INTRO TO RDBMS-OODBMS-ORDBMS

Seema Sirpal

Delhi University Computer Centre

Databases

- Data - raw facts/details
- DATABASE- A shared collection of logically related data (and a description of this data), designed to meet the organization needs of an organization
- The Database Management System (DBMS) - software that enables users to define, create and maintain the database and provides controlled access to the database

Why We Need DBMS

- There is an information explosion in today's society
  - Need to keep accurate records
- Advantages of using a DBMS fall into three main categories:
  - Proper maintenance of the data
  - Providing access to the data
  - Maintaining security of the data

Requirements of a Database System

Efficiency

in the access to and modification of very large amounts of data

Resilience

in the ability of data surviving hardware crashes and software errors without sustaining loss or becoming inconsistent

Access Control

in the ability to permit simultaneous access of data by multiple users in a consistent manner and assuring only authorized access

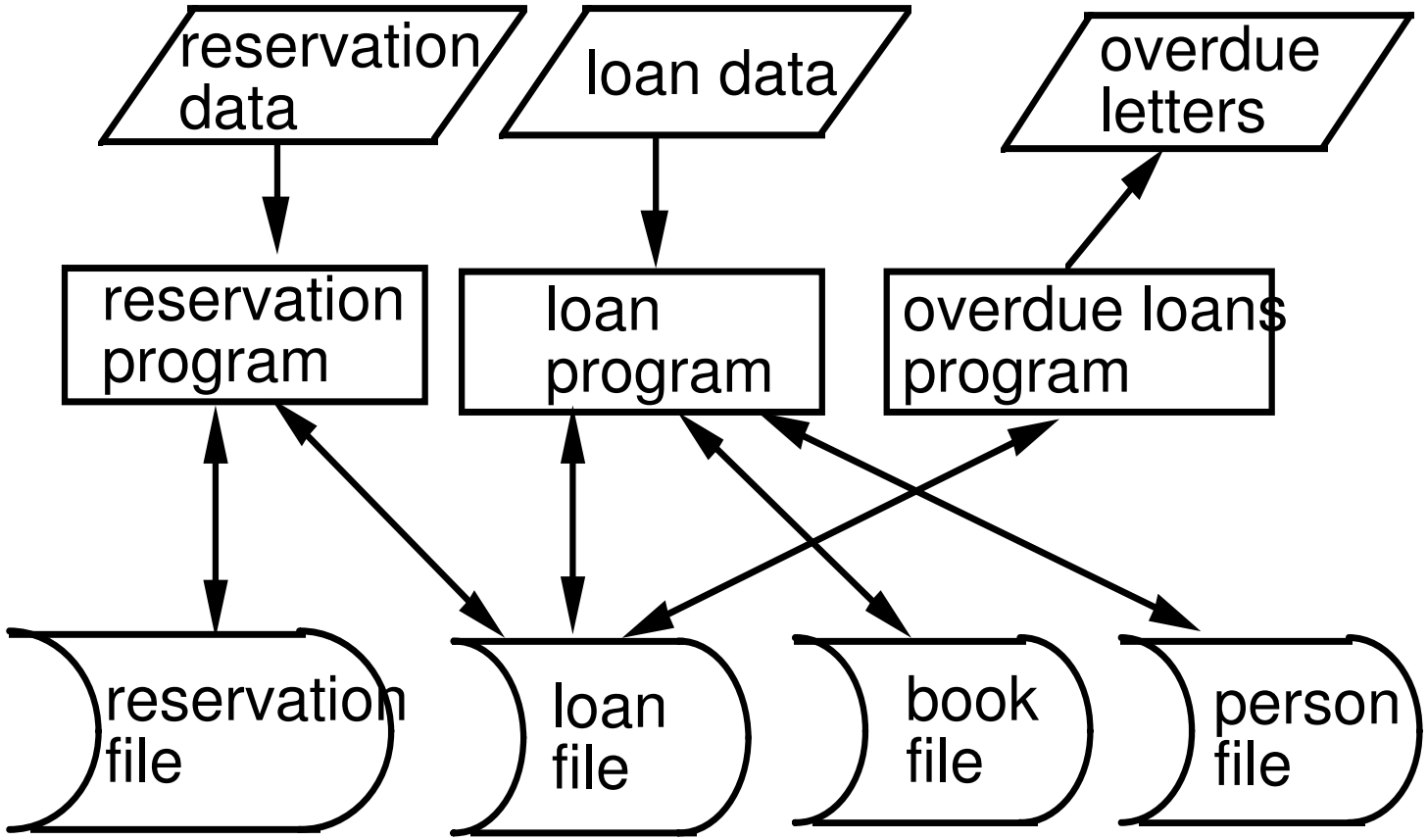
Persistence

the maintenance of data over long periods of time, independent of any programs that access the data

Traditional Approach

- Applications developed in an ad-hoc and opportunistic manner
- Data requirements for applications derived independently
- Data files developed for individual applications
- Application programs are data dependent

