

CE 223- DATABASE SYSTEMS

WEEK #2 RECITATION QUESTIONS

- 2.2.1
- 2.3.1
- 4.1.1
- 4.1.3
- 4.1.9
- 4.2.1
- 4.2.5
- 4.4.4
- 4.5.1
- 4.6.1

2.2.1

<i>acctNo</i>	<i>type</i>	<i>balance</i>
12345	savings	12000
23456	checking	1000
34567	savings	25

The relation Accounts

<i>firstName</i>	<i>lastName</i>	<i>idNo</i>	<i>account</i>
Robbie	Banks	901-222	12345
Lena	Hand	805-333	12345
Lena	Hand	805-333	23456

The relation Customers

2.2.1

Exercise 2.2.1: In Fig. 2.6 are instances of two relations that might constitute part of a banking database. Indicate the following:

- a) The attributes of each relation.
- b) The tuples of each relation.
- c) The components of one tuple from each relation.
- d) The relation schema for each relation.
- e) The database schema.
- f) A suitable domain for each attribute.
- g) Another equivalent way to present each relation.

2.3.1

Exercise 2.3.1: In this exercise we introduce one of our running examples of a relational database schema. The database schema consists of four relations, whose schemas are:

```
Product(maker, model, type)
PC(model, speed, ram, hd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)
```

The **Product** relation gives the manufacturer, model number and type (PC, laptop, or printer) of various products. We assume for convenience that model numbers are unique over all manufacturers and product types; that assumption is not realistic, and a real database would include a code for the manufacturer as part of the model number. The **PC** relation gives for each model number that is a PC the speed (of the processor, in gigahertz), the amount of RAM (in megabytes), the size of the hard disk (in gigabytes), and the price. The **Laptop** relation is similar, except that the screen size (in inches) is also included. The **Printer** relation records for each printer model whether the printer produces color output (true, if so), the process type (laser or ink-jet, typically), and the price.

2.3.1

Write the following declarations:

- a) A suitable schema for relation **Product**.
- b) A suitable schema for relation **PC**.
- c) A suitable schema for relation **Laptop**.
- d) A suitable schema for relation **Printer**.
- e) An alteration to your **Printer** schema from (d) to delete the attribute **color**.
- f) An alteration to your **Laptop** schema from (c) to add the attribute **od** (optical-disk type, e.g., cd or dvd). Let the default value for this attribute be 'none' if the laptop does not have an optical disk.

