CIS 560 – Database System Concepts

Lecture 8

E/R Diagrams

September 13, 2013

Credits for slides: Suciu, Chang, Ullman.

Outline

Last time:

- DB design: E/R Diagrams (Sections 4.1-4.5)

Today:

- DB design: E/R Diagrams (Sections 4.1-4.5)
  - HW1 graded, key posted online
  - HW2 due by midnight
  - HW3 will be posted

Next:

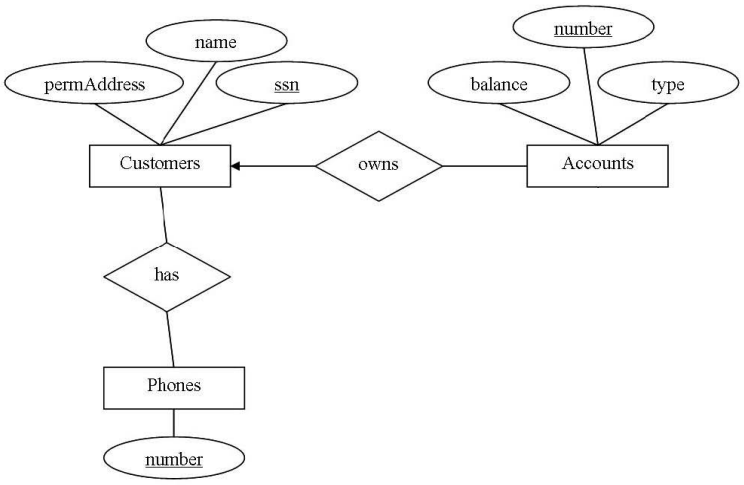
- DB design: Functional Dependencies (Sections 3.1-3.2)
- DB design: Normal Forms (Sections 3.3-3.4)

2

Review

Draw an ER diagram to design a database for a banking system, which maintains information about customers and their accounts with the following assumptions:

- Each customer has a name, a permanent address, and a social security number.
- Each customer can have multiple phone numbers, and the same phone number may be shared by multiple customers.
- A customer can own multiple accounts, but each account is owned by a single customer.
- Each account has an account number, a type (such as saving, checking, etc.), and a balance.



4

FROM E/R to Relations?

Translate your E/R diagram into a set of relations. Select approaches that yield the smallest number of relations; merge relations where appropriate.

Customers(ssn, name, permAddress)

Accounts(number, type, balance, ssn)

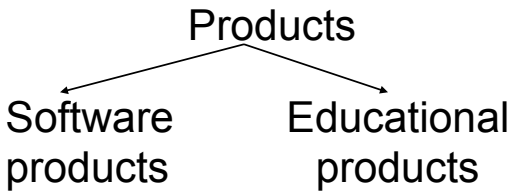
- Accounts.ssn is a foreign key that references Customers.ssn
- Accounts.ssn is also a key for Accounts.

Pnones(number)

Has(ssn, number)

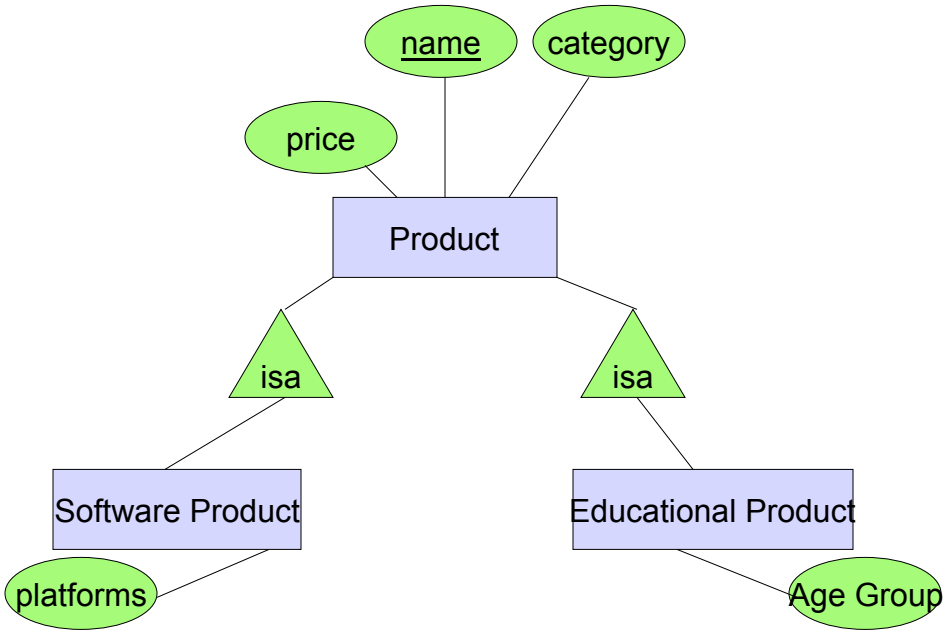
- Has.ssn is a foreign key that references Customers.ssn

Modeling Subclasses



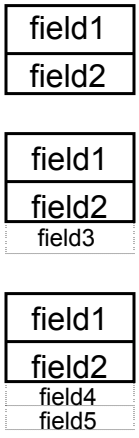
6

Subclasses



Understanding Subclasses

- Think in terms of records:
  - Product
  - SoftwareProduct
  - EducationalProduct