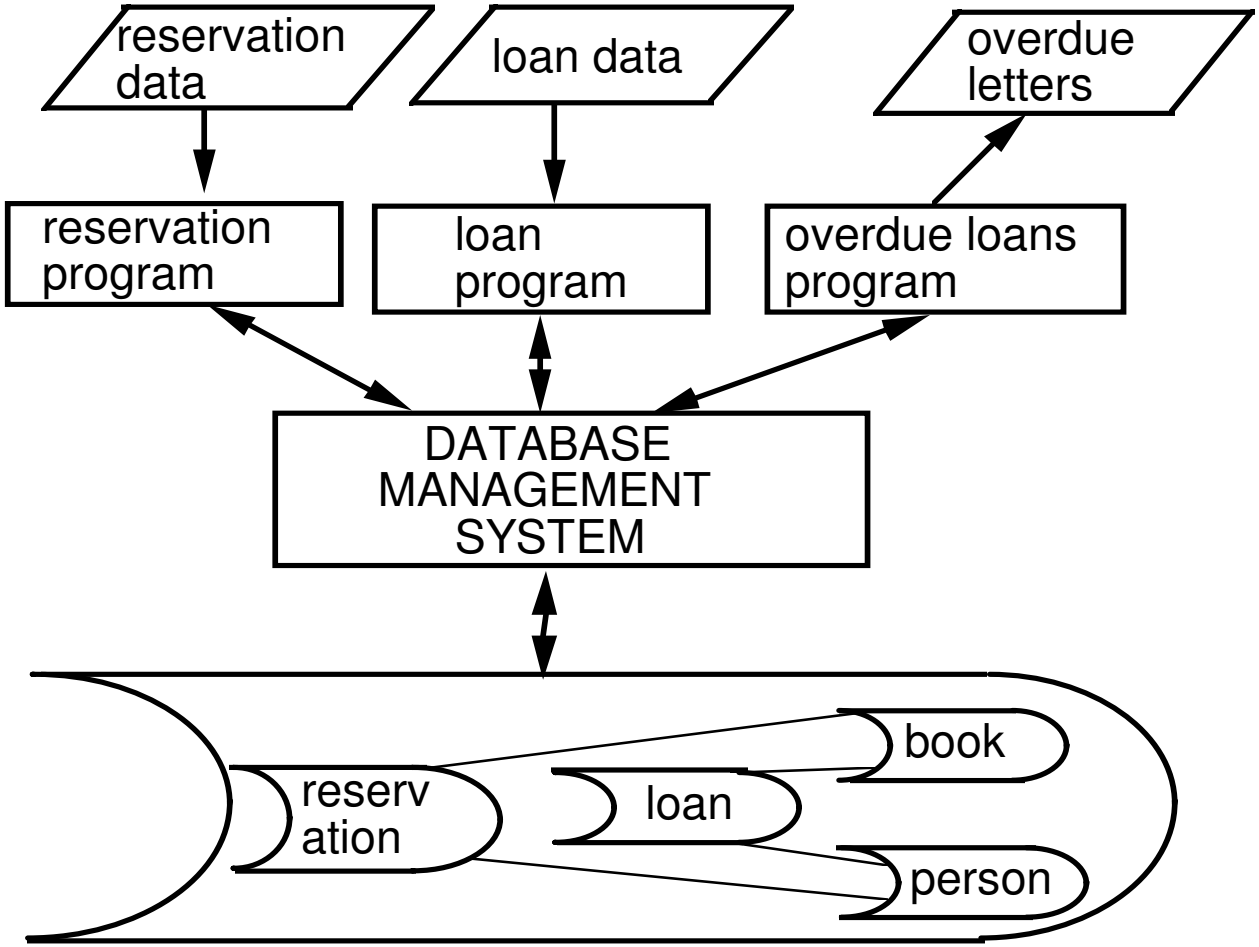
Files Dedicated to Application Programs



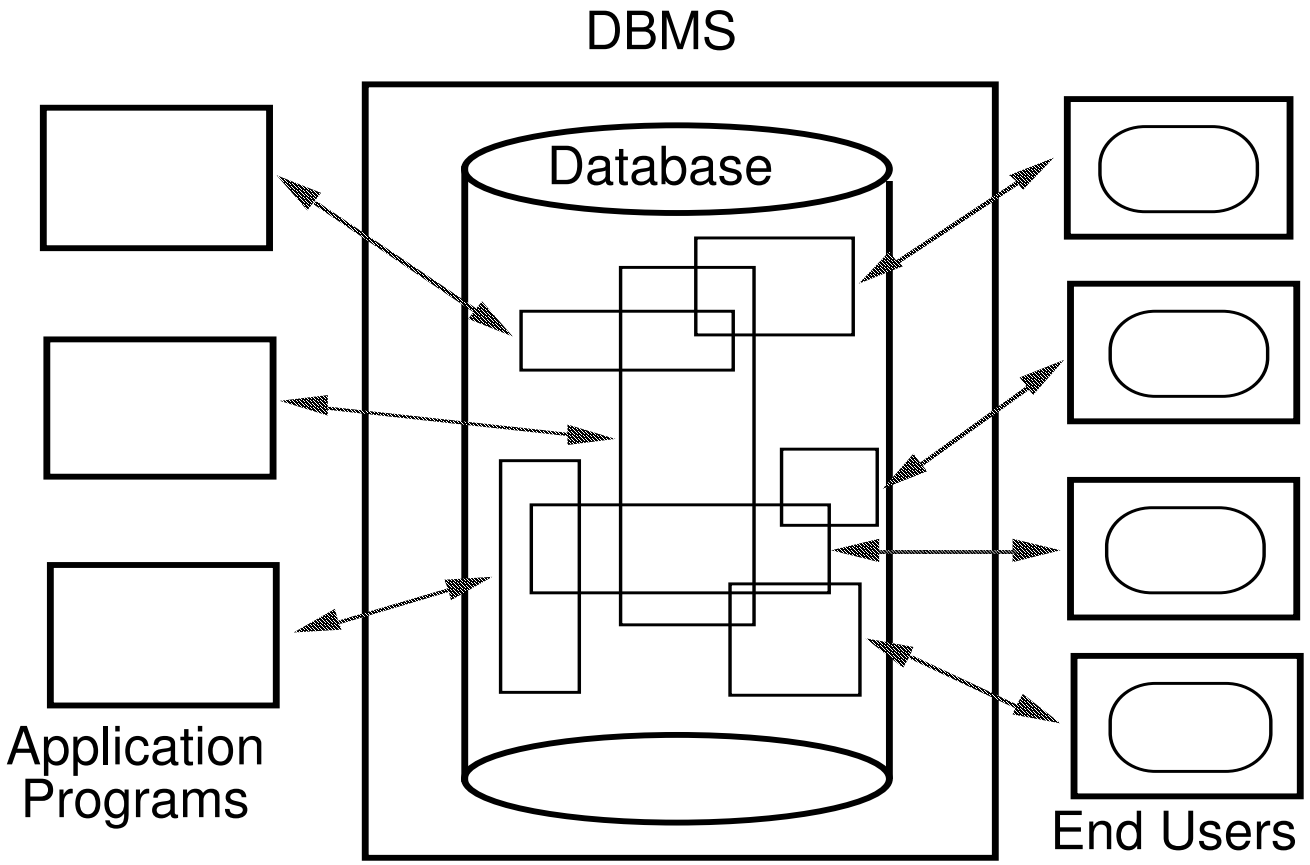
Database Approach

- Centralization of information management
- Data shared by different groups of users and application programs
- Provision of multiple interfaces
- Representation of complex relationships between data
- Integrity constraint handling
- Advanced facilities for backup and recovery

