DATABASE SYSTEMS

**Bases of database systems: concept, characteristic, architecture. Data models.Normalization.Integrity constraint on data. Query tuning and their processing. Fundamentals of SQL. Parallel processing of data and their restoration. Design and development of databases.Technology of programming of ORM. The distributed, parallel and heterogeneous databases.**

<https://www.tutorialspoint.com/dbms/dbms_overview.htm>

<https://people.inf.elte.hu/miiqaai/elektroModulatorDva.pdf>

<https://theswissbay.ch/pdf/Gentoomen%20Library/Databases/Molina%2CUllman%20-%20Database%20Systems%20The%20Complete%20Book.pdf>

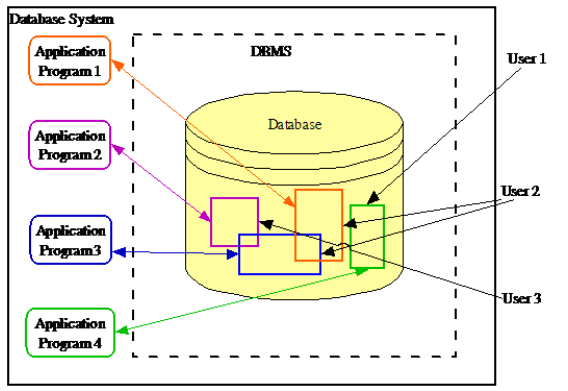
**BASES OF DATABASE SYSTEMS: CONCEPT, CHARACTERISTIC, ARCHITECTURE.**

Most likely you have heard about term «database». If you have some data, you need to store this data somewhere. This data could be anything: customer’s data, product data, employee data and etc. This data can come in many forms. Some common ones are text, numeric, multimedia, models, audio, code, software, video, and instrument.

Database is a collection of computerized data files that is managed by a DBMS (Database management system). Users of the system can perform a variety of operations involving such files:

* Adding a new files to the database
* Inserting data into existing files
* Retrieving data from existing files
* Deleting data from existing files
* Changing data in existing files
* Removing existing files from the database

Database management system (DBMS) is a software package designed to make all listed operations in database. Application program accesses the data stored in the database by sending requests to the DBMS. (picture)



The first database management systems evolved from file systems, which provide support the storage of very large amounts of data, file systems store data over a long period of time, and they allow the storage of large amounts of data. However, file systems do not generally guarantee that data cannot be lost if it is not backed up, and they don’t support efficient access to data items whose location in a particular file is not known. But modern DBMS has the following characteristics:

1. Sharing data – DBMS supports multi-user environment and allows them to access and manipulate data in parallel.

2. Reducing redundancy – DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values.

3. Avoiding inconsistency – DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems

4. Providing transactions – DBMS follows the concepts of Atomicity, Consistency, Isolation, and Durability (ACID). ACID properties help the database stay healthy in multi-transactional environments and in case of failure.

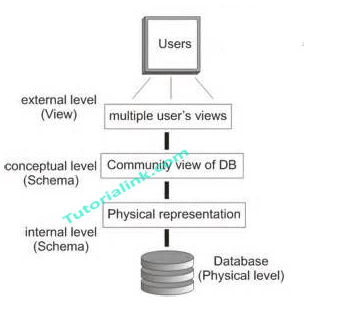
5. Maintaining integrity – DBMS can prevent the entry of invalid information into tables by using integrity constraints.

6. Enforcing security – DBMS offers many different levels of security features, which enables multiple users to have different views with different features.

We are now in a position to present an architecture for a database system. DBMS architecture describes how data in databases is viewed by the user. Today most of the DBMS architecture is based on the ANSI-SPARC database architecture.

The ANSI-SPARC architecture is divided into three levels, usually named as the internal, external and conceptual levels.

* The internal level (also known as storage level) is the one closest to physical storage, it is the one concerned with the way the data is stored inside the system.
* The external level (also known as user logical level) is the one closest to the users, it is the one concerned with the way the data is seen by individual users.
* The conceptual level (also known as the community logical level) is a level of indirection between the other two.

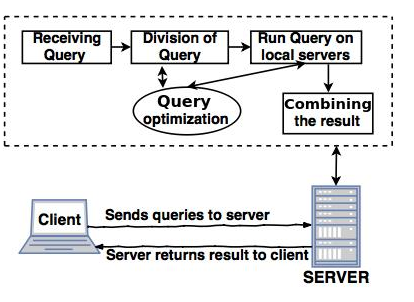


DBMS architecture

If we discussed before database systems from the point of view of the so-called ANSI/SPARC architecture, now we will offer you a slightly different perspective on this subject.

