Database Management Systems (DBMS)

Lec 8: Entity-Relationship (ER) Model

Ramesh K. Jallu

IIIT Raichur

Date: 16/2/21

Today's plan

- DRC expression
- ER Model

Example queries

Instructor

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

Teaches

ID	*course_id*	*sec_id*	*semester*	*year*
10101	CS-101	1	Fall	2017
10101	CS-315	1	Spring	2018
10101	CS-347	1	Fall	2017
12121	FIN-201	1	Spring	2018
15151	MU-199	1	Spring	2018
22222	PHY-101	1	Fall	2017
32343	HIS-351	1	Spring	2018
45565	CS-101	1	Spring	2018
45565	CS-319	1	Spring	2018
76766	BIO-101	1	Summer	2017
76766	BIO-301	1	Summer	2018
83821	CS-190	1	Spring	2017
83821	CS-190	2	Spring	2017
83821	CS-319	2	Spring	2018
98345	EE-181	1	Spring	2017

Q_1: Find the names of all instructors in the Computer Science department together with the course id of all courses they teach

$$\{< n, c > \mid \exists\, d, s\, (< i, n, d, s > \in \text{Instructor} \wedge d = \text{“Comp. Sci.”}) \wedge \exists\, i, a, se, y\, (< i, c, a, se, y > \in \text{Teaches})\}$$

Example queries

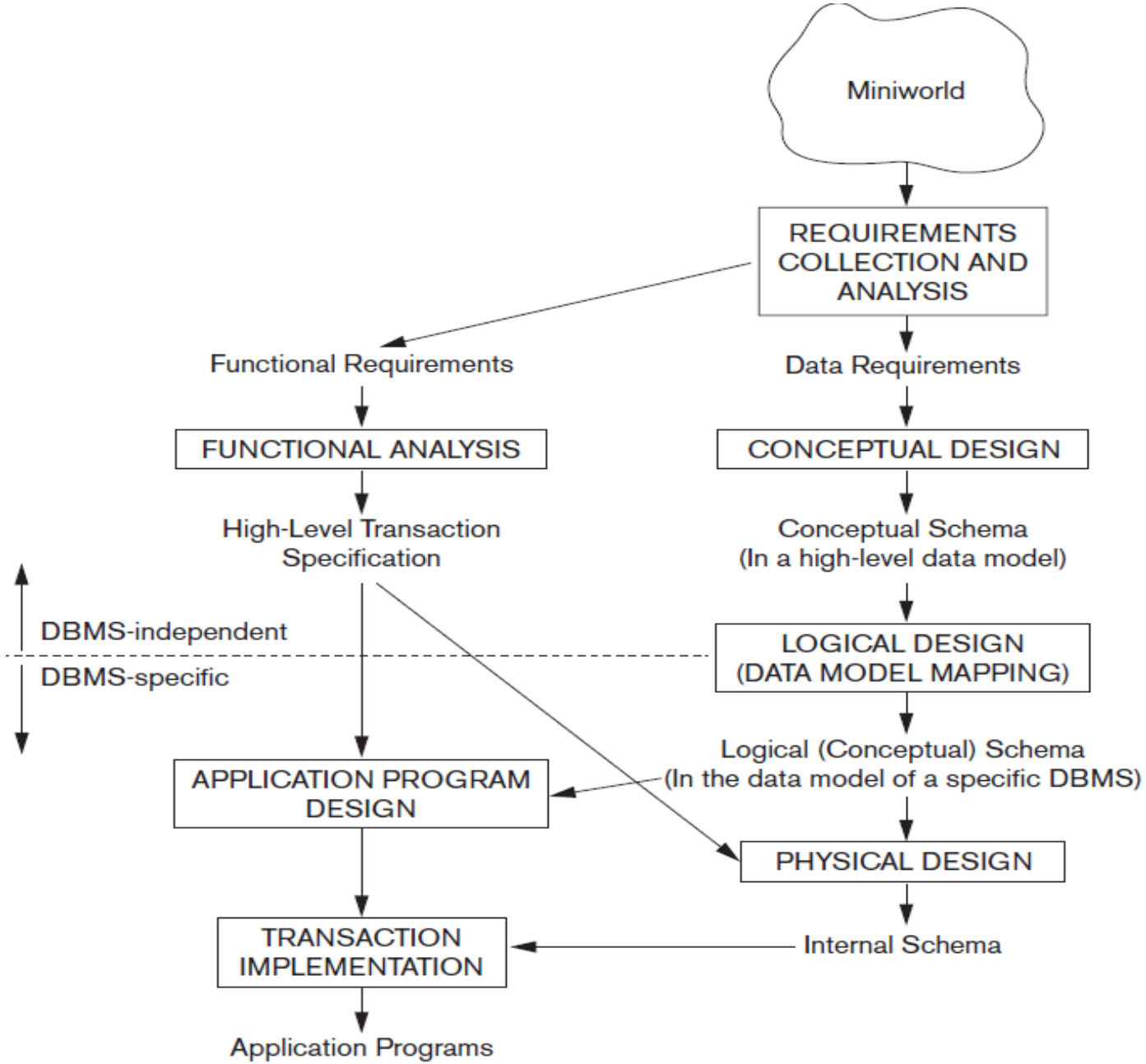
classroom(*building*, *room_number*, *capacity*)
department(*dept_name*, *building*, *budget*)
course(*course_id*, *title*, *dept_name*, *credits*)
instructor(*ID*, *name*, *dept_name*, *salary*)
section(*course_id*, *sec_id*, *semester*, *year*, *building*, *room_number*, *time_slot_id*)
teaches(*ID*, *course_id*, *sec_id*, *semester*, *year*)
student(*ID*, *name*, *dept_name*, *tot_cred*)
takes(*ID*, *course_id*, *sec_id*, *semester*, *year*, *grade*)
advisor(*s_ID*, *i_ID*)
time_slot(*time_slot_id*, *day*, *start_time*, *end_time*)
prereq(*course_id*, *prereq_id*)