Data Sharing in a Database Environment



Interacting with a Database



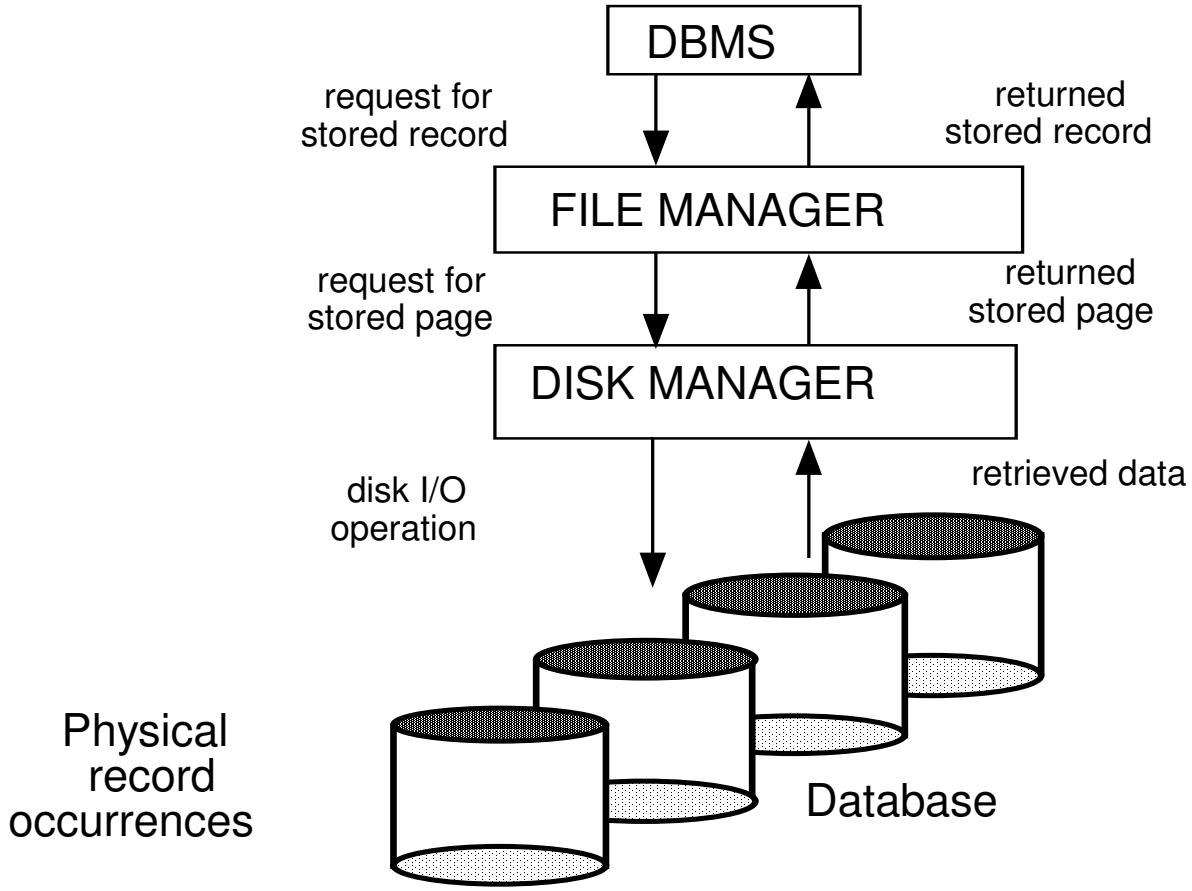
Advantages

- Sharing of data
- Enforcement of security
- Enforcement of development and maintenance standards
- Reduction of redundancy
- Avoidance of inconsistency across files
- Maintenance of integrity
- Data independence

