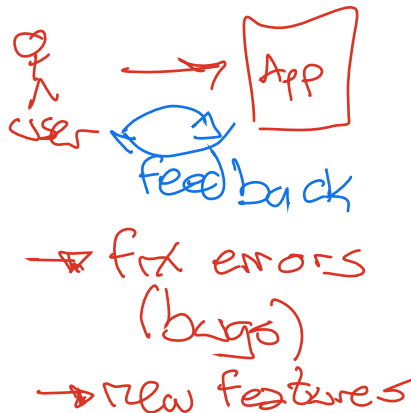- **Software Applications** are the final product of software development.
- Software Development is the process of creating software applications.

Modular Software Components

- **Software Components** are the building blocks of software systems.

- **Modular Software** is a software design technique that emphasizes separating the functionality of a program into independent, interchangeable **modules**.

- **Software Applications** are the final product of software development.

- **Software Development** is the process of creating software applications.

History of Data

Ubidots → IoT / ⇒ Petabytes/week   IoT

HISTORY OF DATA

| 19,000 BC | 1600s | 1800s | 1900s | 1990s |

The Ishango bone holds the first evidence of data collection and storage.

John Graunt introduces the concept of data analysis in 1663.

Herman Hollerith designs a machine that helped complete the US census in 1890.

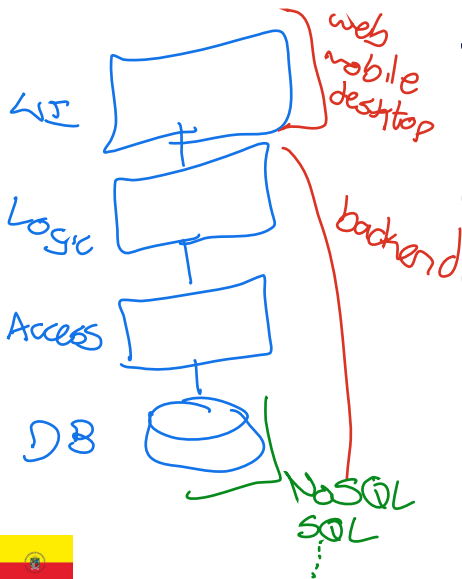Fritz Pfleumer invents the magnetic tape which later inspired the invention of floppy disks and hard disk drives.

Sir Tim Berners Lee invents the World Wide Web.

Applications

- Software based on **layers** of abstraction and modularity lets implement different database strategies.

- Database Systems are fundamental for **data management**.

- Data analysis, data mining, data visualization, and data interpretation are **applications** of **database systems**.

Applications

- Software based on **layers** of abstraction and modularity lets implement different database strategies.
- Database Systems are fundamental for **data management**.
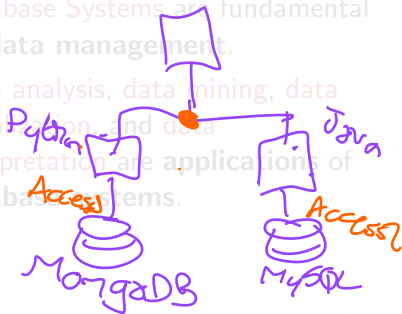- Data analysis, data mining, data visualization, data interpretation are **applications** of database systems.

Applications

- Software based on **layers** of abstraction and modularity lets implement different database strategies.
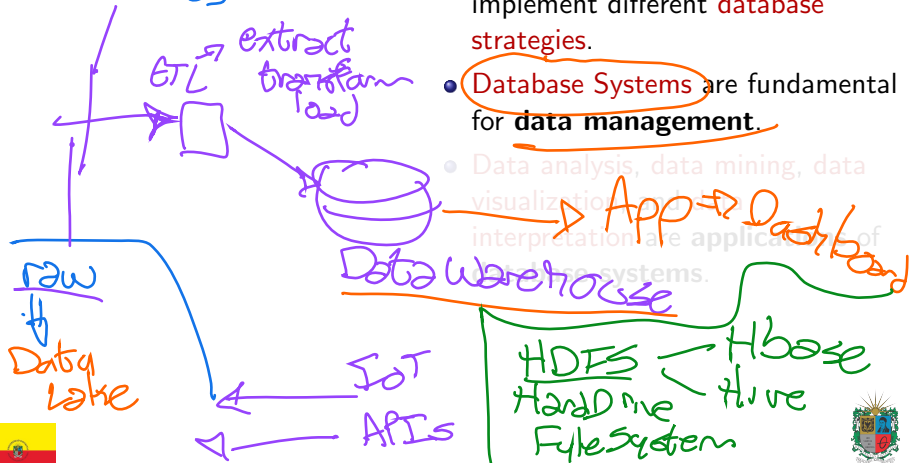
- Database Systems are fundamental for **data management**.