11.1.1

11.1. SEMISTRUCTURED DATA

485

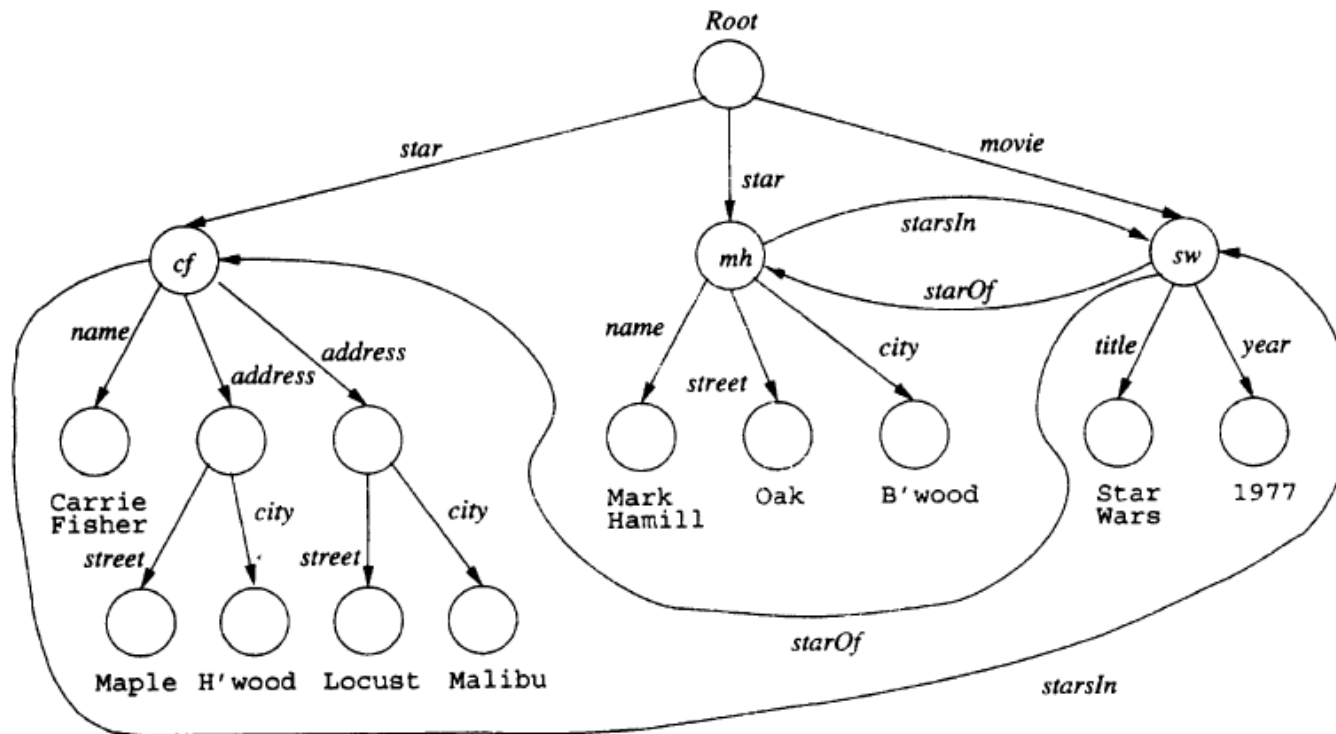


Figure 11.1: Semistructured data representing a movie and stars

11.1.1

Exercise 11.1.1: Since there is no schema to design in the semistructured-data model, we cannot ask you to design schemas to describe different situations. Rather, in the following exercises we shall ask you to suggest how particular data might be organized to reflect certain facts.

- a) Add to Fig. 11.1 the facts that *Star Wars* was directed by George Lucas and produced by Gary Kurtz.
- b) Add to Fig. 11.1 information about *Empire Strikes Back* and *Return of the Jedi*, including the facts that Carrie Fisher and Mark Hamill appeared in these movies.
- c) Add to (b) information about the studio (Fox) for these movies and the address of the studio (Hollywood).

4.1.1 (pg 138)

Design a database for a bank, including information about customers and their accounts. Information about a customer includes their name, address, phone, and Social Security number. Accounts have numbers, types (e.g., savings, checking) and balances. Also record the customer(s) who own an account.

Draw the E/R diagram for this database. Be sure to include arrows where appropriate, to indicate the multiplicity of a relationship.

4.1.3 (pg 139)

Give an E/R diagram for a database recording information about teams, players, and their fans, including:

1. For each team, its name, its players, its team captain (one of its players), and the colors of its uniform.
2. For each player, his/her name.
3. For each fan, his/her name, favorite teams, favorite players, and favorite color.

Remember that a set of colors is not a suitable attribute type for teams. How can you get around this restriction?

4.1.9 (pg 140)

Design a database suitable for a university registrar. This database should include information about students, departments, professors, courses, which students are enrolled in which courses, which professors are teaching which courses, student grades, TA's for a course (TA's are students), which courses a department offers, and any other information you deem appropriate.

Note that this question is more free-form than the questions above, and you need to make some decisions about multiplicities of relationships, appropriate types, and even what information needs to be represented.

4.2.1 (pg 145)

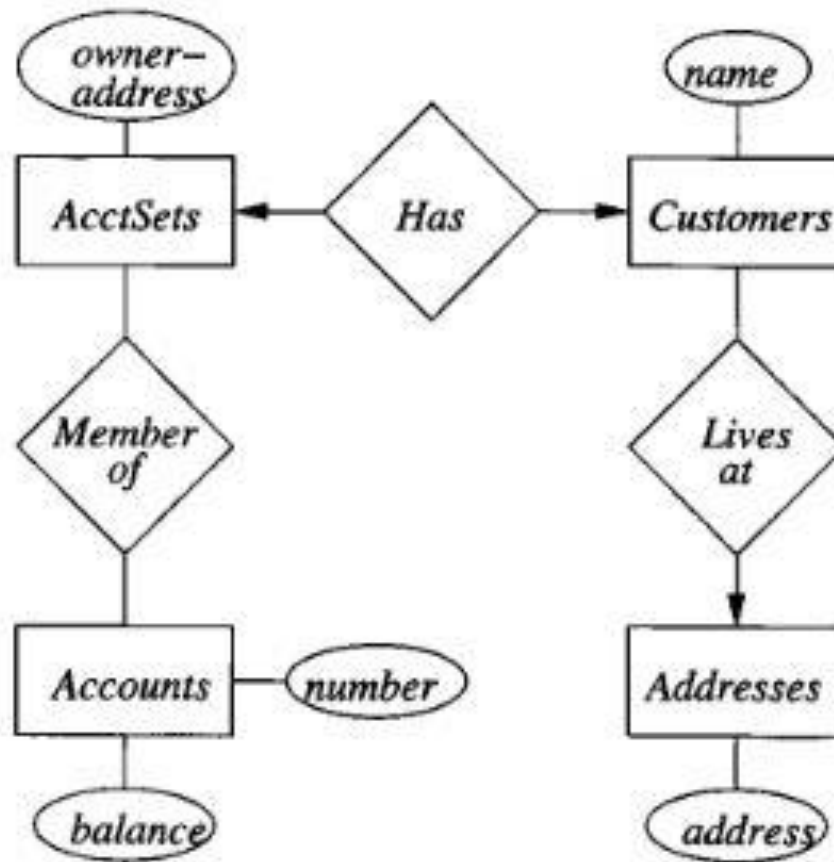


Figure 4.14: A poor design for a bank database

4.2.1 (pg 145 con't)