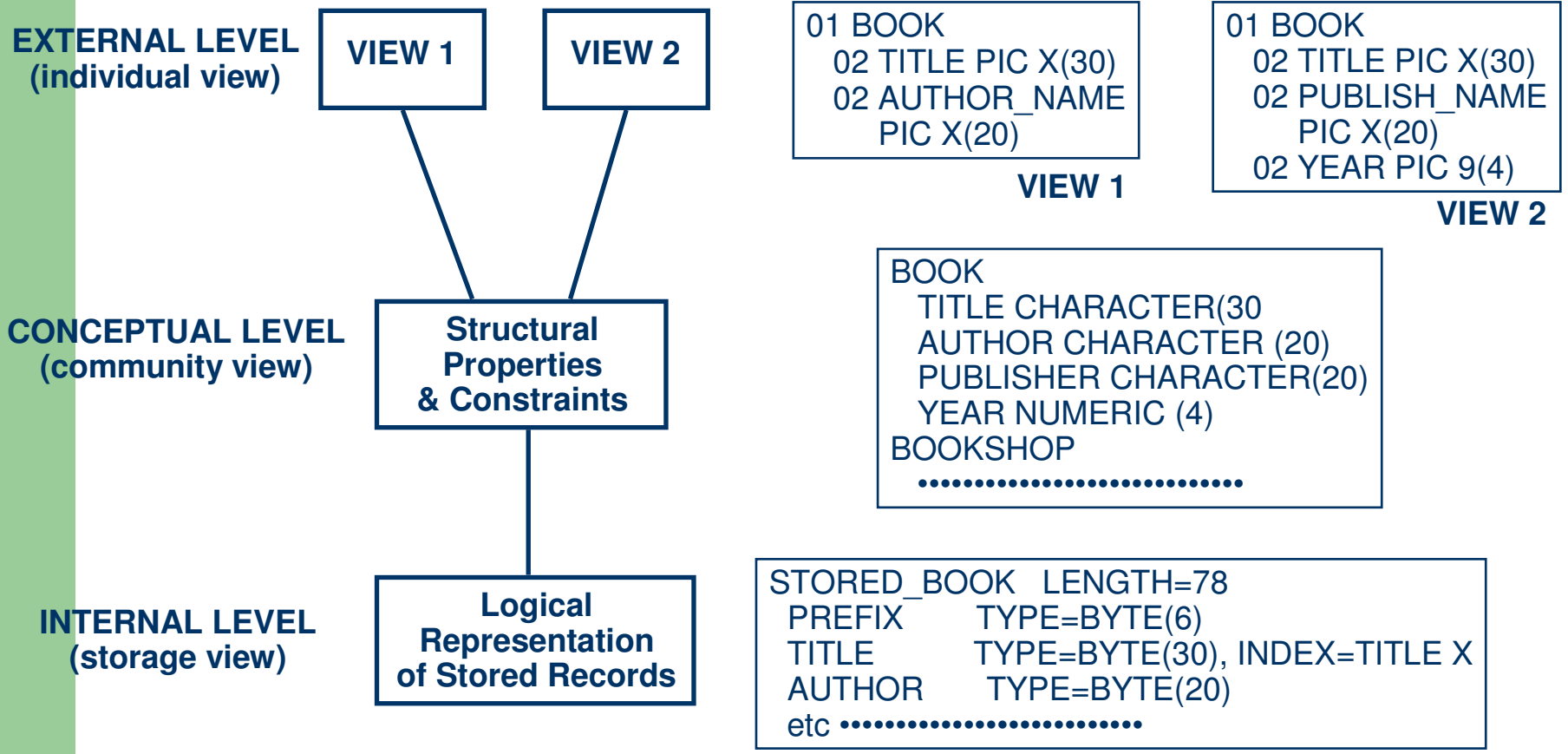
Functions of a DBMS

- Data storage, retrieval, update
- A user-accessible catalog
- Transaction support
- Concurrency control
- Recovery services
- Authorization services
- Integrity services
- Data independence
- Utility services

Interaction of DBMS with Operating System



3-levels Architecture



Complete DBMS

- Hardware
- Software
  - Utilities
- Data
- Users
- Procedures

Hardware

- The actual computer system used for keeping and accessing the database.
- Large organization usually has a network with a central server and many client programs running on desktops.
- Smaller organizations may have the DBMS and its clients reside in a single computer.

Software

- The actual DBMS.
  - Allows the users to communicate with the database.
  - Controls access
  - Helps maintain the consistency of the data
  - Includes utilities
    - Report writers
    - Application development tools
    - Design aids
  - Examples of DBMS software
    - Microsoft Access
    - Oracle Corporation Personal Oracle™
    - IBM DB2™

Data

- The database should contain all the data needed by the organization.
  - One of the major pieces of databases is the actual data are separated from the programs that use the data.
  - The set of facts represented in a database is called the Universe of Discourse (UOD).
  - The UOD should only include facts that are relevant to its users.
  - A database should always be designed, built and populated for a particular audience for a specific purpose.

Data continued…

- Emphasis is on the relevant data pertaining to one or more objects or entities.
  - Entity: a thing of significance about which information needs to be known.
- The characteristics that describe or qualify an entity are called attributes of the entity.

Data continued…

- For example:
  - In a student database, the basic entity is the student.
  - Entity attributes (information recorded about that entity) may include:
    - First and last name
    - Grade point average
    - Home address
    - Current address
    - Date of birth
    - Course opted

Data continued…

- The system would not be interested in
  - the type of clothes
  - the number of friends
  - the movies the student attends
  - etc.
- This information is not relevant to the user and should not be part of the UOD.

Data continued…

- For each attribute, the set of possible values that the attribute can take is called the domain of the attribute.
  - The domain of the date of birth would be all the dates that might be reasonable in the student body.
    - No date in the 1700s would be expected.
  - Undergraduate class levels would probably be restricted to
    - Part I
    - Part II
    - Part III
    - No other values would be allowed.

Users

- There are a number of users who can access or retrieve data on demand using the applications and interfaces provided by the DBMS.
- Each type of user needs different software capabilities:
  - The database administrator (DBA) is the person or group in charge of implementing the database system within the organization.
  - The end users are the people who sit at workstations and interact directly with the system.
  - The application programmers interact with the database by accessing the data from programs written in high-level languages such as Visual Basic or C++.

Maintaining a Database System - the Task of the DBA

- Maintaining information structure of the database (metadata in the Data Dictionary)
- Define user requirements of data usage
- Monitor use of database
- Determine and optimize storage structure and access strategies
- Define authorization checks
- Define strategies for backup and recovery
- Define testing procedures and changeover policy

Procedures

- An integral part of any system is the set of procedures that control the behavior of the system.
  - The actual practices the users follow to obtain, enter, maintain, and retrieve the data.
    - For example, in a payroll system, how are the hours worked received by the clerk and entered into the system?
    - Exactly when are monthly reports generated and to whom are they sent?

Data Models

- Models generally allow people to conceptualize an abstract idea more easily
  - Model airplanes
  - Model homes
- A data model is a way of explaining the logical layout of the data and the relationship of various parts to each other and the whole.
  - Different data models have been used throughout the years.

Data Models continued…

- In the early years, a flat file system or a simple text file with all the data listed in some order was used.
- When only large mainframes were available, both the hierarchical model and the network model were prevalent.
- Then, in 1970, the relational database model became the standard.

Classification of DBMS

1. **Classical DBMS**
   - Hierarchical
   - Network
   - Relational
2. **New Directions**
   - Extended Relational
   - Object-Oriented
   - Distributed

Data Models continued…

- The Relational Database Model:
  - Relational database management systems, where all data are kept in tables or relations.
  - More flexible & easy to use.
  - Almost any item of data can be accessed more quickly than the other models.
  - Retrieval time is reduced so that interactive access becomes more feasible.
- This is what is referred to as Relational Database Management Systems (RDBMS)

Relational DBMS (RDBMS)

- Edgar F. Codd at IBM invented the relational database in 1970. Referred to as RDBMS
- RDBMS allows operations in a human logical environment.
- The main elements of RDBMS are based on Codd's 13 rules for a relational system.
- The relational database is perceived as a collection of tables.
- Each table consists of a series of row/column intersections.
- Tables (or relations) are related to each other by sharing a common entity characteristic.

RDBMS (Continued)

- Advantages
  - Improved conceptual simplicity
  - Easier database design, implementation, management, and use
  - Ad hoc query capability (SQL)
  - Powerful database management system
- Disadvantages
  - Possibility of poor design and implementation
  - relational databases do not have enough storage area to handle data such as images, digital and audio/video.
  - the requirement that information must be in tables where relationships between entities are defined by values.