Q_2: Find all students who have taken all courses offered in the Computer Science department

$$\{ < i > \mid \exists\, n, d, tc\, (< i, n, d, tc > \in \text{student}) \wedge$$
$$\forall\, ci, ti, dn, cr\, (< ci, ti, dn, cr > \in \text{course} \wedge dn = \text{“Com. Sci.”} \Rightarrow$$
$$\exists\, si, se, y, g\, (< i, ci, si, se, y, g > \in \text{takes}\,))\}$$

Recap: Phases in a database design

1. Requirements collection and analysis
   - This phase involves assessing the informational needs of an organization
2. Conceptual design
   - A description of the structure of database, relationships among data, and rules
3. Logical design
   - Implementation of the database design: creation
4. Physical design
   - Aims to maximize database efficiency by finding ways to speed up the performance
   - Application programs are designed and implemented as transactions


Miniworld
REQUIREMENTS COLLECTION AND ANALYSIS
Functional Requirements
Data Requirements
FUNCTIONAL ANALYSIS
CONCEPTUAL DESIGN
High-Level Transaction Specification
Conceptual Schema (In a high-level data model)
DBMS-independent
DBMS-specific
LOGICAL DESIGN (DATA MODEL MAPPING)
Logical (Conceptual) Schema (In the data model of a specific DBMS)
APPLICATION PROGRAM DESIGN
PHYSICAL DESIGN
TRANSACTION IMPLEMENTATION
Internal Schema
Application Programs

Recap: Data models

- A collection of tools for describing data, data relationships, data semantics, and data constraints
- Categories:

1. Relational model
2. Entity-Relationship model
3. Semi-structured model: flexible format for data exchange
4. Object-based model

The Entity-Relationship (ER) Model

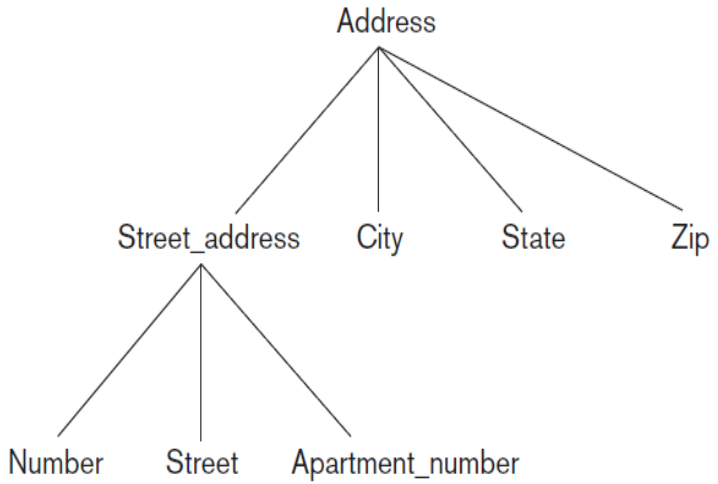
- It is a ***high-level data model*** used to define the data elements and relationship
- It develops a ***conceptual design*** for the database and provides a very simple and easy to design view of data
- The database structure is represented as a diagram called an ***entity-relationship diagram***

The elements in the ER Model

- The ER model describes data as ***entities***, ***relationships***, and ***attributes***
- An ***entity*** is a thing or object in the real world with an independent existence
  - An entity may be an object with a physical existence
    - e.g., a particular person, car, department, employee, etc.
  - Or it may be an object with a conceptual existence
    - e.g., a company, job, university course, grade, etc.
  - Each entity has ***attributes*** – the particular properties that describe the entity
    - e.g., Name, SSN, Age, Address, Salary, etc.

