Introduction & Entity Relationship (ER) Model

MODULE I

Jacob P Cherian
Asst.Professor
Dept.of CSE, Saintgits College of Engineering

Contents

Characteristics of Database system

Database Users

Structured, Semi-structured and Unstructured data

Data Models & Schema

Three Schema Architecture

Database Languages

Database Architectures and Classification.

ER Model -Basic concepts, Entity Set & Attributes

Relationships and constraints

Cardinality, Participation, Weak entities,

Relationships of Degree 3.

1.Data & Database

Data are known facts that can be recorded and that have implicit meaning.

For example, consider the names, telephone numbers, and addresses of the people.

A database is a collection of related data.

A database is a logically coherent collection of data with some inherent meaning

1.1 Database-Perspective

Designed, built, and populated with data for a specific purpose.

Intended group of users.

Source of Data

Interaction with events in the real world

Audience that is actively interested in its contents

1.2 Database Management System

"A database management system (DBMS) is a computerized system that enables users to create and maintain a database. The DBMS is a general-purpose software system that facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications".

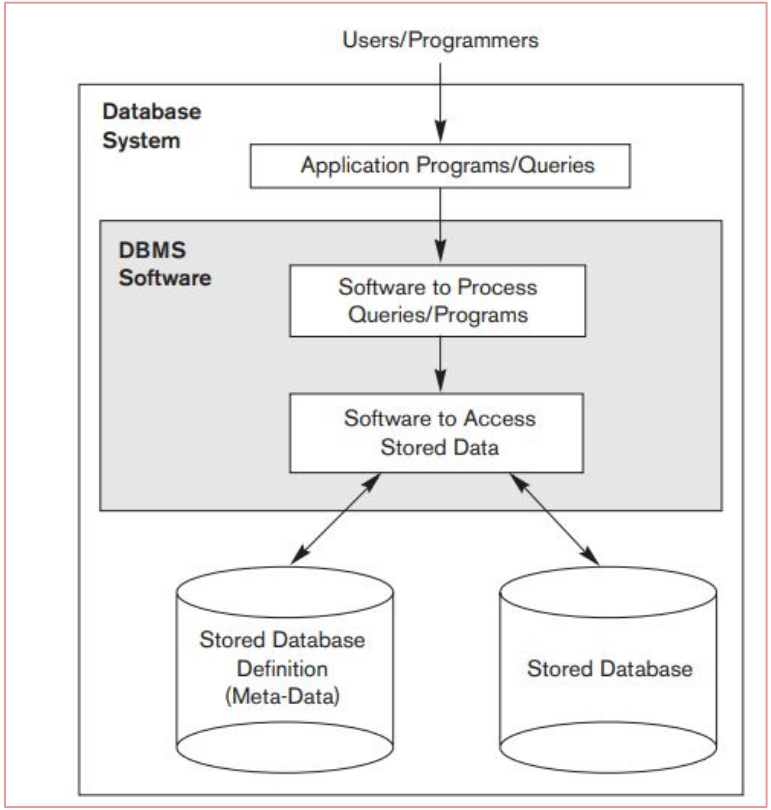
1.3 Simple Database System Environment

Application program accesses the database by sending queries or requests for data to the DBMS

Key Features

System protection against hardware or software malfunction and security protection against unauthorized or malicious access.

Must be able to maintain the database system by allowing the system to evolve as requirements change over time.



1.4 Example-University DB

STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

1.5 Examples for DB Manipulation

Retrieve the transcript—a list of all courses and grades of 'Smith'

List the names of students who took the section of the 'Database' course offered in fall 2008 and their grades in that section

List the prerequisites of the 'Database' course

1.5 Examples for DB Updation

Change the class of ‘Smith’ to sophomore

Create a new section for the ‘Database’ course for this semester

Enter a grade of ‘A’ for ‘Smith’ in the ‘Database’ section of last semester

Characteristics of Database Approach

Reading Material - *Click Here*

2. Characteristics of DB System

Self-describing nature of a database system

Insulation between programs and data, and data abstraction

Support of multiple views of the data

Sharing of data and multi-user transaction processing

Self-Describing Nature of a Database System

A database contains a complete definition or description of the database structure and constraints.

The DBMS Catalog contains information such as the *structure of each file*, *the type and storage format of each data item*, and *various constraints on the data.*

Information stored in the catalog is called metadata.

Self-Describing Nature of a Database System

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
....	....	.....
....	....	.....
....	....	.....
Prerequisite_number	XXXXNNNN	PREREQUISITE

Metadata describes the structure of the primary database

Data Abstraction

A DBMS provides users with a conceptual representation of data that does not include many of the details of how the data is stored or how the operations are implemented.

A data model is a type of data abstraction that is used to provide this conceptual representation.

The data model hides storage and implementation details that are not of interest to most database users.

Data Abstraction

Data Item Name	Starting Position in Record	Length in Characters (bytes)
Name	1	30
Student_number	31	4
Class	35	1
Major	36	4

Database user is not concerned with the location of each data item within a record or its length; rather, the user is concerned that when a reference is made to Name of STUDENT, the correct value is returned

Multiple Views

Each user may require a different perspective or view of the database.

A view may be a subset of the database or it may contain virtual data that is derived from the database files but is not explicitly stored.

Multiple Views

TRANSCRIPT

Student_name	Student_transcript				
	Course_number	Grade	Semester	Year	Section_id
Smith	CS1310	C	Fall	08	119
	MATH2410	B	Fall	08	112
Brown	MATH2410	A	Fall	07	85
	CS1310	A	Fall	07	92
	CS3320	B	Spring	08	102
	CS3380	A	Fall	08	135

COURSE_PREREQUISITES

Course_name	Course_number	Prerequisites
Database	CS3380	CS3320
		MATH2410
Data Structures	CS3320	CS1310

Sharing of Data & Multi-user Transaction Processing

Essential if data for multiple applications is to be integrated and maintained in a single database.

The DBMS must include concurrency control software to ensure that several users trying to update the same data do so in a controlled manner so that the result of the updates is correct.

Sharing of Data & Multi-user Transaction Processing

Online transaction processing (**OLTP**) Applications - Air Ticket Reservation System , Movie Ticket Booking , ATM Transactions etc.

A fundamental role of multi-user DBMS software is to ensure that concurrent transactions operate correctly and efficiently.

3. Actors on the Scene!

Database Administrator

Database Designers

3. Actors on the Scene!

Database Administrator

Database Designers

Responsible for authorizing access to the database.

Coordinates & monitors its use.

Acquire software & Hardware resources when required.

Accountable for Security Breach, poor response time etc.