The overall purpose of a database system is to support the development and execution of database applications. From a high-level point of view, therefore, such a system can be regarded as having a very simple two-part structure, consisting of a server, also called the back end, and a set of clients, also called the front ends.

The server is just the DBMS itself. It supports all of the basic DBMS functions – data definition, data manipulation, data security and integrity, and so on.

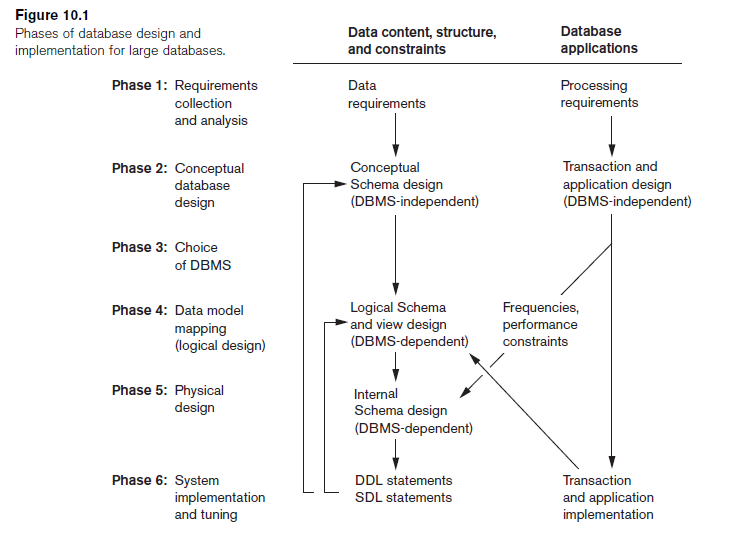


As you can see server architecture is responsible for executing the current transactions across the databases.

Database design

The goals of database design is:

* Satisfy the information content requirements of the specified users and applications
* Provide a natural and easy-to-understand structuring of the information
* Support processing requirements and any performance objectives, such as response time, processing time, and storage space.



We can identify siz main phases of the database design and implementation process:

1. Requirements collection and analysis

2. Conceptual database design

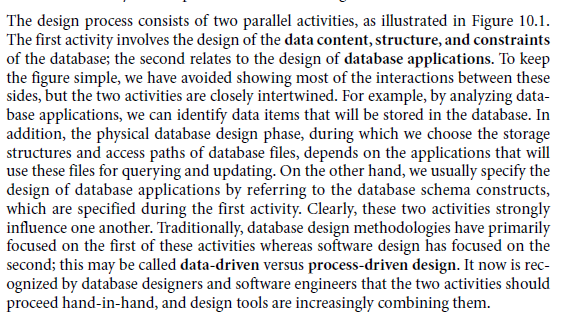
3. Choice of a DBMS

4. Data model mapping (logical database design)

5. Physical database design

6. Database system implementation and tuning

Design process, as you can see(picture), consists of two activities: first involves tht design of data content, structure and constraints of the database; the second relates to the design of database applications.



**DATA MODEL**

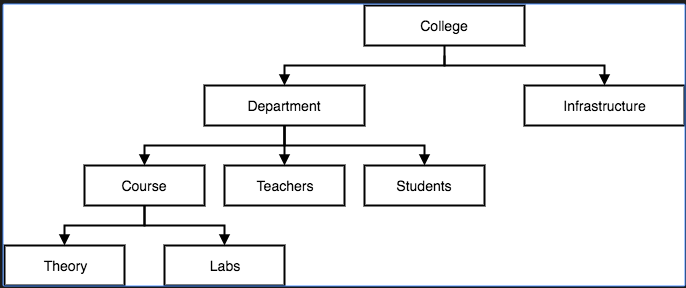
The notion of a “data model” is one of the most fundamental in the study of database systems. A data model is a notation for describing data or information. It defines how data is connected to each other and how they are processed and stored inside the system.

The description of data model generally consists of three parts:

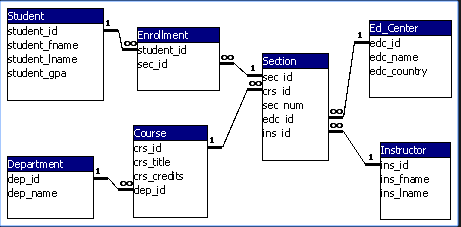
1. Structure of the data – is a collection of data structure types (the building blocks of any database that conforms to the model)
2. Operations on the data – is a collection of operators or inferencing rules, which can be applied to any valid instances of the data types.
3. Constraints on the data – is a collection of general integrity rules, which implicitly or explicitly the set of consistent database states.