Types of attributes

1. ***Simple*** ***vs*** ***Composite***



2. ***Single-valued*** ***vs*** ***Multi-valued***

   - Single-valued attribute can either be simple or composite

3. ***Stored*** ***vs*** ***Derived***

Entity types, entity sets, keys, and domain

1. ***Entity type:*** a collection of entities that have the same attributes

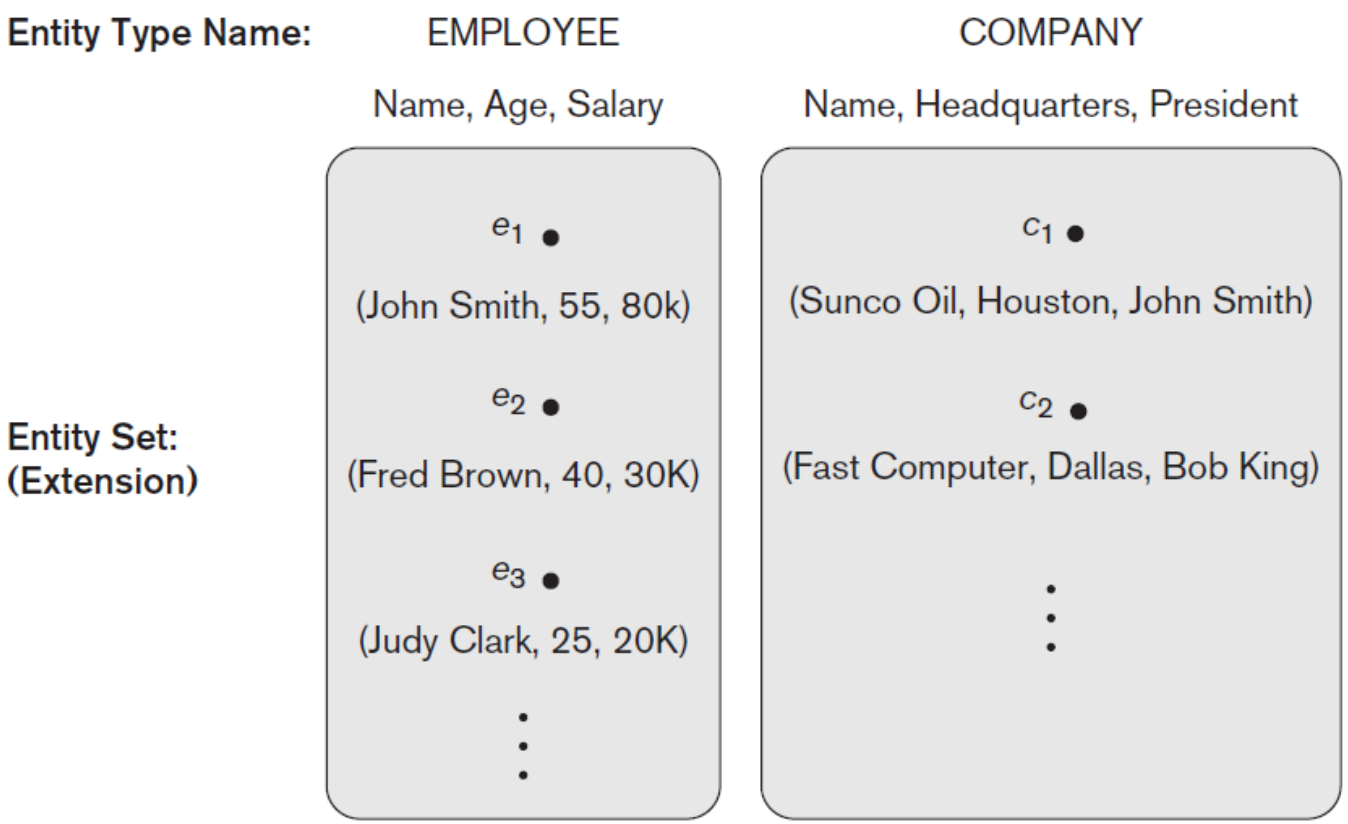
2. ***Entity set:*** the collection of all entities of a particular entity type in the database at any point in time

3. ***Key:*** the set of attributes whose values are distinct for each individual entity in the entity set

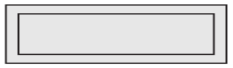
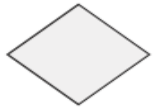
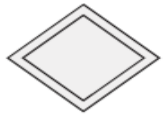
4. ***Weak entity:*** an entity type that do not have a key attribute