In Fig. 4.14 is an E/R diagram for a bank database involving customers and accounts. Since customers may have several accounts, and accounts may be held jointly by several customers, we associate with each customer an "account set," and accounts are members of one or more account sets. Assuming the meaning of the various relationships and attributes are as expected given their names, criticize the design. What design rules are violated? Why? What modifications would you suggest?

4.2.5 (pg 147)

In this and following exercises we shall consider two design options in the E/R model for describing births. At a birth, there is one baby (twins would be represented by two births), one mother, any number of nurses, and any number of doctors. Suppose, therefore, that we have entity sets Babies, Mothers, Nurses, and Doctors. Suppose we also use a relationship Births, which connects these four entity sets, as suggested in Fig. 4.15. Note that a tuple of the relationship set for Births has the form (baby, mother, nurse, doctor). If there is more than one nurse and/or doctor attending a birth, then there will be several tuples with the same baby and mother, one for each combination of nurse and doctor.

4.2.5 (pg 147)

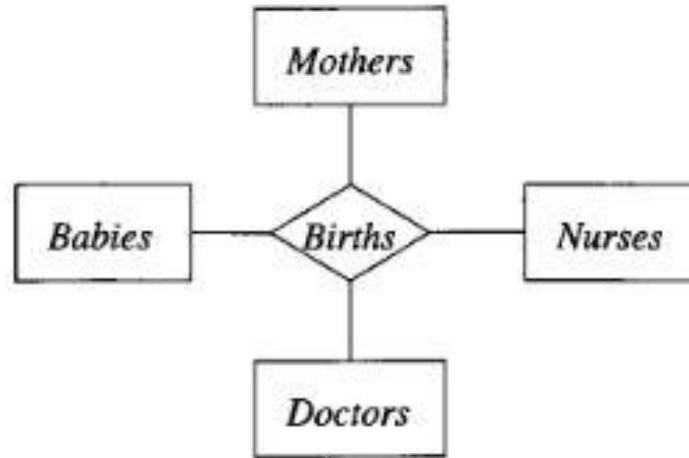


Figure 4.15: Representing births by a multiway relationship

There are certain assumptions that we might wish to incorporate into our design. For each, tell how to add arrows or other elements to the E/R diagram in order to express the assumption.

- a) For every baby, there is a unique mother.
- b) For every combination of a baby, nurse, and doctor, there is a unique mother.
- c) For every combination of a baby and a mother there is a unique doctor.

4.4.4 (pg 156)

Draw E/R diagrams for the following situations involving weak entity sets. In each case indicate keys for entity sets.

a) Entity sets Courses and Departments. A course is given by a unique department, but its only attribute is its number. Different departments can offer courses with the same number. Each department has a unique name.

4.4.4 (pg 156)

b) Entity sets Leagues, Teams, and Players. League names are unique. No league has two teams with the same name. No team has two players with the same number. However, there can be players with the same number on different teams, and there can be teams with the same name in different Leagues.

4.5.1 (pg 163)

Convert the E/R diagram of Fig. 4.29 to a relational database schema.

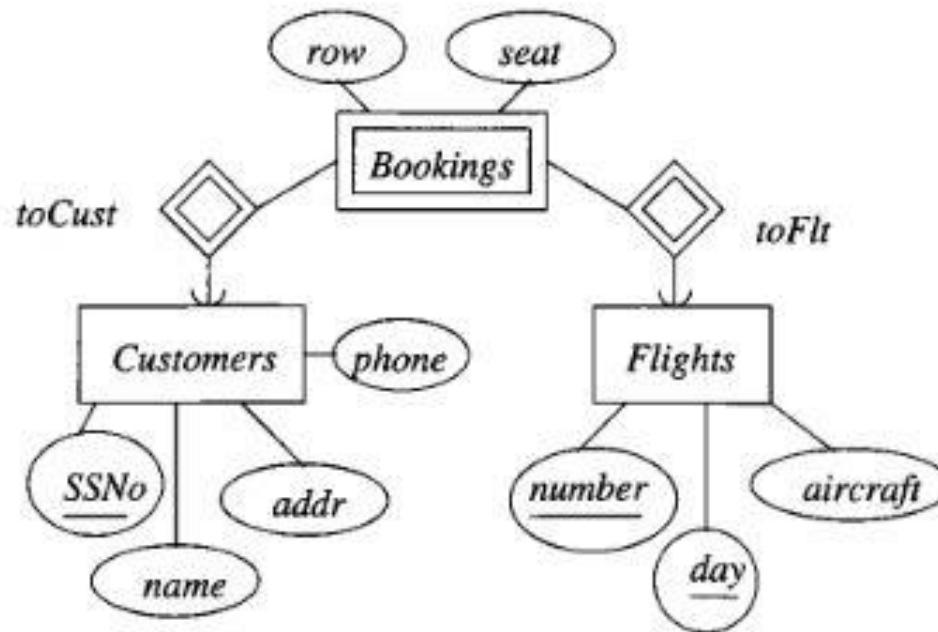


Figure 4.29: An E/R diagram about airlines

4.5.2 (pg 163)

