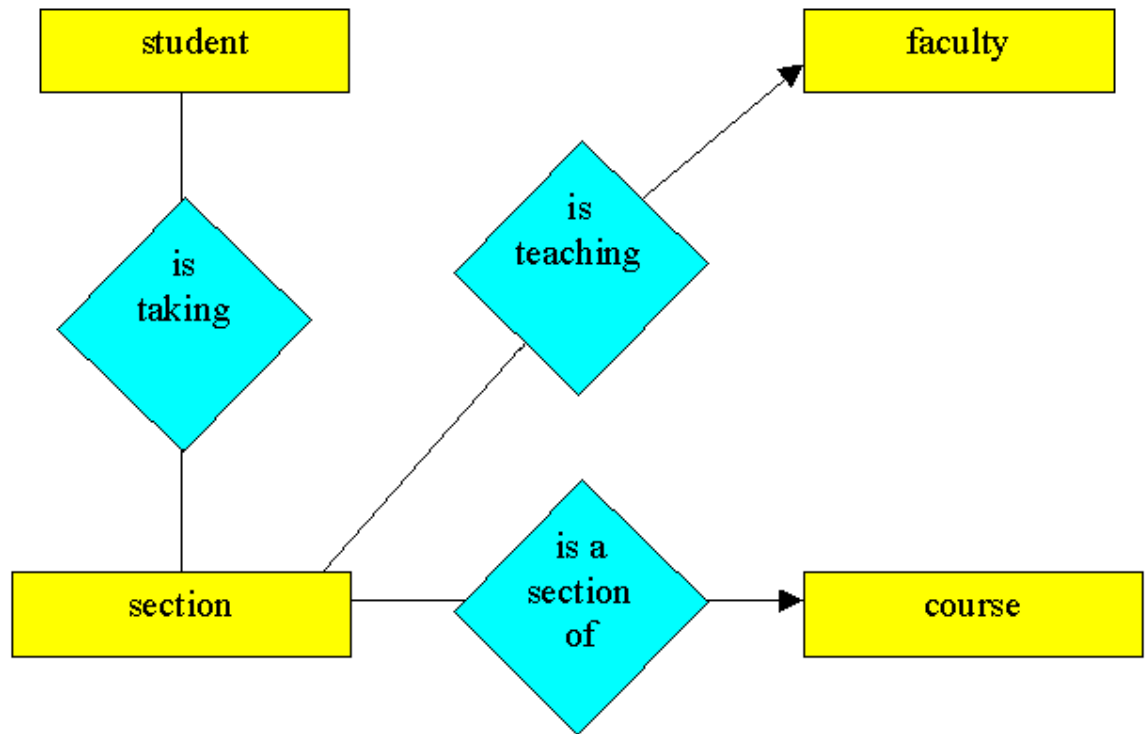# Converting an E-R diagram to a relational schema

Below describes a mechanical procedure for converting an E-R diagram to a relational schema. We will use the student-section-course database as an example:

1. (non-weak) Entity sets.

Create a relation for each entity set. It can use the same name and same set of attributes as the entity set. The key of the entity set is used as the primary key of the relation. For example,

Student(ssn, last, first, streetaddress, city, state, country, zip)
Course(dept, number, title, description, credits)
Section(crn, days, time, room)
Faculty(ssn, last, first, title, dept, office, phone)

2. Relationships

a. Binary many-to-many relationships

For each such relationship, create a relation. The attributes of this relation are the attributes of the relationship, together with the primary keys of the two related entity sets. The primary key of the relation combines the primary keys of the related entity sets.

In the example database, the "is taking" relationship would be converted into

Enrolled(ssn, crn, gradeOption, credits)

In the case of a recursive many-to-many relationship, it is necessary to rename the primary keys borrowed from the related entity sets. For example, if we want to implement the "is a prerequisite of" relationship between courses, we would create a relation as follows:

Prerequisite(prereqDept, prereqNumber, successorDept, successorNumber)

b. Binary one-to-many relationships

There are two ways to handle this type of relationship. One is to follow essentially the same approach as with the many-to-many relationship, by creating a new relation to represent the relationship. For example, for the "is teaching" relationship, we would create a relation

Teaches(crn, ssn)

Note that the primary key of the relation is just the primary key of the section, not the combination of both primary keys.