Database management systems can be classified according to data model into:

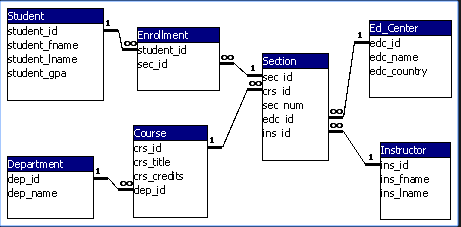
**Hierarchical**. The hierarchical data model represents data as a hierarchical tree structures. Each hierarchy represents a number of related records.



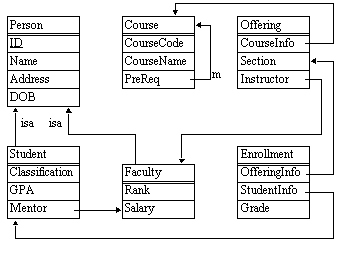
**Relational**. The relational data model represents a database as a collection of tables, where each table can be stored as a separate file.



**Network**. The network data model represents data as record types and also represents a limited type of 1:N relationship, called a set type.

****

**Object-oriented**. The object-oriented data model defines a database in terms of objects, their properties and their operations.



(Explanation of the schema <https://stackoverflow.com/questions/7815560/query-on-an-object-oriented-database>)

**FUNDAMENTALS OF SQL**

Whether you are writing applications, performing administrative tasks, generating reports, you will need to know how to manage and interact with data in your database. Despite the fact that today there are many types of database management system (DBMS), the relational database model is most popular and the SQL systems have come to dominate the DBMS marketplace.

SQL is the language for generating, manipulating, and retrieving data from a relational database. One of the reasons for the popularity of relational databases is that properly designed relational databases can handle huge amounts of data.

SQL includes both data definition language (DDL) component and a data manipulation language (DML) component. The SQL DML can operate at both the external level (on views) and the conceptual (on base tables). The SQL DDL can be used to define objects at external level (views), the conceptual level (base tables), and the interval level (indexes or other auxiliary structures).

Having defines the database, we can now start operating on it by means of the SQL manipulative operating SELECT, INSERT, DELETE and UPDATE.

SELECT – statement which selects data from database tables

INSERT – statement which inserts new data into database tables

DELETE – statement which deletes existing data from database tables

UPDATE – statement which updates existing data into database tables

DDL statements

CREATE – statement which creates database or its objects

DROP – statement which deletes objects from the database

ALTER – statement which alters the structure of the database

**Integrity constraint on data.**

As we mentioned, a data model has two parts: a manipulative part, defining the types of operation that are allowed on the data, and a set of integrity constraints, which ensure that the data is accurate.

There are two important integrity rules, which are constraints or restrictions that apply to all instances of the database. The two principal rules for the relational model are known as entity and referential integrity. Other types of integrity constraint are multiplicity and general constraints.

*Entity integrity* ensures that there are no duplicate records within the table and that the field that identifies each record within the table is unique and never null.

*Referential integrity* ensures the relationships between tables in a database remain accurate by applying constraints to prevent users or applications from entering inaccurate data or pointing to data that doesn’t exist.

*Multiplicity*

*General constraints*

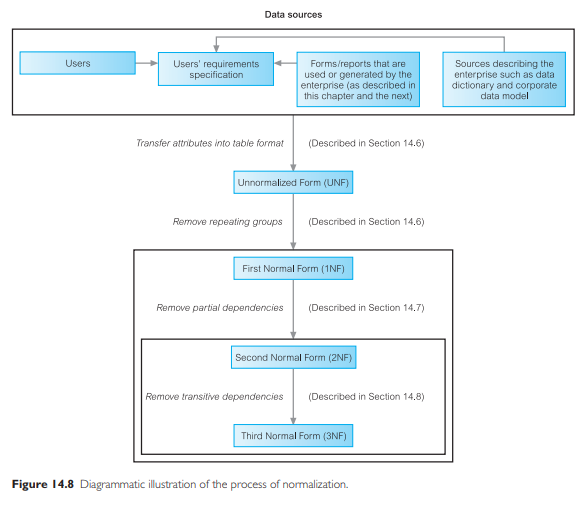
**Normalization**

When we design a database, the main objective is to create an accurate representation of data, relationships between the data, and constaints the data. To help achieve this objective, we can use more database design techniques. And of this techniques is normalization.

Normalization is a technique of organizing data in database. Normalization uses a series of tests (described as normal forms) to help identify the optimal grouping for these attributes to ultimately identify a set of suitable relations that supports the data requirements.