3. Actors on the Scene!

Database Administrator

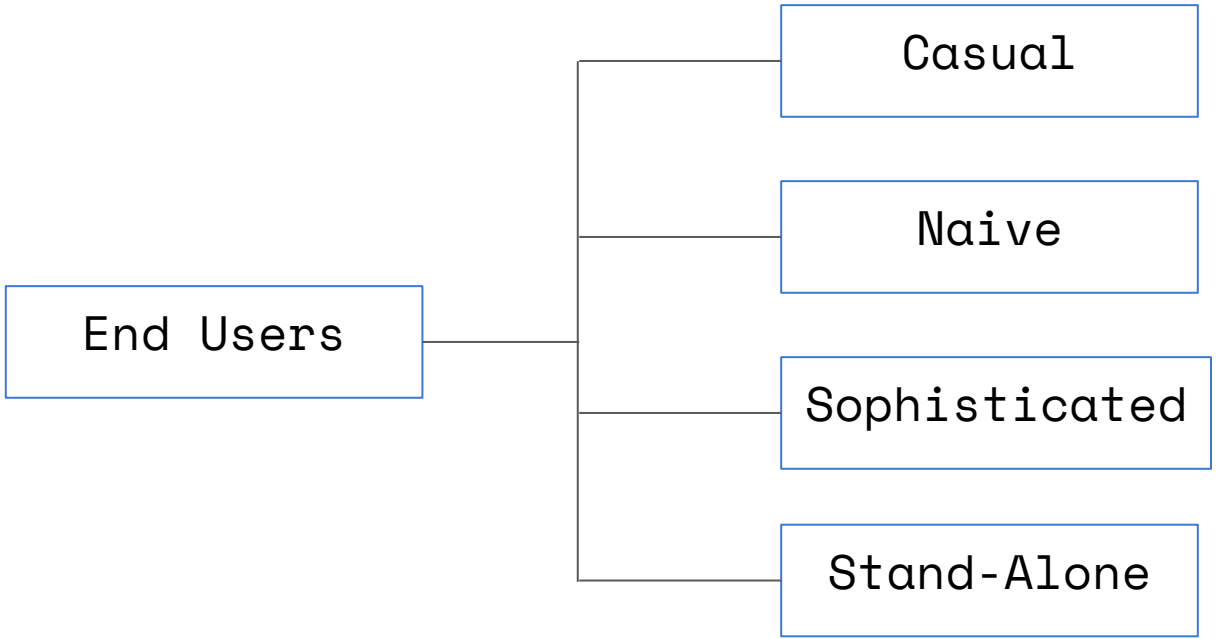
Database Designers

Identifies the data to be stored in the database.

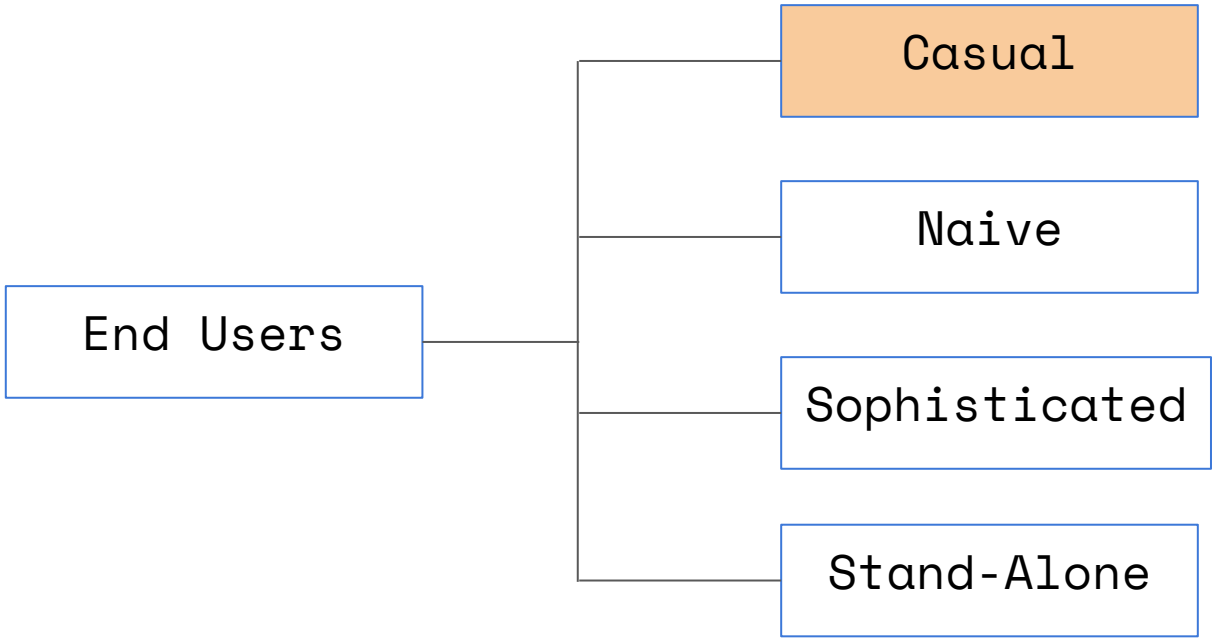
Choose appropriate structure to store this data.

Communicates with end users to create the proper database design.