Object Oriented Database - OODBMS

- Main objective of OODBMS, is to provide consistent, data independent, secure, controlled and extensible data management services to support the object-oriented model. They were created to handle big and complex data that relational databases could not.

- The most important characteristic is the joining of object-oriented programming with database technology, which provides an integrated application development system. Object-oriented programming results in 4 main characteristics: inheritances, data encapsulation, object identity, and polymorphism.

Object Oriented Database - OODBMS

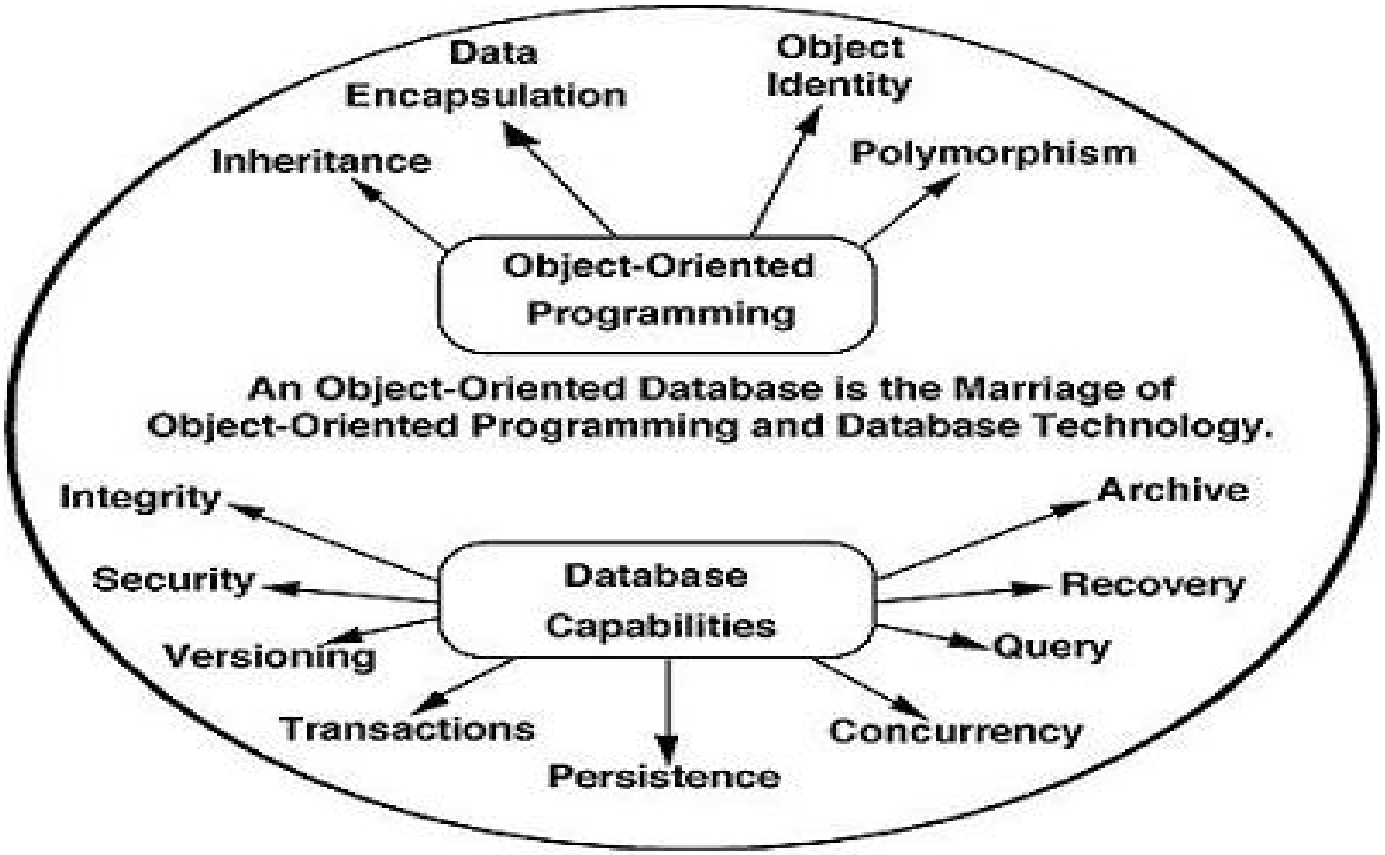


Figure 1. Makeup of an Object-Oriented Database

Object Relational - ORDBMS

- The traditional RDBMS extended to include Object Oriented concepts and structures such as abstract datatype, nested tables and varying arrays.
- ORDBMS was created to handle new types of data such as audio, video, and image files that relational databases were not equipped to handle. In addition, its development was the result of increased usage of object-oriented programming languages, and a large mismatch between these and the DBMS software.
- Advantages of ORDBMS
  - it allows organizations to continue using their existing systems, without having to make major changes.
  - it allows users and programmers to start using object-oriented systems in parallel.

RDBMS

- We will use the terms tables and relations interchangeably.
  - In a RDBMS, the data is logically perceived as tables.
    - Tables are logical data structures that we assume hold the data that the database intends to represent.
    - Tables are not physical structures.
    - Each table has a unique name.

RDBMS - Tables

- Tables consist of a given number of columns or attributes.
  - Every column of a table must have a name.
  - No two columns of the same table may have identical names.
  - The total number of columns or attributes that comprises a table is known as the degree of the table.
    - In this class, we use the terms column and attribute interchangeably.