An alternative is to simply include the primary key of the "one" entity set in the relation representing the "many" entity set. In this case, we would simply add the instructor ssn to the section relation:

Section(crn, days, time, room, instructorSsn)

What if the one-to-many relationship has attributes?

Which method is preferable?

c. Binary one-to-one relationships

As with the one-to-many relationships, they can be implemented by adding a new relation or by including the primary key of one entity set in the relation representing the other one.

d. n-ary relationships

Create a relation representing the relationship. Include the primary key of each participating entity set. Any attributes of the relationship become attributes of the relation. For example, if students, majors, and faculty advisors are related by a ternary "advises in" relationship, we would create a relation:

Advises(studentSsn, advisorSsn, major)

What is the primary key of the resulting relation?

3. Weak entity sets

The procedure described above for representing entity sets does not work for weak entity sets, because they do not contain their own primary keys as attributes. Some allowance must be made to handle this situation.

The solution is simply to copy the needed key attributes from related entities into the relation representing the weak entity. For example, if departments are modeled as entities, and the department code is part of the key of courses, then course becomes a weak entity. Department code must be included as an attribute of course, so it can be used as part of the primary key.

Exercise 4.5.1: Convert the E/R diagram of Fig. 4.29 to a relational database schema.
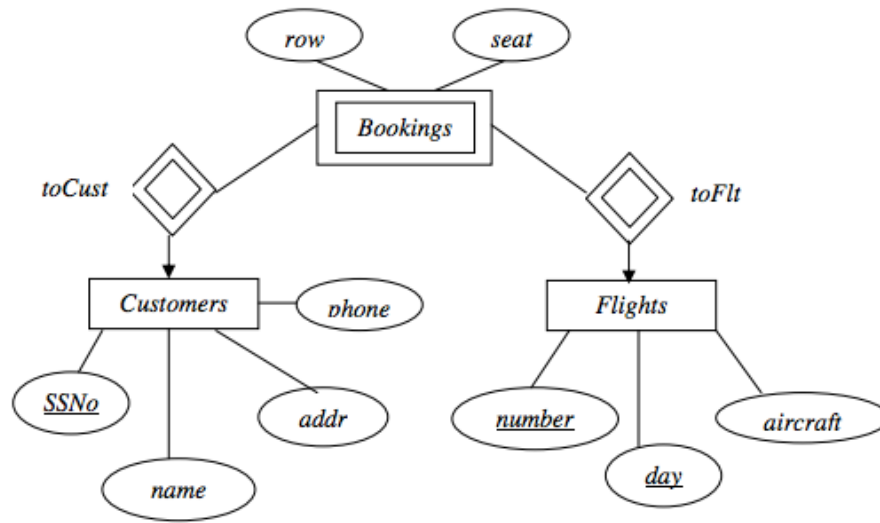

Figure 4.29: An E/R diagram about airlines

Customers(<u>SSN</u>,name,addr,phone);

Flights(<u>number,day</u>,aircraft)

Bookings(<u>SSN, number,day</u>,row,seat)