3. Actors on the Scene!



3. Actors on the Scene!

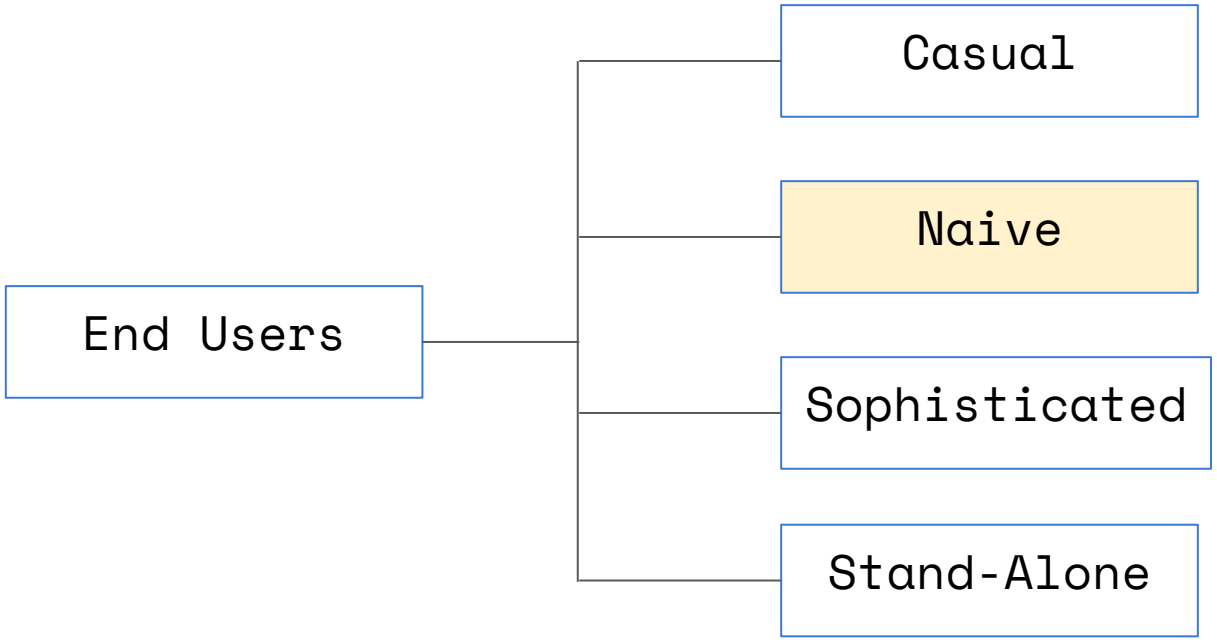


Occasionally Uses the DB.

They may need different information each time.

Use a sophisticated database query interface to specify their requests

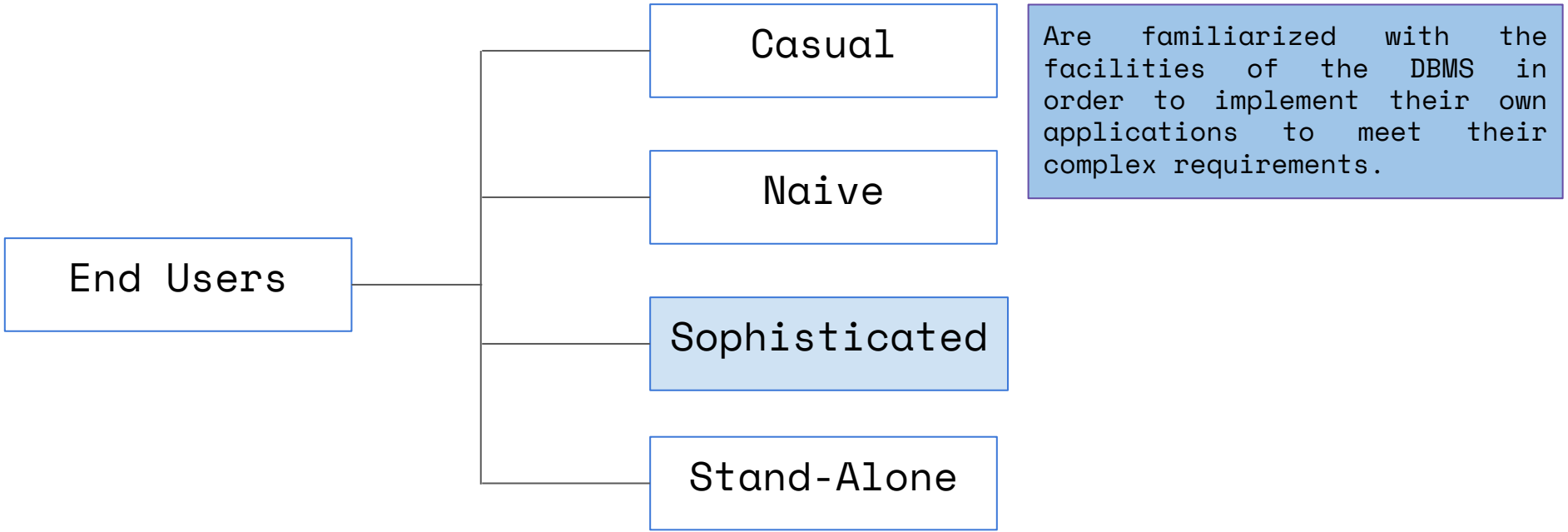
3. Actors on the Scene!



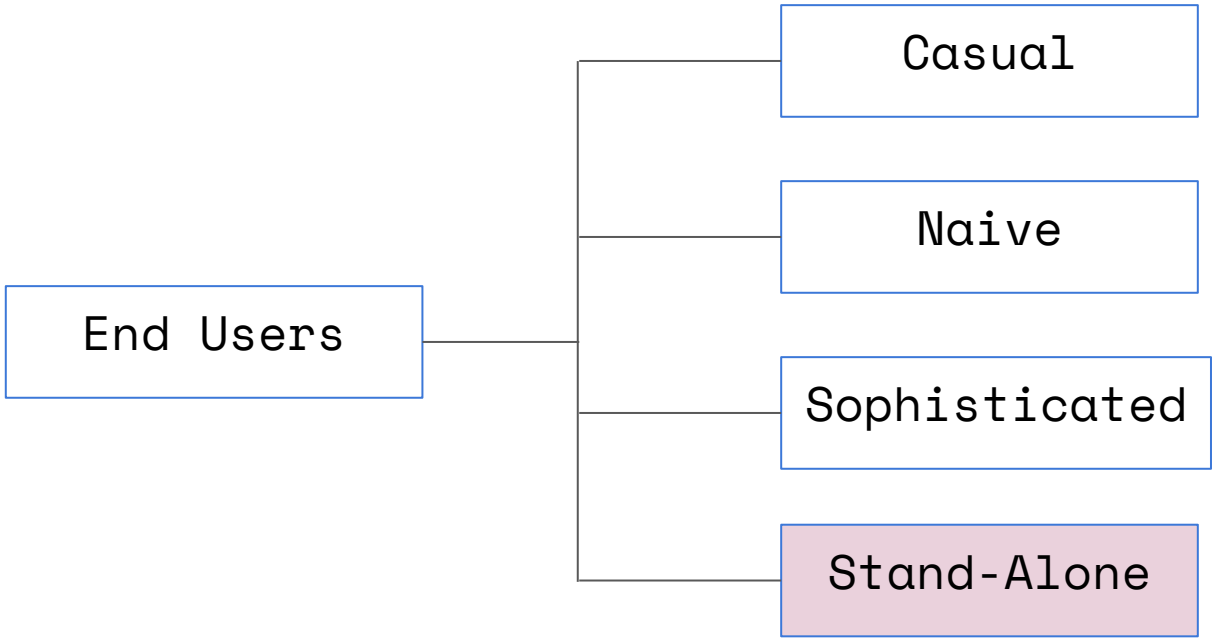
They constantly query and update the database, using standard types of queries and updates called *canned transactions*

Clerical Staff in a Bank, Reservation Agents, Social Media Users etc.

