

E/R Methodology

Tom Kelliher, CS 318

Feb. 6, 2002

1 Administrivia

Announcements

Assignment

Read 5.5–7.

From Last Time

PostgreSQL lab, SQL DDL.

Outline

1. Introduction.
2. Entities.
3. Relationships.
4. Participation constraints.

Coming Up

E/R example and limitations.

2 E/R Methodology Introduction

1. Design methodology, *not* a model — distinct from relational data model.
2. Design begins from a specification. In our case, the project description and application specification.

3. Entities — objects:

(a) Courses.

(b) Faculty.

(c) Students.

Have properties — attributes.

4. Relationships — associations between entities:

(a) Student **takes** course.

(b) Faculty **teaches** course.

(c) Student **takes** course **taught by** faculty.

(d) Employee **reports to** employee.

Relationships can also have properties: how long a faculty member has taught a course.

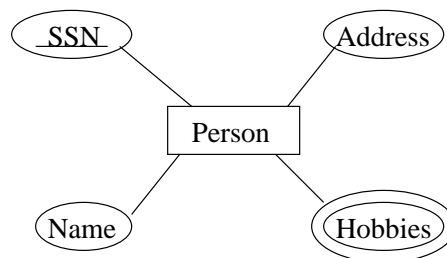
5. Constraints: A student can't take two courses offered at the same time.

3 Entities

Steps in the design process:

1. Select entities. Specific objects: butter pecan, vanilla, bittersweet mint, DQ, Moxleys, PSU Creamery, Ben & Jerry's.
2. Determine entity types (groups of semantically related entities):
 - (a) Ice cream.
 - (b) Ice cream makers.
3. Determine attributes and domains — essential properties and their types.
 - (a) Ice cream maker: ID, name, address, phone, hot line, etc.
 - (b) Attributes can be *set-valued*: hobbies, children, courses.

These are not supported in the relational model. Reconciliation?
4. Choose key. Similar to candidate key, but watch out for set-valued attributes.
5. Derive schema. Similar to relation model, but graphical:



Key features:

- (a) Shapes.
- (b) Key indicator.
- (c) Set-valued attribute indicator.

6. Represent in relation model.

Implications of set-valued attributes:

- (a) “Same” tuple occurs multiple times.
- (b) Set-valued attributes must be included in key.

```
CREATE TABLE Person (  
    SSN            INTEGER,  
    Name           CHAR(20),  
    Address        CHAR(50),  
    Hobby          (CHAR(10),  
    PRIMARY KEY (SSN, Hobby) );
```

4 Relationships

Steps in the design process:

1. Determine what relationships are needed.

Identify the related entities: Person **likes** ice cream.

2. Define roles: How each entity enters into the relationship.

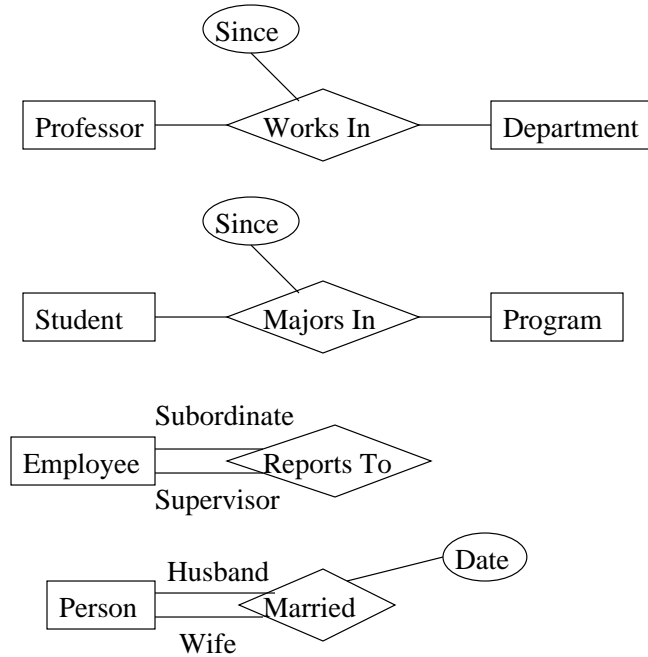
Usually obvious, but what about: Employee **reports to** employee? Subordinate, supervisor roles.

3. Define any attributes specific to the relationship. Examples:

- (a) Start date of employment.
- (b) Date of major declaration.
- (c) Affinity for ice cream flavor.

4. Design the schema:

- (a) Attributes: Choose domain, decide scalar or set-valued.
 - (b) Roles: Derive entity types.
 - (c) Determine constraints.
5. Produce relationship diagram. Examples:



Features:

- (a) Shapes.
- (b) Role indications.
- (c) Key indicators: arrow, set.

What are the keys?

Key:

- (a) All relationship instances within the relationship type must be unique on key (obvious).

(b) Minimal subset of roles, attributes.

(c) Again, not exactly the same as the candidate key.

6. Representation in relational model:

(a) Attributes: carry over directly.

(b) Roles: Include the primary key of each *entity type* as attributes.

Not the primary key of the relation!

(c) Candidate keys: Use keys of relationship. Dealing with set-valued attributes.

(d) Foreign keys: Attributes corresponding to roles become foreign keys referring back to relation created from the entity type.

(e) Examples:

```
CREATE TABLE WorksIn (  
    Since          DATE,  
    ProfId         INTEGER,  
    DeptId         CHAR(4),  
    PRIMARY KEY (ProfId),  
    FOREIGN KEY (ProfId) REFERENCES Professor (Id),  
    FOREIGN KEY (DeptId) REFERENCES Department );
```

```
CREATE TABLE Married (  
    Date          DATE,  
    SSNHusband   INTEGER,  
    SSNWife       INTEGER,  
    PRIMARY KEY (SSNHusband),  
    UNIQUE (SSNWife),  
    -- Assume we dealt with Hobby so that SSN is now  
    -- primary key in person.  
    FOREIGN KEY (SSNHusband) REFERENCES Person (SSN),  
    FOREIGN KEY (SSNWife) REFERENCES Person (SSN) );
```

7. IsA relationships and entity type hierarchies.

(a) Student **IsA** Person.

(b) Professor **IsA** Employee **Person**.

Use inheritance from object design.

5 Participation Constraints

1. How do you guarantee that an entity participates in a particular relationship?

Examples:

(a) A professor is a member of some department.

(b) A student is taking courses this semester.

2. Implementation: define a foreign key from the entity type relation to the relationship relation.

Or, use a trigger (don't have the keys we'd like — transcript relation in student example). Performance tradeoffs.