Database Management System

Lecture 02

Professor Dr. Md. Rakib Hassan
Department of Computer Science and Mathematics,
Bangladesh Agricultural University,
Mymensingh-2202.
Email: rakib@bau.edu.bd

Levels of Abstraction

- **View level:**
  - Highest level of abstraction describes only part of the entire database
  - Several views of the database are defined, and database users see these views.
  - Views can also hide information (such as an employee's salary) for security purposes.

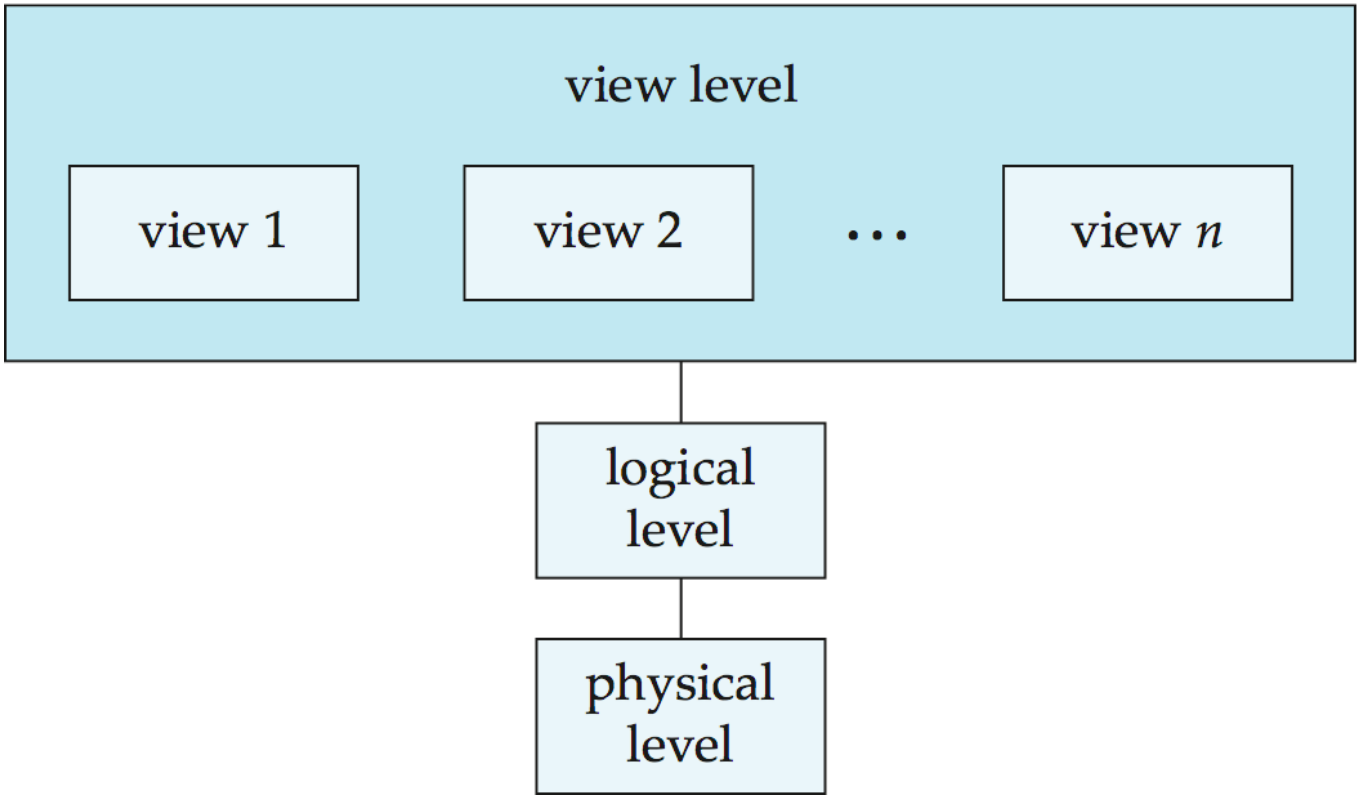
- **Logical level:**
  - Describes what data are stored in the database, and what relationships exist among the data.
  - Example:

```
type instructor = record
                    ID : string;
                    name : string;
                    dept_name : string;
                    salary : integer;
                  end;
```

- **Physical level:**
  - Lowest level of abstraction
  - Describes *how* the data are actually stored
  - Describes complex low-level data structures in detail

Views of Data

- The three levels of data abstraction



Views of Data

- A major purpose of a database system is to provide users with an *abstract view* of the data.
  - That is, the system hides certain details of how the data are stored and maintained.

Database Languages

- Data-Definition Language (DDL):
  - to specify the database schema
- Data-Manipulation Language (DML):
  - to express database queries and updates
- In practice, the DDL and DML are not two separate languages
  - instead they simply form parts of a single database language, such as the widely used SQL (Structured Query Language)

Data Manipulation Language (DML) (Cont.)

- Query
  - Statement requesting the retrieval of information
- Query language
  - The portion of a DML that involves information retrieval
- SQL is the most widely used query language

Data Definition Language (DDL)

- Specification notation for defining the database schema
  - Example: create table instructor (
    ID char(5),
    name varchar(20),
    dept_name varchar(20))

- DDL compiler generates a set of table templates stored in a data dictionary

- Data dictionary contains metadata (i.e., data about data)
  - Database schema
  - Integrity constraints
  - Authorization

DDL (Cont.)