3. Actors on the Scene!



3. Actors on the Scene!



Maintain personal databases by using ready-made program packages that shall provide easy-to-use menu-based or graphics-based interfaces.

3. Workers Behind the Scene!

DBMS System Designers & Implementers

Tool Developers

Operators & Maintenance Personnel

3. Workers Behind the Scene!

DBMS System Designers & Implementers

Design and implement the DBMS modules and interfaces as a software package.

Handles the Different Modules - Design , Implementation

Ensures interfacing with OS and various programming languages.

3. Workers Behind the Scene!

Tool Developers

Tools - optional packages that are often purchased separately. Includes packages for database design, performance monitoring, natural language or graphical interfaces, prototyping, simulation, and test data generation.

Tool Developers design & implement the tools.

3. Workers Behind the Scene!

Operators & Maintenance Personnel

Responsible for the actual running and maintenance of the hardware and software environment for the database system.

Advantages of DBMS Approach

Controlling Redundancy

Providing Storage Structures and Search Techniques for Efficient Query Processing

Restricting Unauthorized Access

Providing Backup and Recovery

Providing Persistent Storage for Program Objects

Enforcing Integrity Constraints

Advantages of DBMS Approach

Permitting Inferencing and Actions Using Rules and Triggers

Additional Implications of Using the Database Approach This section discusses a few additional impl

Providing Multiple User Interfaces

Representing Complex Relationships among Data

Structured,Semi-structured & Unstructured data

Reading Material - *Click Here*

Structured Data

Structured data is data with a high degree of organization, typically stored in a spreadsheet-like manner.

It concerns all data which can be stored in database SQL in a table with rows and columns.

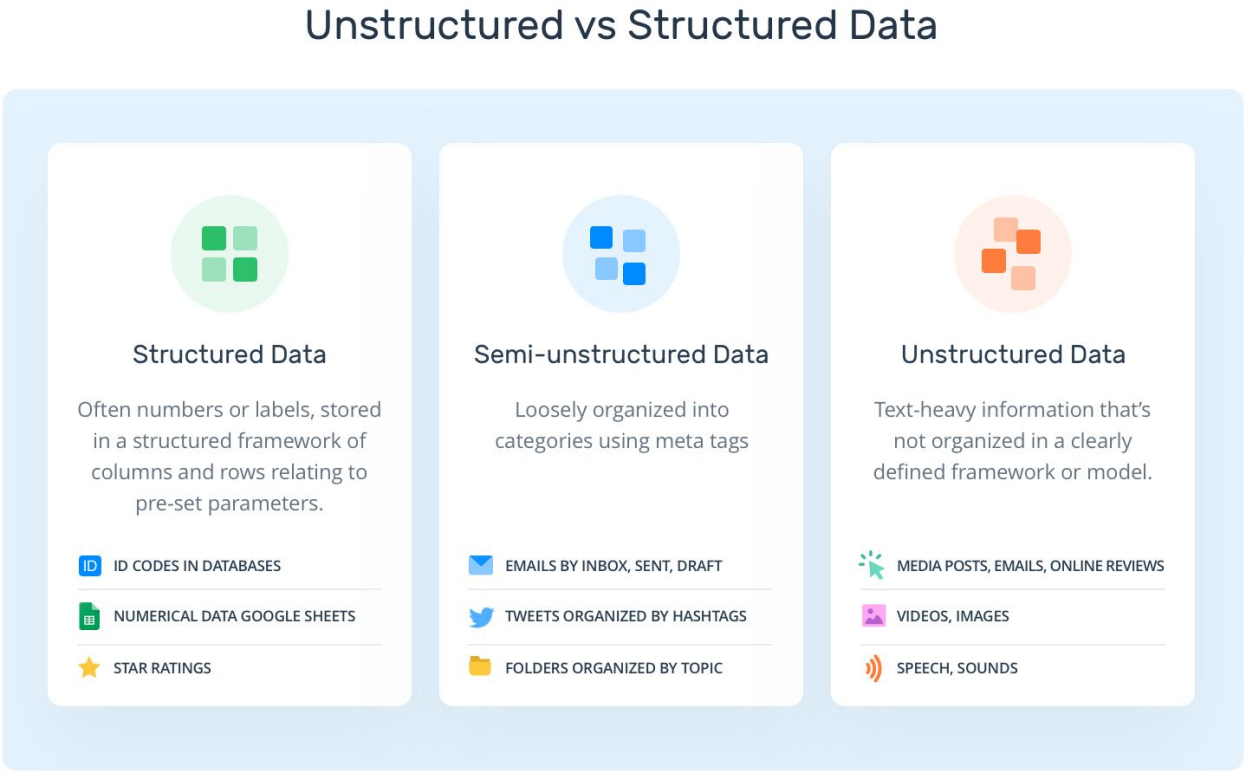
They have relational keys and can easily be mapped into pre-designed fields

Structured Data- Example

Patient Id	Name	D.o.B	Gender	Phone	Doctor Id	Doctor	Room
134	Jeff	4-Jul-1993	Male	7876453	01	Dr Hyde	03
178	David	8-Feb-1987	Male	8635467	02	Dr Jekyll	06
198	Lisa	18-Dec-1979	Female	7498735	01	Dr Hyde	03
210	Frank	29-Apr-1983	Male	7943521	01	Dr Hyde	03
258	Rachel	8-Feb-1987	Female	8367242	02	Dr Jekyll	06