There is another E/R diagram that could describe the weak entity set Bookings in Fig. 4.29. Notice that a booking can be identified uniquely by the flight number, day of the flight, the row, and the seat; the customer is not then necessary to help identify the booking.

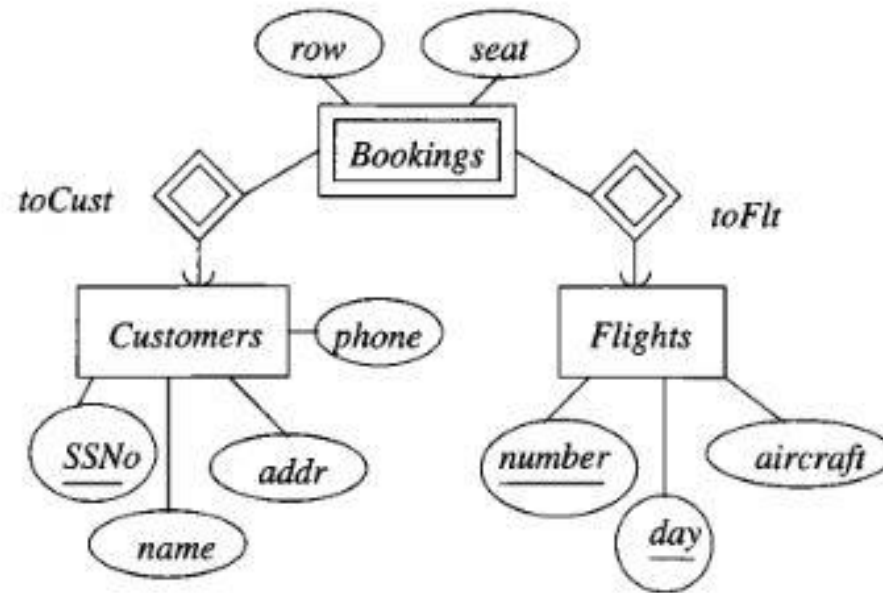


Figure 4.29: An E/R diagram about airlines

4.5.2 (pg 163)

- a) Revise the diagram of Fig. 4.29 to reflect this new viewpoint.
- b) Convert your diagram from (a) into relations. Do you get the same database schema as in Exercise 4.5.1?

4.6.1 (pg 171)

Convert the E/R diagram of Fig. 4.32 to a relational database schema, using each of the following approaches:

- a) The straight-E/R method.
- b) The object-oriented method.
- c) The nulls method.

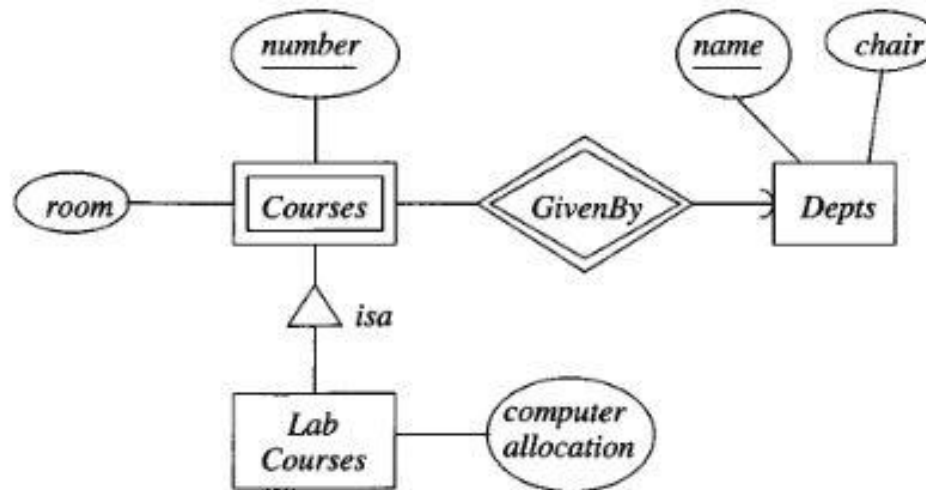


Figure 4.32: E/R diagram for Exercise 4.6.1