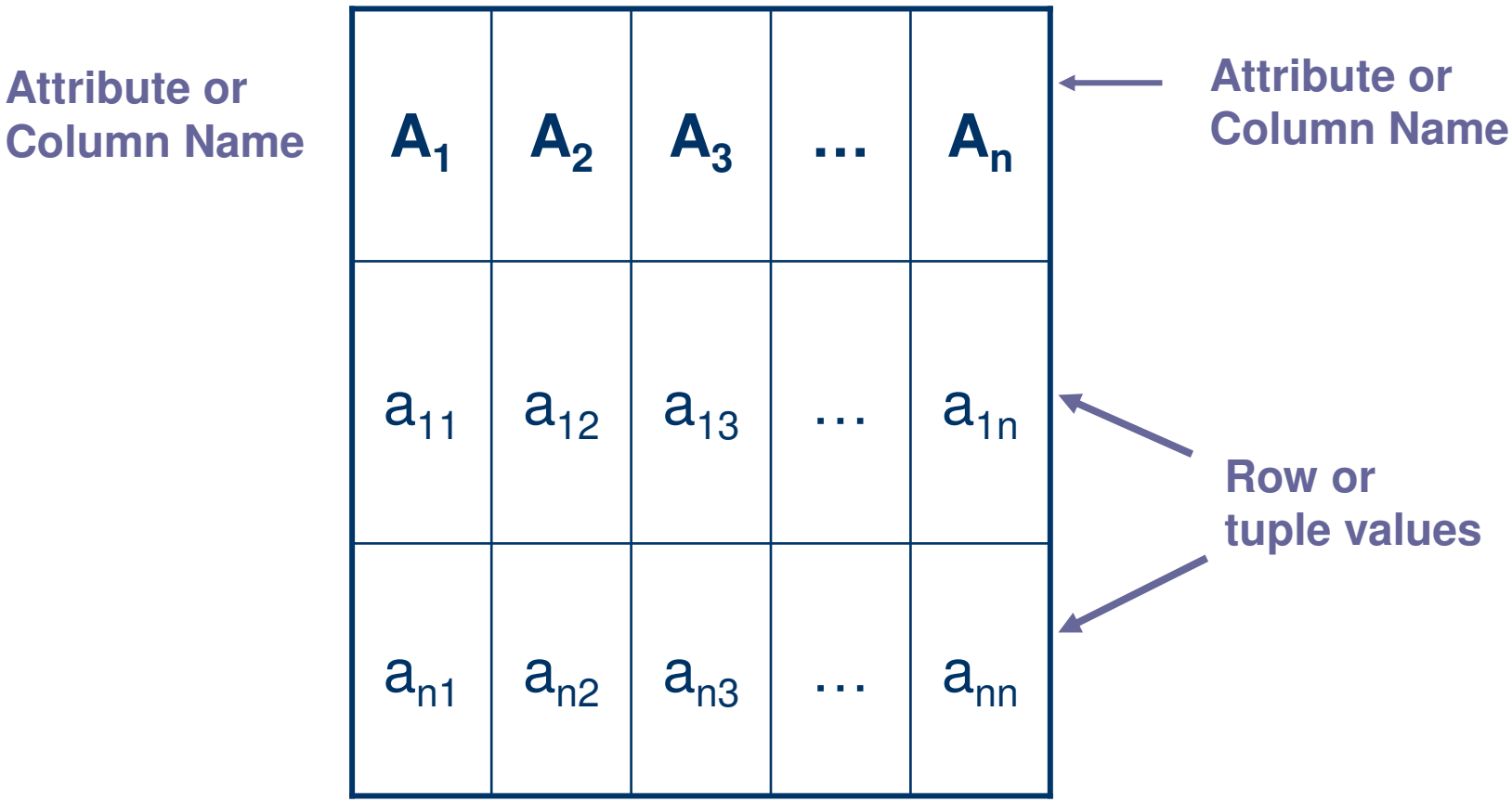
RDBMS - Tables

- The data in the table appears as a set of rows or n-tuples.
  - Where n is the number of attributes of the table.
  - Whenever the number of attributes of the table is understood, we can omit the prefix n and refer to the rows of the table as just rows or tuples.
  - All rows of a table have the same format and represent some object or relationship in the real world.

RDBMS - Tables

- The total number of rows present in a table *at any one time* is known as the cardinality of the table.
- In legacy systems, the terms field and record are used as synonyms of the terms attribute and row respectively.

General format of a relation when represented as a table



RDBMS - Tables

- We will call the content of a table at any particular point in time a snapshot or instance of the table.
  - In general, when tables are defined, the number of columns remain fixed for the duration of the table, however, the number of rows present in the table is bound to vary
  - Tables are required to have at least one column.
  - Tables are not required to have rows.
  - A table with no rows is called an empty table.
  - The process of inserting tuples for the very first time into a table is called populating the table.

RDBMS - Tables

- Whenever a relation is represented by means of a table, we will assume the following conditions hold:
  - The table has a unique name
  - Each column of the table has a unique name.
    - No two columns of the same table may have identical names
  - The order of the columns within the table is irrelevant
  - All rows of the table have the same format and the same number of entries.

RDBMS - Tables

- The values under each column belong to the same domain
  - Strings of characters
  - Integer values
  - Real values
  - Etc.
- Every entry (the intersection of a row and column) of every tuple of the relation must be a single value
  - No list or collection of values is allowed

Candidate Key and Primary Key of a Relation

- The notion of a key is a fundamental concept in the relational model
  - Provides the basic mechanism for retrieving tuples within any table of the database.
    - To distinguish a candidate key
      - No two different tuples of the relation will have identical entries in all attributes of the key
      - The number of attributes that comprises the key must be minimal

Candidate Key and Primary Key of a Relation

- Since a relation may have more than one candidate key, one of these candidate keys should be designated as the primary key (PK) of the relation.
  - The values of the primary key can be used as the identification and addressing mechanism of the relation.
  - We will differentiate between the different rows of the relation on the basis of their PK values.
  - We will also uniquely retrieve tuples from a relation based on the values of their PK values.

Candidate Key and Primary Key of a Relation

- Once a primary key has been selected, the remaining candidate keys, if they exist are called alternate keys.
- A RDBMS allows only one primary key per table.
- A primary key may be composed of
  - a single attribute (single primary key)
  - More than one attribute (composite primary key)

Candidate Key and Primary Key of a Relation

- Since the primary key is used to identify the tuples or rows of a relation, none of its attributes may be NULL.
  - In a relation, the NULL value is used to represent missing information, unknown, or inapplicable data.
    - A NULL value is not a zero value
    - A NULL value doesn't represent a particular value within the computer.
  - This imposes an additional condition or constraint on the keys known as the integrity constraint.

Foreign Keys

- Because columns that have the same underlying domain can be used to relate tables of a database, the concept of foreign key (FK) allows the DBMS to maintain consistency among the rows of two relations or between the rows of the same relation.
  - The attributes of a FK have the same underlying domain as the set of attributes defined for the PK.
  - The FK values in any tuple of a relation are either NULL or must appear as the PK values of a tuple of the corresponding relation.

Foreign Keys

- The table that contains the foreign key is the child table
- The table that contains the referenced attribute(s) is the parent table.
- The FK value in each row of a child table is either
  - NULL or
  - Must match the PK value of a tuple of the parent table.