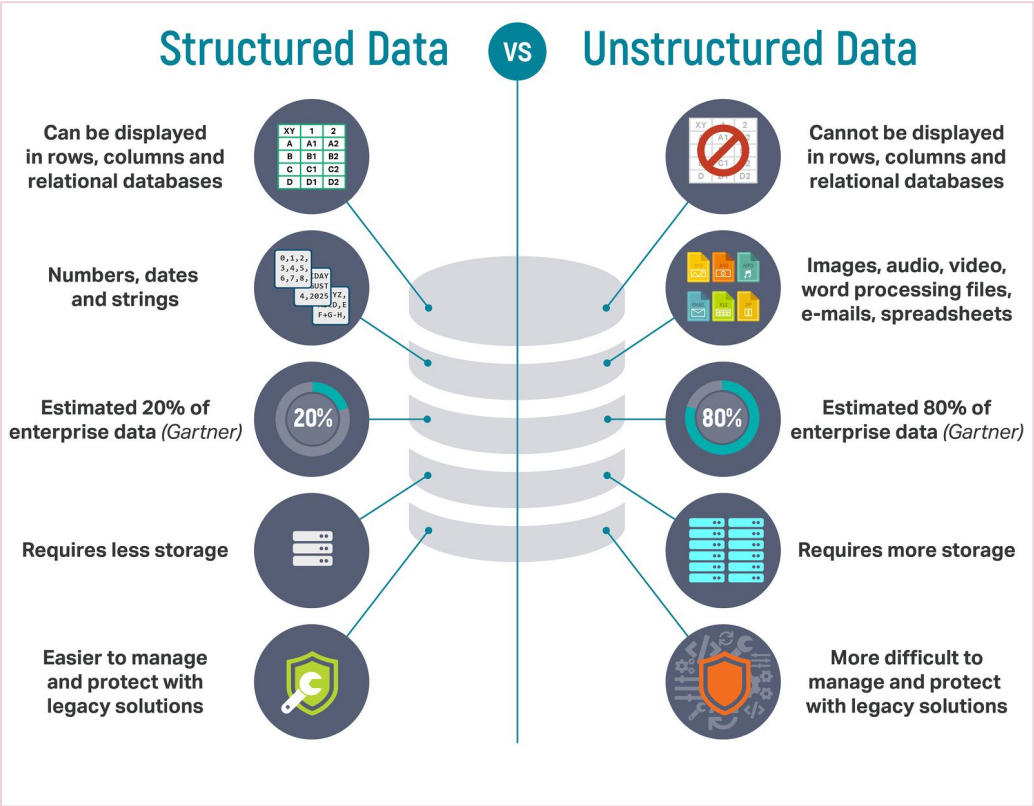
Source: Google

Structured vs Unstructured



Source:https://monkeylearn.com/blog/semi-structured-data/

Structured vs Unstructured



Source:https://lawtomated.com/tag/knowledge-management/

Unstructured Data

Unstructured data is data with no pre-defined organizational form or specific format.

Unstructured data is any data which is not structured or semi-structured.

The vast majority of all data created today is unstructured.

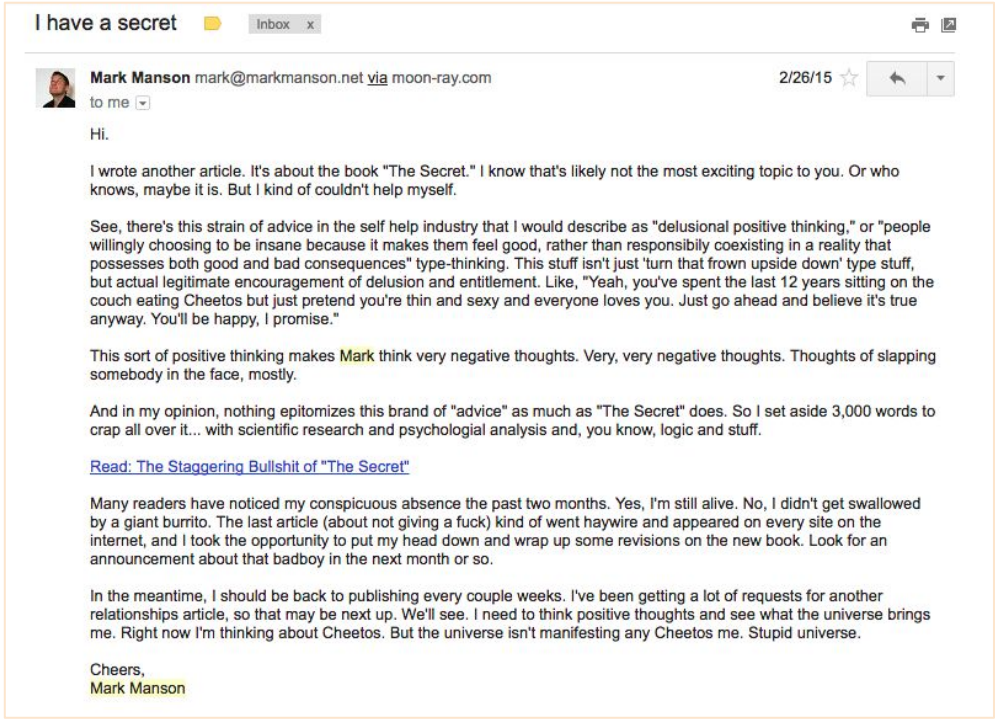
Unstructured Data

Just think of all the text, chat, video and audio content that is generated every day around the world!

Unstructured data is typically easy to consume for humans (e.g. images, videos and PDF-documents).

But due to the lack of organization in the data, it is less machine-readable.

Unstructured Data



I have a secret

Inbox x

Mark Manson mark@markmanson.net via moon-ray.com 2/26/15

to me

Hi.

I wrote another article. It's about the book "The Secret." I know that's likely not the most exciting topic to you. Or who knows, maybe it is. But I kind of couldn't help myself.

See, there's this strain of advice in the self help industry that I would describe as "delusional positive thinking," or "people willingly choosing to be insane because it makes them feel good, rather than responsibily coexisting in a reality that possesses both good and bad consequences" type-thinking. This stuff isn't just 'turn that frown upside down' type stuff, but actual legitimate encouragement of delusion and entitlement. Like, "Yeah, you've spent the last 12 years sitting on the couch eating Cheetos but just pretend you're thin and sexy and everyone loves you. Just go ahead and believe it's true anyway. You'll be happy, I promise."

This sort of positive thinking makes Mark think very negative thoughts. Very, very negative thoughts. Thoughts of slapping somebody in the face, mostly.

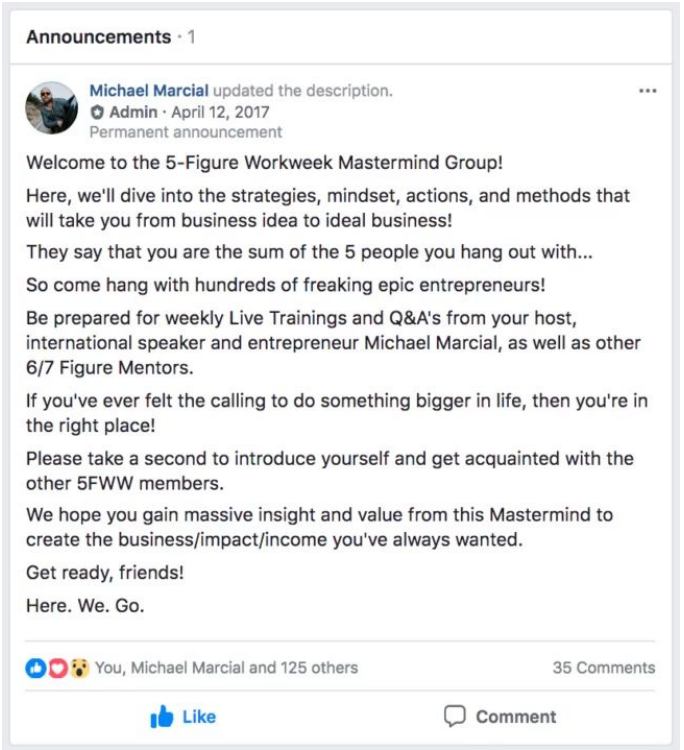
And in my opinion, nothing epitomizes this brand of "advice" as much as "The Secret" does. So I set aside 3,000 words to crap all over it... with scientific research and psychologial analysis and, you know, logic and stuff.