- Integrity constraints specified by DDL:
  - Domain constraints (e.g., integer types, character types)
    - Most elementary
  - Referential integrity
    - Ensures that a value that appears in one relation for a given set of attributes also appears in a certain set of attributes in another relation
  - Assertions
    - An assertion is any condition that the database must always satisfy.
      - Domain constraints and referential-integrity constraints are special forms of assertions.
    - Other assertion:
      - Every department must have at least five courses offered every semester
    - When an assertion is created, the system tests it for validity.
    - If the assertion is valid, then any future modification to the database is allowed only if it does not cause that assertion to be violated.
- Authorization
  - Example: read, write, update

Data Models

- Data Model:
  - Data model is the underlying structure of a database.
  - It is a collection of conceptual tools for describing
    - Data
    - Data relationships
    - Data semantics (meaning)
    - Consistency constraints

- Categories of data models:
  - Relational model
  - Entity-Relationship model (mainly for database design)
  - Object-based data models (Object-oriented and Object-relational)
  - Semi-structured data model (XML)

- Other older models:
  - Network model
  - Hierarchical model

Relational Model

- The relational model uses a collection of tables to represent both data and the relationships among those data.
- Each table has multiple columns, and each column has a unique name.
- Tables are also known as relations.
- The relational data model is the most widely used data model, and a vast majority of current database systems are based on the relational model.

Relational Database

- A relational database consists of a collection of tables, each of which is assigned a unique name

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

Figure: Instructor table

Relational Database (Cont.)

- In general, a row in a table represents a relationship among a set of values.
- A table is a collection of these relationships,

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

Figure: Instructor table

Relational Database (Cont.)

- Relation
  - In the relational model the term relation is used to refer to a table
- Tuple
  - refers to a row.
- Attribute
  - refers to a column of a table.

Domain of an attribute

- Domain of an attribute:
  - For each attribute of a relation, there is a set of permitted values, called the domain of that attribute.

- Examples:
  - the domain of the salary attribute of the instructor relation is the set of all possible salary values
  - the domain of the name attribute is the set of all possible instructor names.

Atomic domain

- A domain is atomic if elements of the domain are considered to be indivisible units.
  - Example:
    - Domain of phone number would not be atomic, since an element of the domain is a set of phone numbers

- Suppose now that the phone number attribute stores a single phone number.
  - If we treat each phone number as a single indivisible unit, then the attribute phone number would have an atomic domain.

Null value

- The null value is a special value that signifies that the value is unknown or does not exist.

- Example:
  - Phone number may not exist for a user.

Instances and Schemas

- Instances:
  - Databases change over time as information is inserted and deleted.
  - The collection of information stored in the database at a particular moment is called an ***instance*** of the database.

- Schema:
  - The overall design of the database is called the database ***schema***.
  - Schemas are changed infrequently, if at all.

Schema and Instance

- Schema:
  - section (course_id, sec_id, semester, year, building, room_number, time_slot_id)

- Relation instance:
  - refers to a specific instance of a relation, i.e., containing a specific set of rows.
  - Sample instance of this relation:

course_id	*sec_id*	*semester*	*year*	*building*	*room_number*	*time_slot_id*
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C

Superkey

- No two tuples in a relation are allowed to have exactly the same value for all attributes.

- A superkey is a set of one or more attributes that, taken collectively, allow us to identify uniquely a tuple in the relation.
  - For example, the ID attribute of the relation instructor is sufficient to distinguish one instructor tuple from another.
    - Thus, ID is a superkey.
  - The name attribute of instructor, on the other hand, is not a superkey,
    - because several instructors might have the same name.

Candidate key

- A superkey may contain extra attributes.
  - For example, the combination of ID and name is a superkey for the relation instructor.
- A candidate key of a relation is a minimal superkey for that relation

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

Fig. Instructor relation

Candidate key (Cont.)

- Several distinct sets of attributes could also serve as a candidate key
  - Suppose that a combination of name and dept_name is sufficient to distinguish among members of the instructor relation.
  - Then, both {ID} and {name, dept name} are candidate keys.
  - Although the attributes ID and name together can distinguish instructor tuples, their combination, {ID, name}, does not form a candidate key
    - since the attribute ID alone is a candidate key.

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

Fig. Instructor relation

Primary key

- Primary key
  - Candidate key chosen by the database designer as the principal means of identifying tuples within a relation.
- Primary keys must be chosen with care such that its attribute values are never, or very rarely, changed.
  - For instance, the address field of a person should not be part of the primary key, since it is likely to change.
- It is customary to list the primary key attributes of a relation schema before the other attributes
- Primary key attributes are underlined.

Foreign key

Foreign key

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

Fig. Instructor relation

(referencing relation)

Primary key

dept_name	building	budget
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

Fig. Department relation

(referenced relation)

Summary

- Superkey:
  - A superkey of a relation is a set of one or more attributes whose values are guaranteed to identify tuples in the relation uniquely.
- Candidate key:
  - A candidate key is a minimal superkey, that is, a set of attributes that forms a superkey, but none of whose subsets is a superkey.
- Primary key:
  - One of the candidate keys of a relation is chosen as its primary key.
- Foreign key:
  - Foreign keys are primary keys of another table.