Foreign Keys

- Foreign keys are usually defined after all the tables have been created and populated.
  - Avoids the problem of circularity:
    - Occurs when one table references values in another table, which in turn may reference the first table.
  - Integrity constraints are automatically enforced by the RDBMS.

Data and Relation

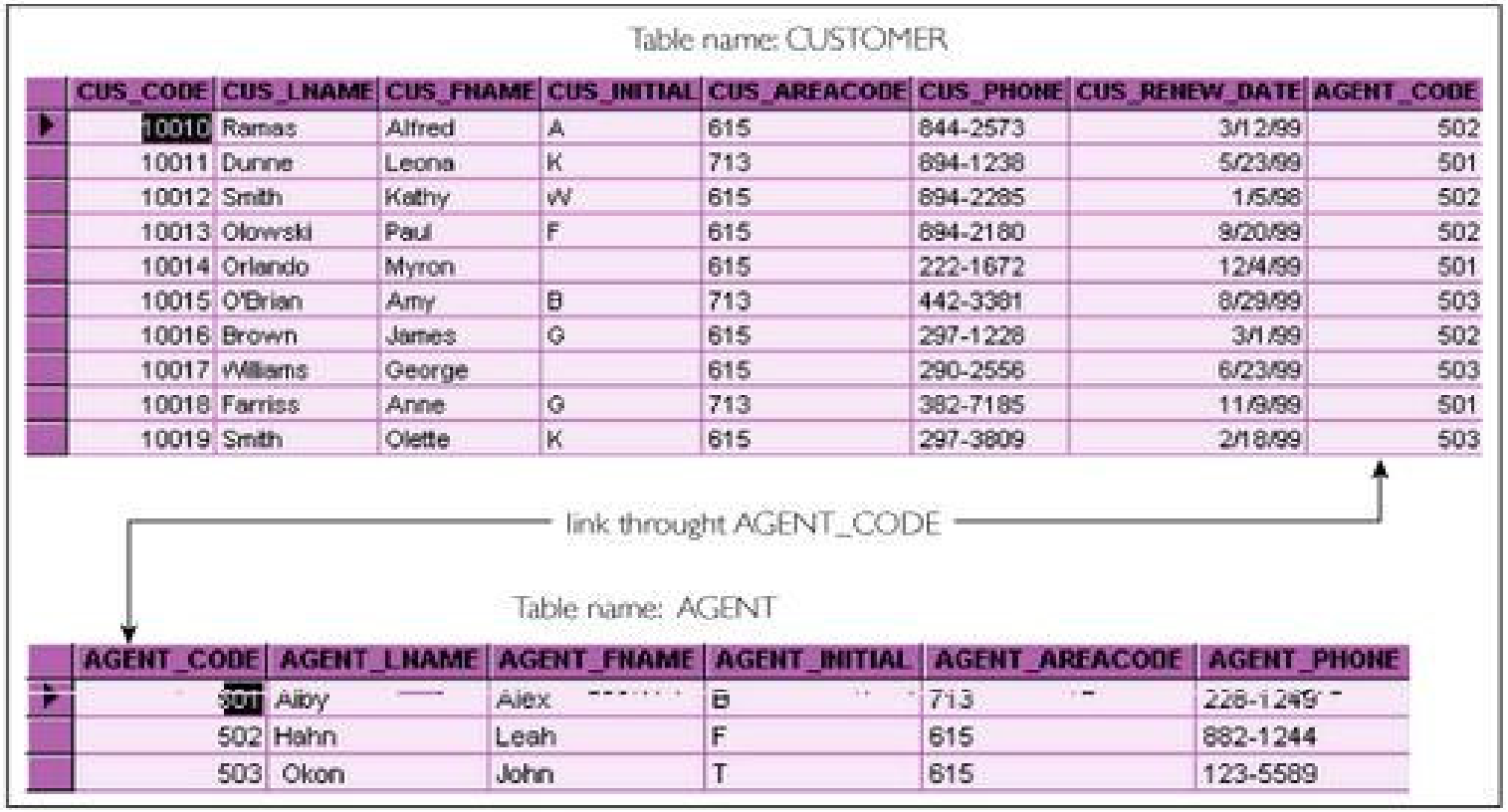


Table name: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	615	844-2573	3/12/99	502
10011	Dunne	Leona	K	713	894-1238	5/23/99	501
10012	Smith	Kathy	W	615	894-2285	1/5/98	502
10013	Olowski	Paul	F	615	894-2180	9/20/99	502
10014	Orlando	Myron		615	222-1672	12/4/99	501
10015	O'Brian	Amy	B	713	442-3381	8/29/99	503
10016	Brown	James	G	615	297-1228	3/1/99	502
10017	Williams	George		615	290-2556	6/23/99	503
10018	Farriss	Anne	G	713	382-7185	11/8/99	501
10019	Smith	Olette	K	615	297-3809	2/18/99	503