Read: The Staggering Bullshit of "The Secret"

Many readers have noticed my conspicuous absence the past two months. Yes, I'm still alive. No, I didn't get swallowed by a giant burrito. The last article (about not giving a fuck) kind of went haywire and appeared on every site on the internet, and I took the opportunity to put my head down and wrap up some revisions on the new book. Look for an announcement about that badboy in the next month or so.

In the meantime, I should be back to publishing every couple weeks. I've been getting a lot of requests for another relationships article, so that may be next up. We'll see. I need to think positive thoughts and see what the universe brings me. Right now I'm thinking about Cheetos. But the universe isn't manifesting any Cheetos me. Stupid universe.

Cheers,
Mark Manson



Announcements · 1

Michael Marcial updated the description.
Admin · April 12, 2017
Permanent announcement

Welcome to the 5-Figure Workweek Mastermind Group!

Here, we'll dive into the strategies, mindset, actions, and methods that will take you from business idea to ideal business!

They say that you are the sum of the 5 people you hang out with...

So come hang with hundreds of freaking epic entrepreneurs!

Be prepared for weekly Live Trainings and Q&A's from your host, international speaker and entrepreneur Michael Marcial, as well as other 6/7 Figure Mentors.

If you've ever felt the calling to do something bigger in life, then you're in the right place!

Please take a second to introduce yourself and get acquainted with the other 5FWW members.

We hope you gain massive insight and value from this Mastermind to create the business/impact/income you've always wanted.

Get ready, friends!

Here. We. Go.

You, Michael Marcial and 125 others 35 Comments

Like Comment

Source : Google

Semi structured Data

Semi-structured is data which has some degree of organization in it.

It is not as rigorously structured as structured data, but also not as messy as unstructured data.

Semi structured Data

This degree of organization is typically achieved with some sort of tags or other elements with defined properties which introduce a hierarchy and system into a file.

The order and amount of such structuring tags and elements may vary.

Eg: HTML / XML / JSON

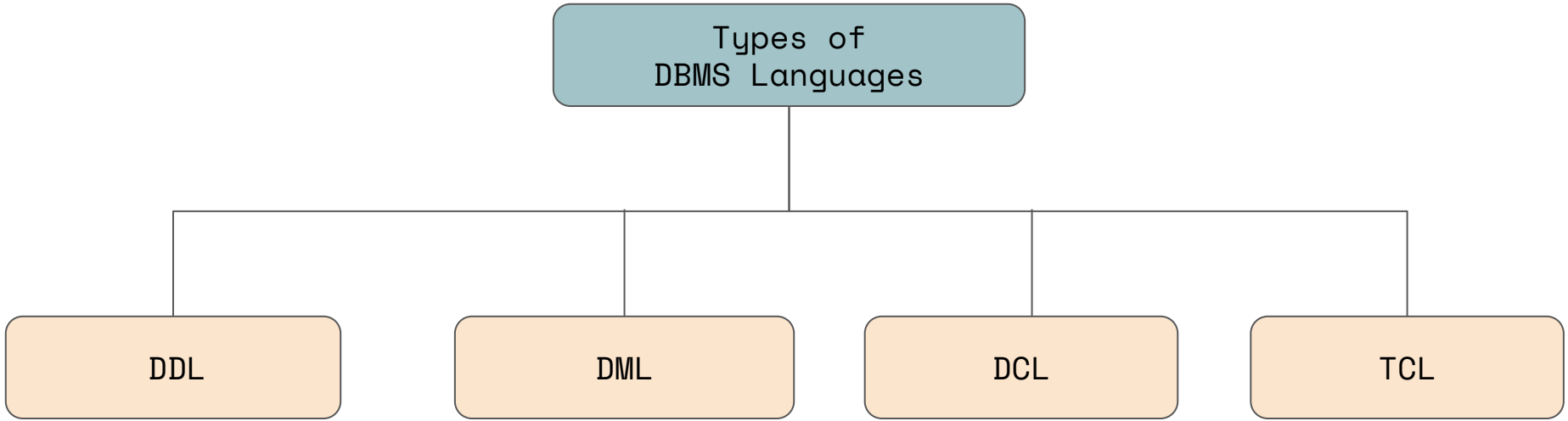
Semi structured Data

```
<Node>
<FirstName>Rachel</FirstName>
<LastName>Wolff</LastName>
</Node>
```

```
{
   "FirstName": "Rachel",
   "LastName": "Wolff"
}
```

Reading Material-Click Here

Database Languages



Data Definition Language(DDL)

It is used to define database structure or pattern.

It is used to create schema, tables, indexes, constraints, etc. in the database.

Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

Data Manipulation Language(DML)

Data Manipulation Language (DML) allows you to modify the database instance by inserting, modifying, and deleting its data.

It is responsible for performing all types of data modification in a database.

Eq: INSERT, UPDATE , DELETE

Data Control Language

DCL commands are used for access control and permission management for users in the database.

With them we can easily allow or deny some actions for users on the tables or records.

Eg: GRANT, DENY, REVOKE

Transaction Control Language

Transaction Control Language commands are used to manage transactions in the database. These are used to manage the changes made by DML-statements.

It also allows statements to be grouped together into logical transactions.

Eg: COMMIT, ROLLBACK, SAVEPOINT

Three Schema Architecture

GOAL !