~~toCust(<u>SSN, number,day</u>)~~
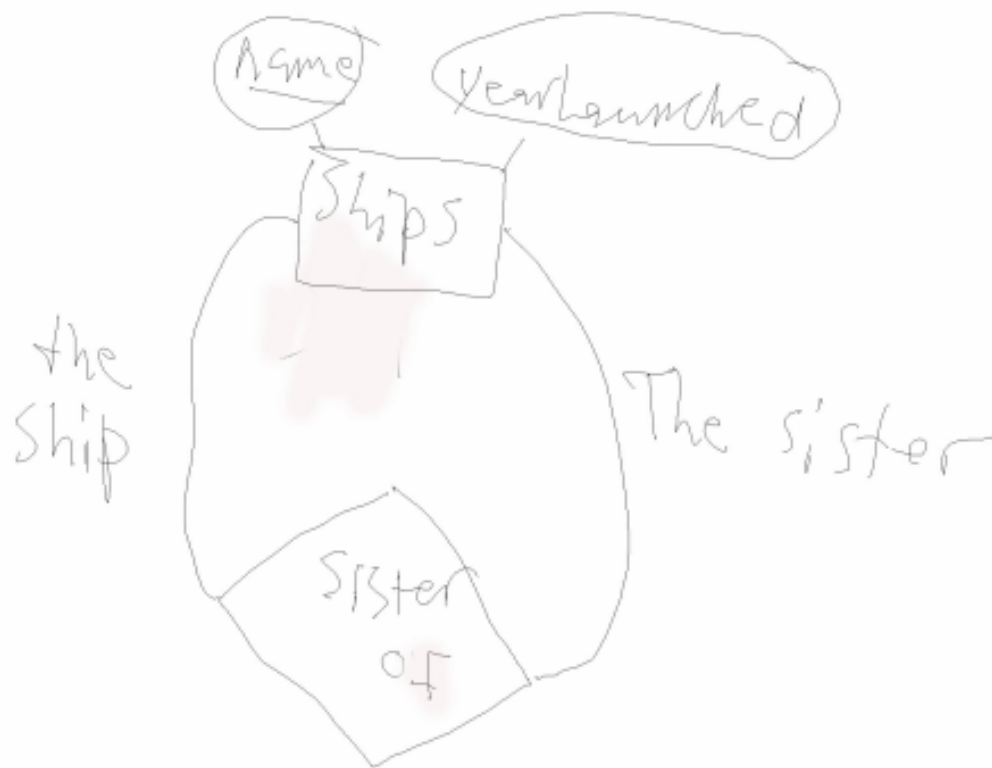
~~toFlt(<u>SSN, number,day</u>)~~

4.5.3

Figure 4: An E/R diagram about sister ships.

Ships(name,yearLaunched);

SistersOf(name,sisterName).

Exercise 4.5.4 a)

Exercise 4.5.4: Convert the following E / R diagrams to relational database schemas.
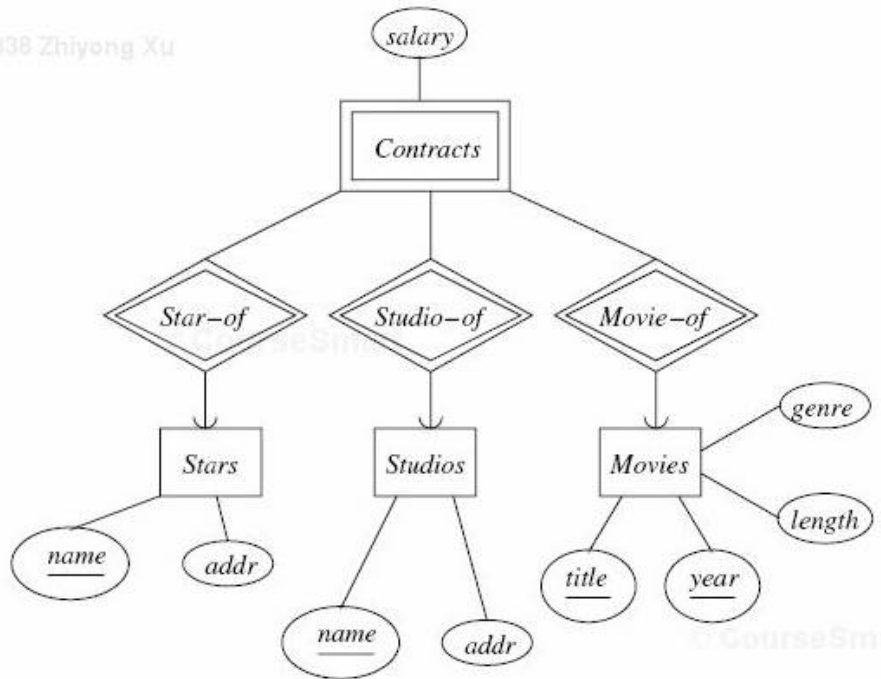a) Figure 4.22.

Figure 4.22: Connecting entity sets are weak

Stars(<u>name</u>,addr);

Studios(<u>name</u>,addr);

Movies(<u>title</u>,<u>year</u>,length,genre);

Contracts(<u>StartName</u>,<u>studiosName</u>,<u>titile</u>,<u>year</u>,salary).