- Data analysis, data mining, data visualization, and data interpretation are **applications** of **database systems**.
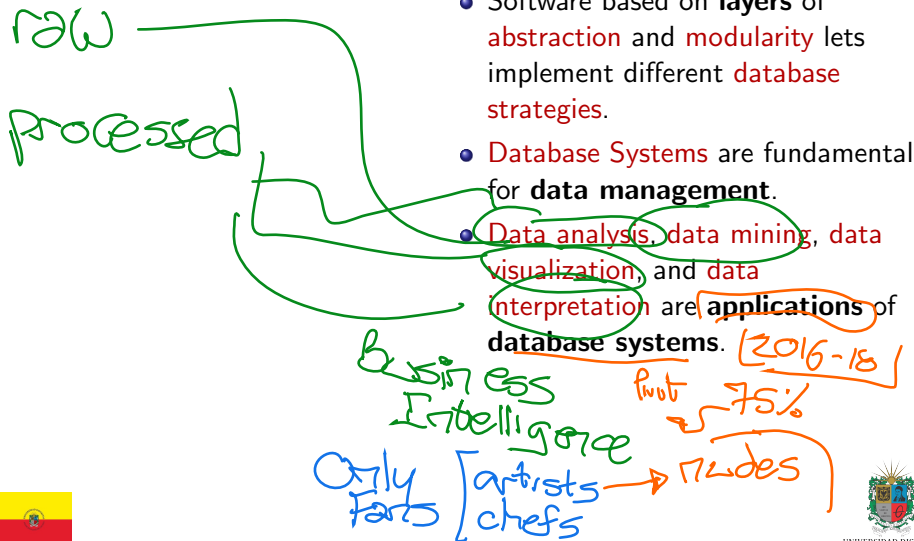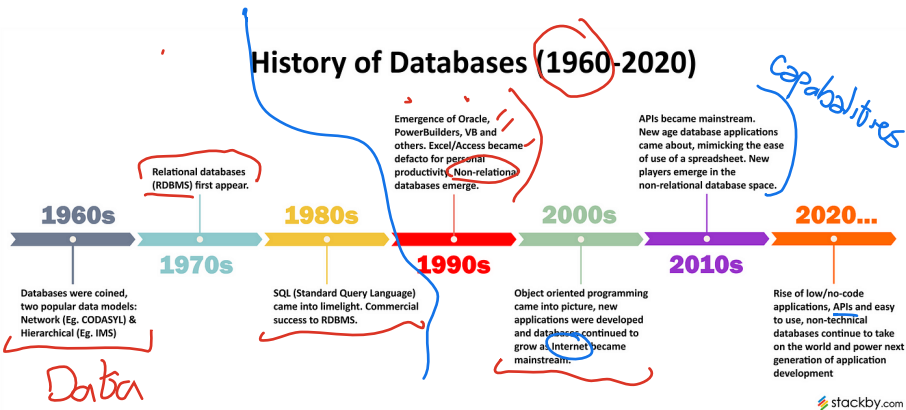
(handwritten annotations)

raw

processed

Business Intelligence

[2016 - 18]

Pivot ~ 75%

Only Facts [artists chefs] → nudes

History of DataBases

History of Databases (1960-2020)

1960s

Databases were coined, two popular data models: Network (Eg. CODASYL) & Hierarchical (Eg. IMS)

Data

1970s

Relational databases (RDBMS) first appear.

1980s

SQL (Standard Query Language) came into limelight. Commercial success to RDBMS.

Emergence of Oracle, PowerBuilders, VB and others. Excel/Access became defacto for personal productivity. Non-relational databases emerge.

1990s

Object oriented programming came into picture, new applications were developed and databases continued to grow as Internet became mainstream.

2000s

APIs became mainstream. New age database applications came about, mimicking the ease of use of a spreadsheet. New players emerge in the non-relational database space.

Capabilities

2010s

2020...

Rise of low/no-code applications, APIs and easy to use, non-technical databases continue to take on the world and power next generation of application development

stackby.com

Case of Study: DataBase System

Outline

From Data to Information

$\text{booooo} \approx 2^{20}$

32_0

- **Data**: is a set of **values** of **qualitative** or **quantitative** variables.

- Data Management: is the process of collecting, storing, processing, and analyzing data

- Data Analysis: is the process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, informing conclusions, and supporting decision-making

$\rightarrow$ Numbers

category

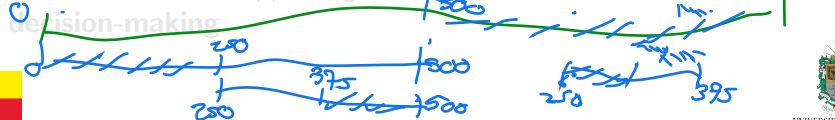attributes (text)

more information

Vector DataBase

faster

text $\Rightarrow$ Transformer $\Rightarrow$ Vector

$2^{10} = 1024$

Binary search

1000

From Data to Information

- Data: is a set of **values** of **qualitative** or **quantitative** variables.

- Data Management: is the process of **collecting**, **storing**, **processing**, and **analyzing** data.