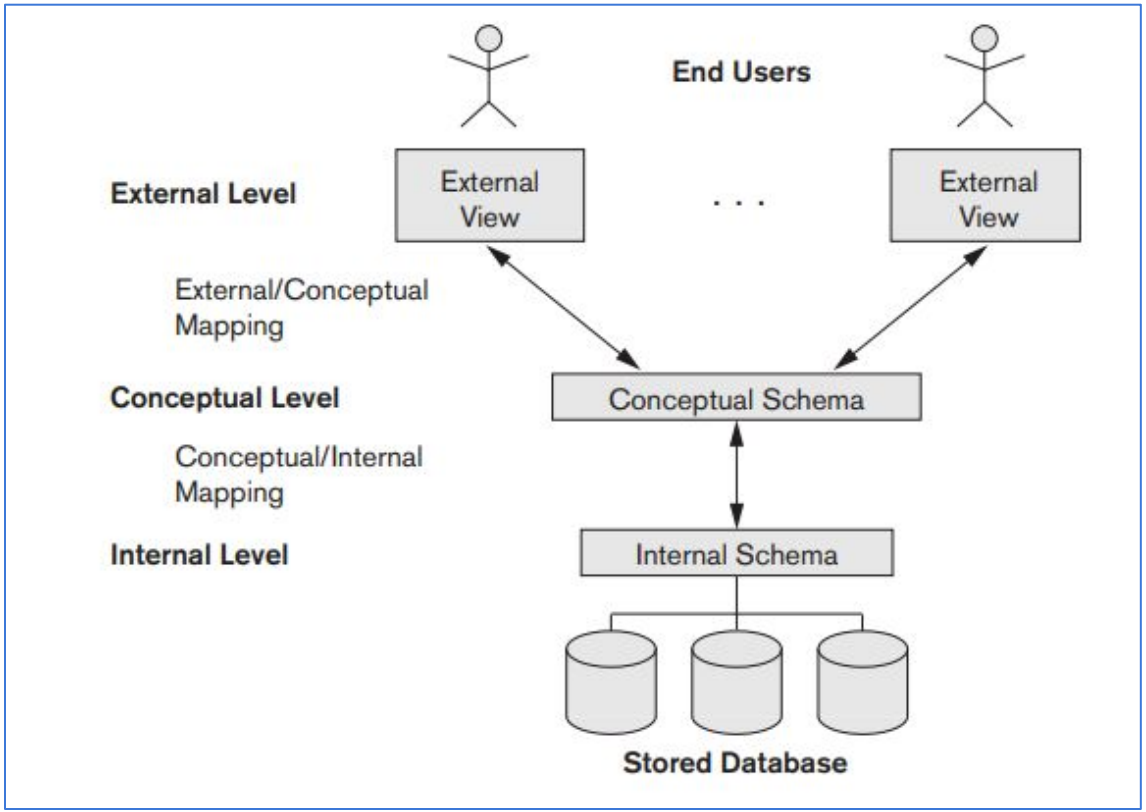
Separates the user applications from the physical database.

Was Proposed to help achieve and visualize the following characteristics

1. Use of a catalog to store the database description

2.insulation of programs and data

3. support of multiple user views



Internal Level

The internal level has an internal schema, which describes the physical storage structure of the database.

The internal schema uses a physical data model and describes the complete details of data storage and access paths for the database.

Conceptual Level(Logical Level)

The conceptual schema describes the design of a database at the conceptual level.

The conceptual schema describes the structure of the whole database.

Describes what data are to be stored in the database and also describes what relationship exists among those data.

Conceptual Level(Logical Level)

Internal details such as an implementation of the data structure are hidden.

Programmers and database administrators work at this level

External Level

At the external level, a database contains several schemas that sometimes called as subschema. The subschema is used to describe the different view of the database.

An external schema is also known as view schema.

External Level

Each view schema describes the database part that a particular user group is interested and hides the remaining database from that user group.

The view schema describes the end user interaction with database systems.

Data Independence

Data independence refers characteristic of being able to modify the schema at one level of the database system without altering the schema at the next higher level.

Logical Data Independence

Physical Data Independence

Logical Data Independence

Logical data independence is the capacity to change the conceptual schema without having to change external schemas or application programs.

Changes to constraints can be applied to the conceptual schema without affecting the external schemas or application programs.

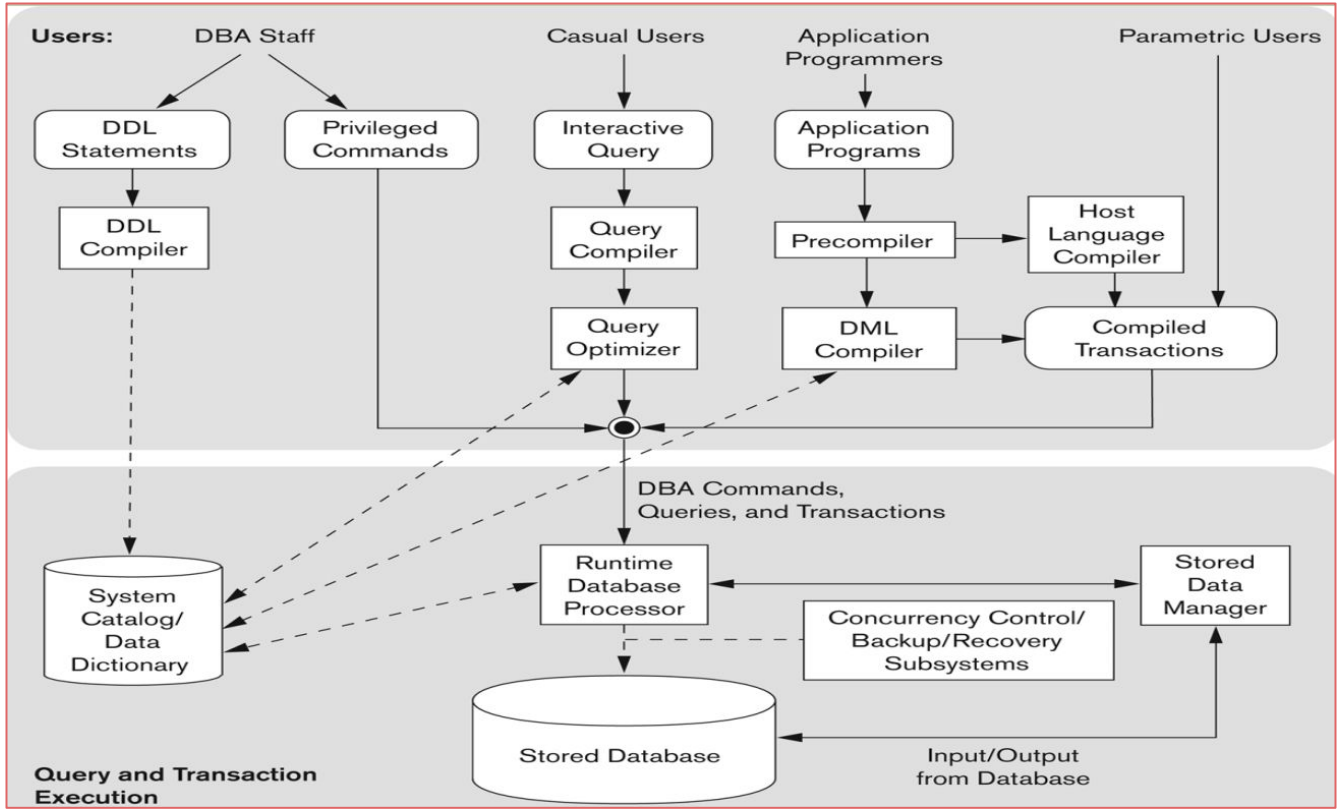
Physical Data Independence

Is the capacity to change the internal schema without having to change the conceptual schema.