The characteristics of a suitable set of relations includes:

* the minimal number of attributes necessary to support the data requirements;
* attributes with a close logical relationship (described as functional dependency)
* minimal redundancy, with each attribute represented only once, with the important exception of attributes that form all ot part of foreign keys, which are essential for the joining of related relations



The steps of normalization are:

* Select the data source and convert into an unnormalized table (UNF)
* Transform the unnormalized data into first normal form (1NF)
* Transform data in first normal form (1NF) into second normal form (2NF)
* Transform data in second normal form (2NF) into third normal form (3NF)

Occasionally, the data may still be subject to anomalies in third normal form. In this case, we may have to perform further transformations.

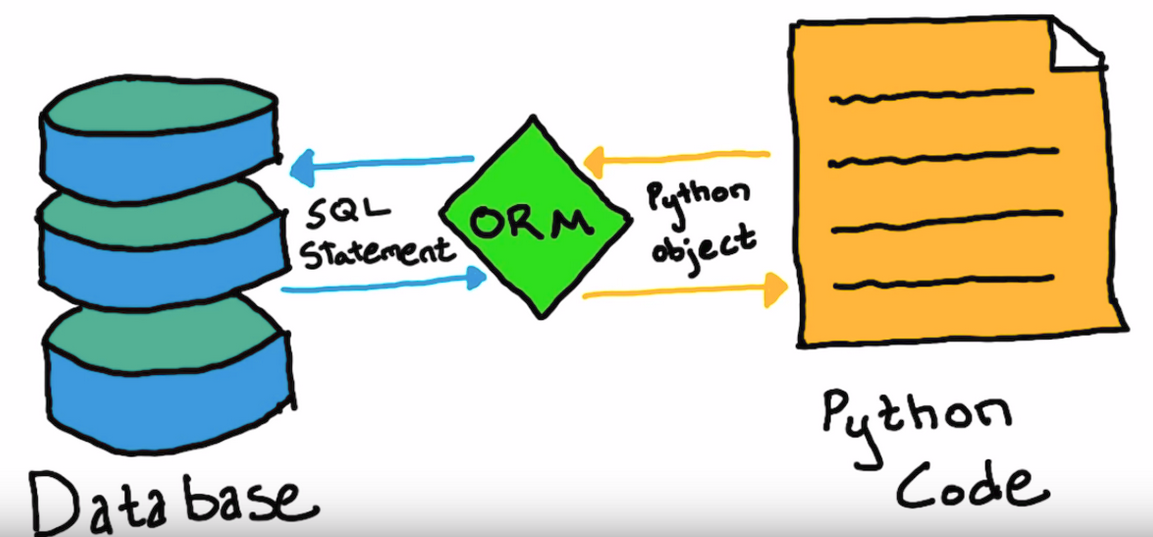
* Transform third normal form to Boyce-Codd normal form (BCNF)
* Transform Boyce-Codd normal form to fourth normal form (4NF)
* Transform fourth normal form to fifth normal form (5NF)

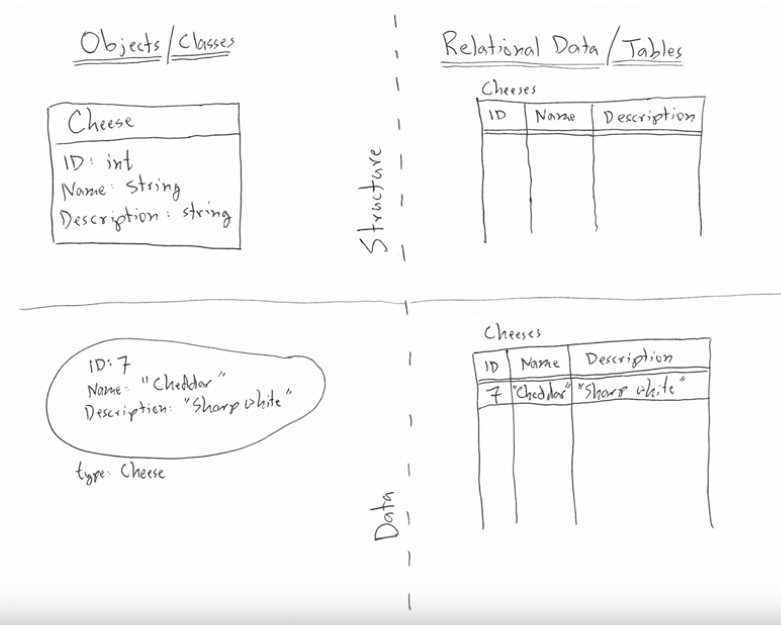
**Technology of programming of ORM (1200)**

Object-relational mapping (ORM) is a technique for storing, retrieving, updating, and deleting from an object-oriented program in a relational database.

ORM utilizes a “data layer” to manage translation between object-oriented and relational

The data layer is typically a library written in the object-oriented languages, like Java or C#, that is part of, or works in conjunction with your web framework.





**The distributed, parallel and heterogeneous databases.**