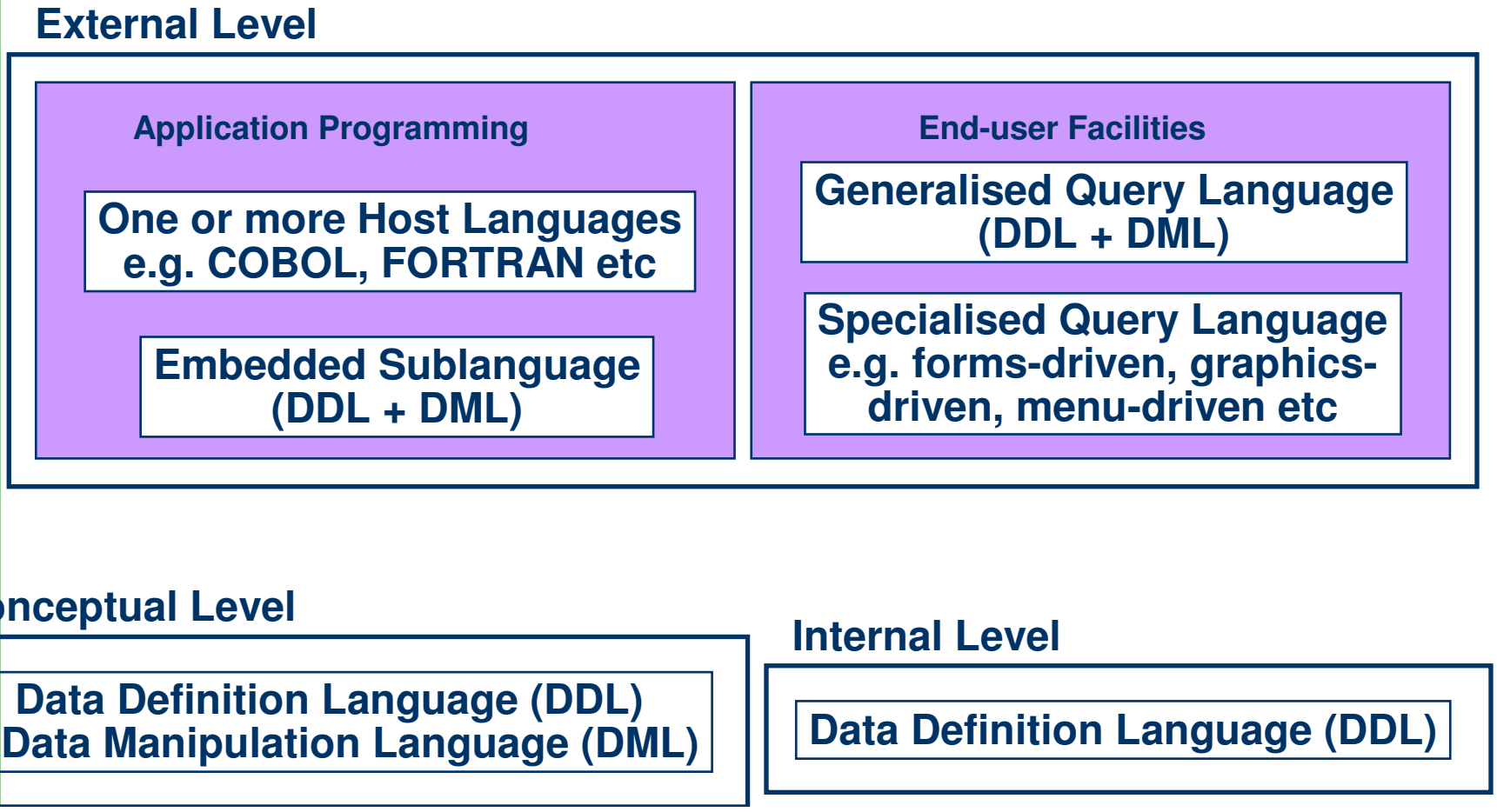
link throught AGENT_CODE

Table name: AGENT

AGENT_CODE	AGENT_LNAME	AGENT_FNAME	AGENT_INITIAL	AGENT_AREACODE	AGENT_PHONE
501	Alby	Alex	B	713	228-1249
502	Hahn	Leah	F	615	882-1244
503	Okon	John	T	615	123-5589

FIGURE 1.11 LINKING RELATIONAL TABLES

Languages in a Database Environment



Data Definition - Functionality of DDL

- Basic functions include:
  - creation of data structure supported by data model e.g. CREATE TABLE for the relational model
  - modification of data structure e.g. ALTER TABLE
  - deletion of data structure e.g. DROP TABLE
  - creation of indexes on particular data items for data access e.g. CREATE INDEX

Data Manipulation - Functionality of DML

- Basic functions include:
  - retrieval of data e.g. SELECT operator for the relational model
  - modification of data e.g. UPDATE operator
  - deletion of data e.g. DELETE operator
  - creation of new data e.g. INSERT operator
- Most DML's have built in functions e.g. SUM, COUNT, AVG etc

The System Catalog (Data Dictionary)

- A system catalog contains the description of the database
- Often referred to as the 'data dictionary'
- For the relational data model the catalog is defined and interrogated using the same languages that can be used at the application level (i.e. DDL and DML) - but includes extra commands
- The catalog describes:
  - the basic make up of the data items
  - base data
  - views
  - authorisation data
  - some integrity constraints

Views

- A view represents virtual data structures which are derived from base data structures

Title	Author	Publisher	Year
Intro to DB Systems	Date	Addison-Wesley	1986
Fund. of DB Systems	Elmasri	Addison-Wesley	1989
London Fields	Amis	Penguin	1989
100 years of solitude	Marquez	Picador	1982
The history man	Bradbury	Arrow Books	1977

```
CREATE VIEW NEW_BOOK
  AS SELECT TITLE, AUTHOR
    FROM BOOK
    WHERE YEAR>1988
```

at run-time

Title	Author
Fund. of DB Systems	Elmasri
London Fields	Amis

- Views permit users to perceive the database in terms of derived data which correspond exactly to their applications
- Views insulate programmers from base data and enable changes to definitions of base data without affecting aplication programs

Requirements for Database Consistency

Recovery

System failures, either of hardware or software, must not result in an inconsistent database.

A transaction must execute in its entirety or not at all.

Concurrency Control

- The simultaneous execution of many different application programs must be such that each transaction does not interfere with another transaction.

- *The concurrent execution of transactions must be such that each transaction appears to execute in isolation.*

ACID properties

- ACID properties are an important concept for databases. The acronym stands for Atomicity, Consistency, Isolation, and Durability.

- In the context of databases, a single logical operation on the data is called a transaction. An example of a transaction is a transfer of funds from one account to another, even though it might consist of multiple individual operations (such as debiting one account and crediting another). The ACID properties guarantee that such transactions are processed reliably.

ACID properties

- **Atomicity refers to the ability of the DBMS to guarantee that either all of the tasks of a transaction are performed or none of them are. The transfer of funds can be completed or it can fail for a multitude of reasons, but atomicity guarantees that one account won't be debited if the other is not credited as well.**

- **Consistency refers to the database being in a legal state when the transaction begins and when it ends. This means that a transaction can't break the rules, or *integrity constraints*, of the database. If an integrity constraint states that all accounts must have a positive balance, then any transaction violating this rule will be aborted.**

- **Isolation refers to the ability of the application to make operations in a transaction appear isolated from all other operations. This means that no operation outside the transaction can ever see the data in an intermediate state;**

- **Durability refers to the guarantee that once the user has been notified of success, the transaction will persist, and not be undone.**