The external schemas need not be changed.

Changes to the internal schema may be needed because some physical files were reorganized

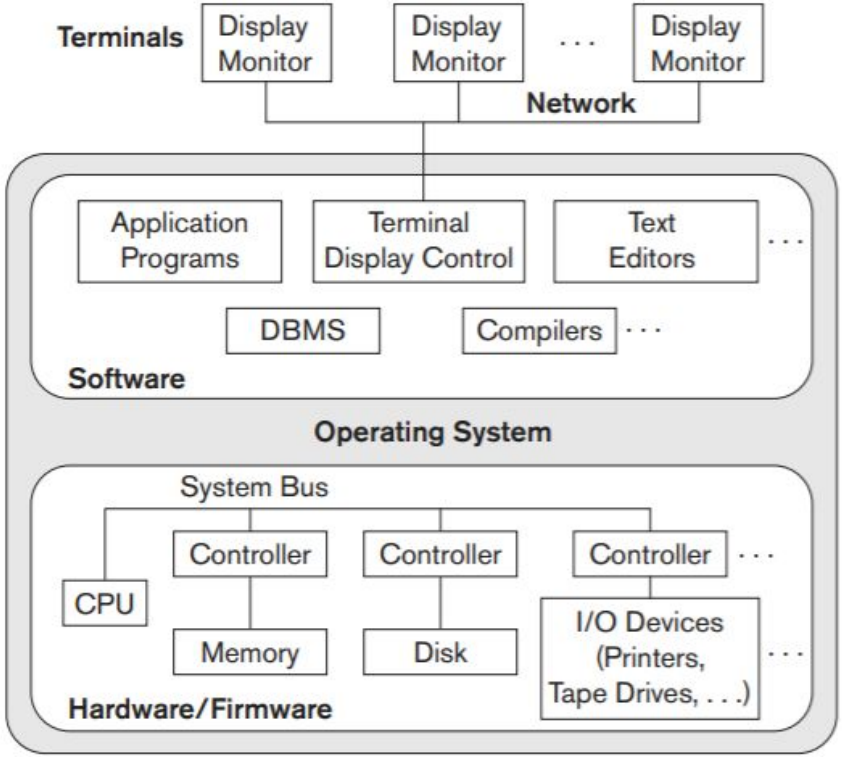
Component Modules of a DBMS



Database Architectures

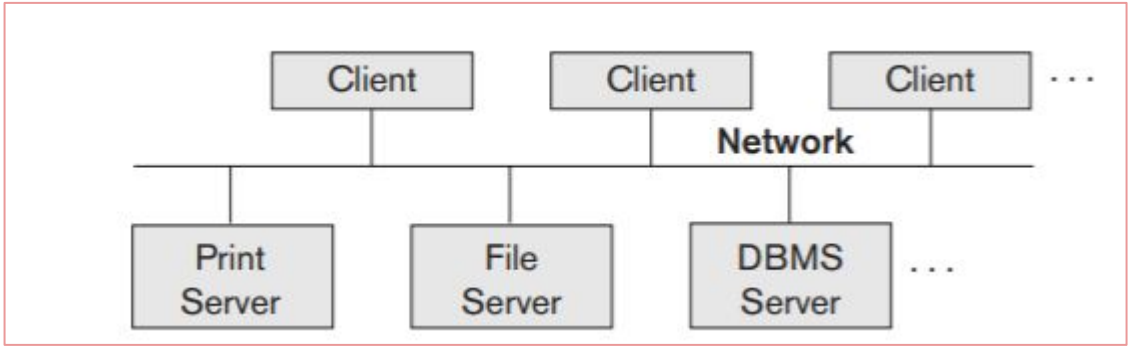
Centralized DBMS Architecture

All DBMS functionality, application program execution, and user interface processing are carried out on one machine.



Basic Client Server Architecture

Specific Functionalities

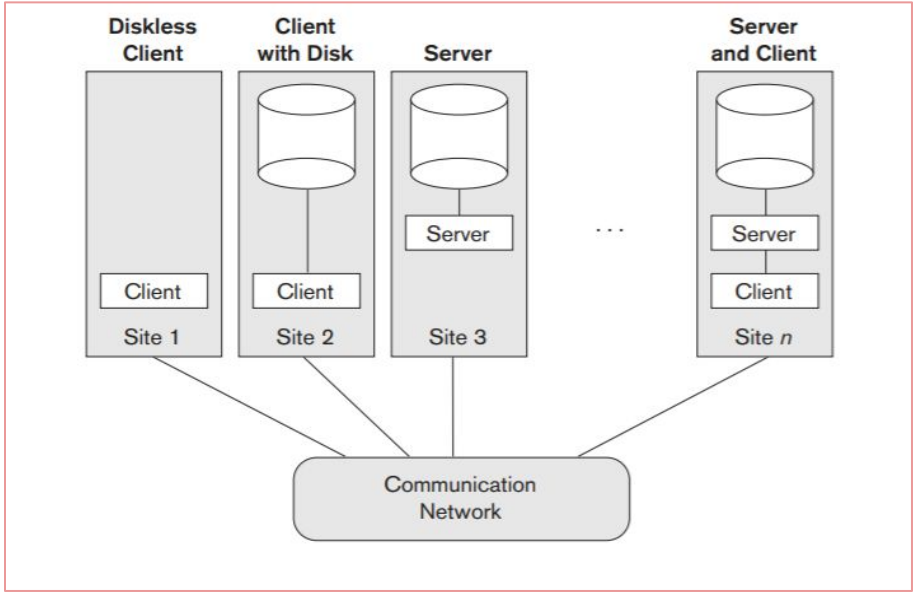


CSA was developed to deal with computing environments in which a large number of PCs, workstations, file servers, printers, database servers, Web servers, e-mail servers, and other software and equipment are connected via a network

Physical 2-Tier C-S Architecture

The user interface programs and application programs run on the client side.

When DBMS access is required, the program establishes a connection to the DBMS (server side);



Query and transaction functionality related to SQL processing remains on the server side.

Three Tier Architecture- for Web Applications

The intermediate server accepts requests from the client, processes the request and sends database queries and commands to the database server,

Data may be processed further and filtered to be presented to the users.