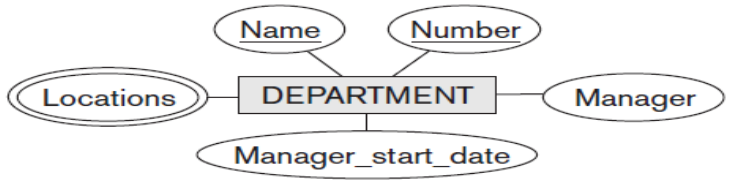
5. ***Value set:*** the set of values that an attribute can have

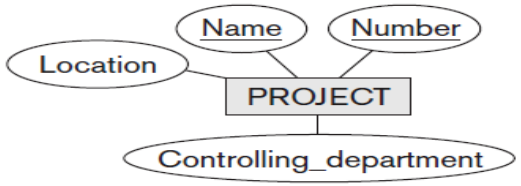
An example

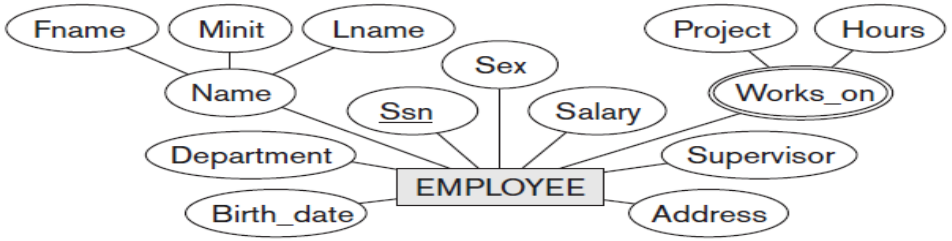

Entity Type Name:
EMPLOYEE
COMPANY
Name, Age, Salary
Name, Headquarters, President
Entity Set:
(Extension)
e1
(John Smith, 55, 80k)
e2
(Fred Brown, 40, 30K)
e3
(Judy Clark, 25, 20K)
c1
(Sunco Oil, Houston, John Smith)
c2
(Fast Computer, Dallas, Bob King)

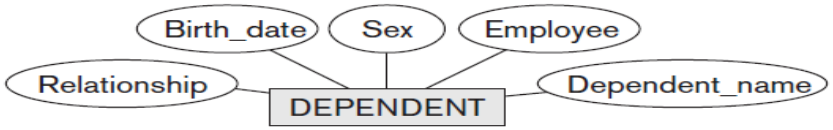
The ER diagram

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Indentifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived Attribute







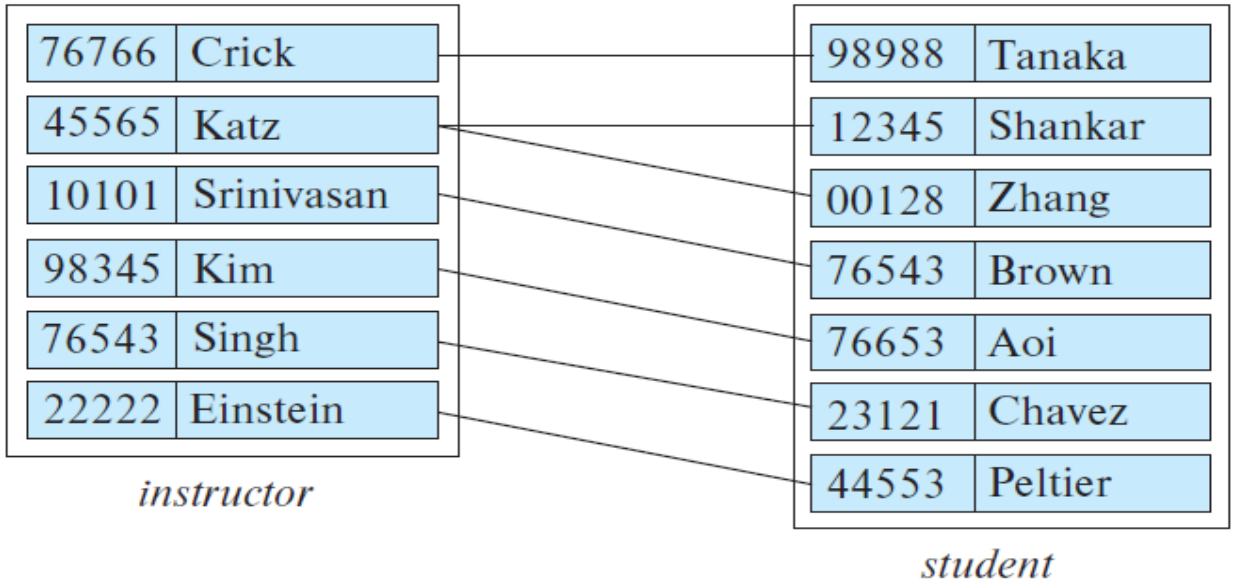


Relationship sets

1. ***Relationship:*** an association among several entity types

2. ***Relationship set:*** a set of relationships of the same type

3. ***Degree***: The number of participating entity types



Thank you!