- Data Analysis: is a process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, informing conclusions, and supporting decision-making.

(Handwritten annotations: DBA ⇒ Data Base Administrator; decision making)

From Data to Information

- **Data**: is a set of **values** of **qualitative** or **quantitative** variables.

- **Data Management**: is the process of **collecting**, **storing**, **processing**, and **analyzing** data.

- **Data Analysis**: is a process of **inspecting**, **cleansing**, **transforming**, and **modeling** data with the goal of **discovering** useful **information**, informing **conclusions**, and supporting **decision-making**.

Structured and Unstructured Data

Structured Data vs Unstructured Data

Structured Data *SQL*

- Can be displayed in rows, columns and relational databases
- Numbers, dates and strings
- Estimated 20% of enterprise data *(Gartner)*
- Requires less storage
- Easier to manage and protect with legacy solutions

Unstructured Data

- Cannot be displayed in rows, columns and relational databases
- Images, audio, video, word processing files, e-mails, spreadsheets
- Estimated 80% of enterprise data *(Gartner)*
- Requires more storage
- More difficult to manage and protect with legacy solutions

Tables, Columns and Rows

- **Table** is a collection of related data held in a structured format within a database.
- **Column** is a set of data values of a particular simple type, one for each row of the table.
- **Row** is a set of data values of a particular relationship, one for each column of the table.

Key-Value Data Structures

- **Key**-**Value Data Structures** are a type of data structure that can map keys to values.
- **Key** is a unique identifier for a record in a data fragment. **Value** is the data that is associated with the key.

Primary and Foreign Keys

- **Primary Key** is a unique identifier for a record in a data set.
- **Foreign Key** is a column or group of columns in a table that links to a primary key in another table.

CRUD Operations

- CRUD is an acronym for **Create**, **Read**, **Update**, and **Delete**.
- Create is the process of **adding** new **records** to a **data set**.
- Read is the process of **retrieving records** from a **data set**.
- Update is the process of **modifying records** in a **data set**.
- Delete is the process of **removing records** from a **data set**.

Outline

1 Software Components and Applications

2 Glosary

3 DataBase Classification

4 MER Diagrams
   - Study Case: Spotify

DataBase Classification

- **DataBase** is a collection of **data** that is **organized** so that it can be **easily accessed**, **managed**, and **updated**.

- Relational DataBase is a type of **database** that stores and provides access to **data points** that are **related** to one another.

- NoSQL DataBase is a type of **database** that provides a mechanism for **storage** and **retrieval** of **data** that is **modeled** in **means** other than the **tabular relations** used in **relational databases**.

DataBase Classification

- **DataBase** is a collection of **data** that is **organized** so that it can be **easily accessed**, **managed**, and **updated**.
- **Relational DataBase** is a type of **database** that stores and provides access to **data points** that are **related** to one another.
- **NoSQL DataBase** is a type of **database** that provides a mechanism for **storage** and **retrieval** of **data** that is **modeled** in **means** other than the **tabular relations** used in **relational databases**.
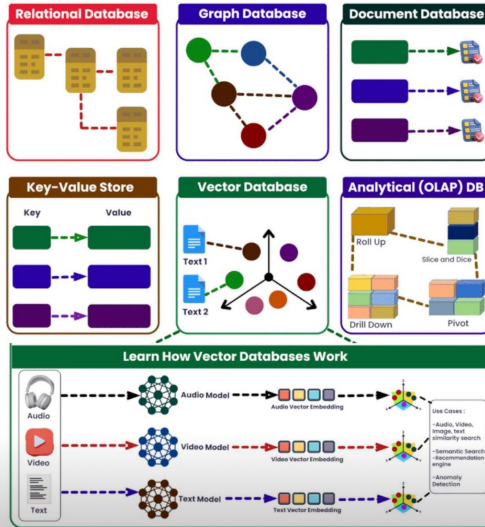
DataBase Classification

- **DataBase** is a collection of **data** that is **organized** so that it can be **easily accessed**, **managed**, and **updated**.
- **Relational DataBase** is a type of **database** that stores and provides access to **data points** that are **related** to one another.
- **NoSQL DataBase** is a type of **database** that provides a mechanism for **storage** and **retrieval** of **data** that is **modeled** in **means** other than the **tabular relations** used in **relational databases**.

Types of Database

Outline

1. Software Components and Applications

2. Glosary

3. DataBase Classification

4. MER Diagrams
   - Study Case: Spotify

Entity-Relationship Model

Entity-Relationship Model is a **data model** for **describing** the **data** or **information** aspects of a **business domain** or its **processes**.

Entity Definition

Entity is a **thing** or **object** in the **real world** that is **distinguishable** from other

Relationship between Entities

Relationship is a **connection** between **entities**. This connection could be **one-to-one**, **one-to-many**, and **many-to-many**.

Creating our own **Espotifai**

Outline

Thanks!

Questions?

Repo: *https://github.com/EngAndres/ud-public/tree/main/courses/databases-foundations*