8

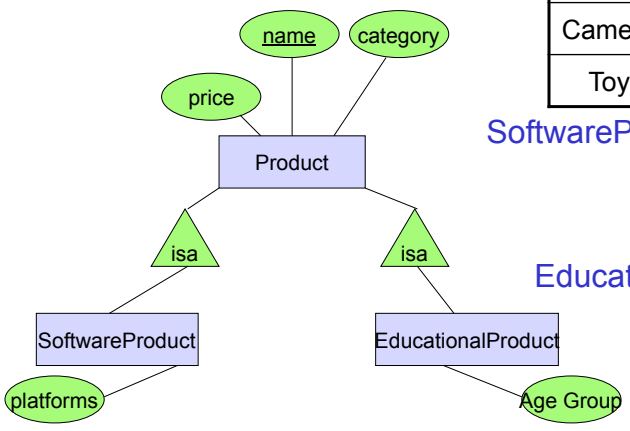
Translating Subclass Entities: The Rules

Three approaches:

1. *E/R style*: create one relation for each subclass, with only the key attribute(s) and attributes attached to that entity set; entity represented in all relations to whose subclass/entity set it belongs.
2. *Object-oriented*: each entity belongs to exactly one class; create a relation for each class, with all its attributes.
3. *Use nulls*: create one relation; entities have null in attributes that don't belong to them.

9

Subclasses to Relations: E/R style



Product

Name	Price	Category
Gizmo	99	gadget
Camera	49	photo
Toy	39	gadget

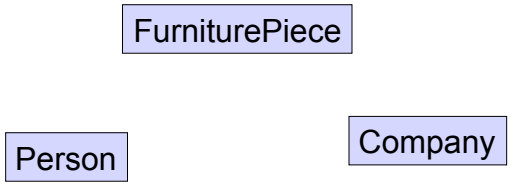
SoftwareProduct

Name	platforms
Gizmo	unix

EducationalProduct

Name	Age Group
Gizmo	todler
Toy	retired

Modeling UnionTypes With Subclasses



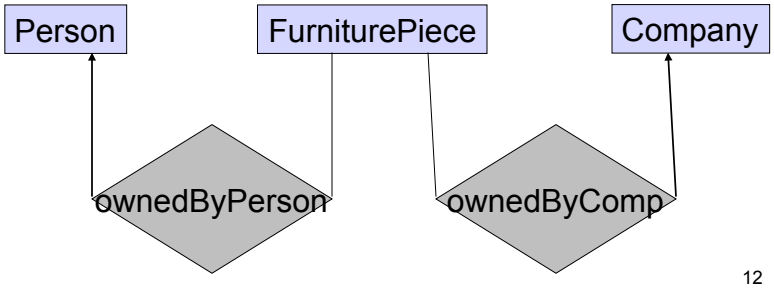
Say: each piece of furniture is owned either by a person, or by a company

11

Modeling Union Types with Subclasses

Say: each piece of furniture is owned either by a person, or by a company

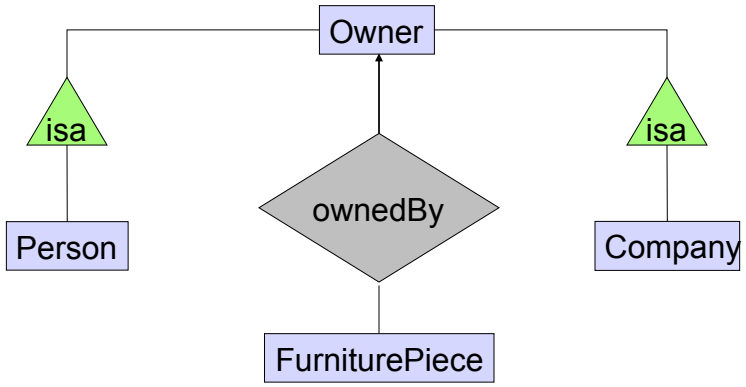
Solution 1. Acceptable (What's wrong ?)



12

Modeling Union Types with Subclasses

Solution 2: More faithful



13

Constraints in E/R Diagrams

Finding constraints is part of the modeling process.
Commonly used constraints:

Keys: social security number uniquely identifies a person.

Single-value constraints: a person can have only one father.

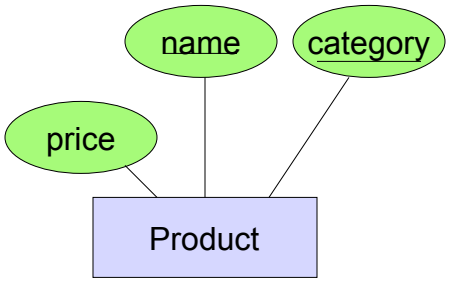
Referential integrity constraints: if you work for a company, it must exist in the database.

Other constraints: peoples' ages are between 0 and 150.

14

Keys in E/R Diagrams

Underline:

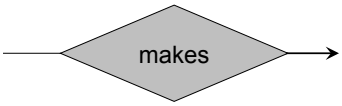


Multi-attribute key v.s. Multiple keys

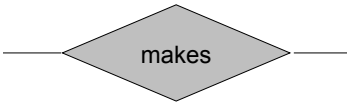
Not possible in E/R

15

Single Value Constraints

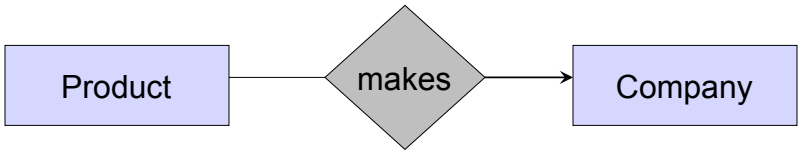


v. s.

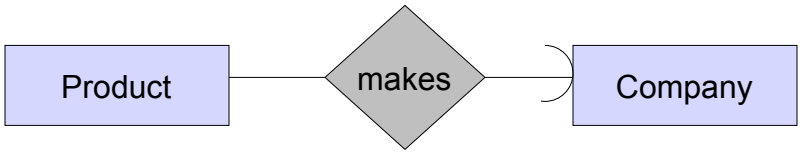


16

Referential Integrity Constraints



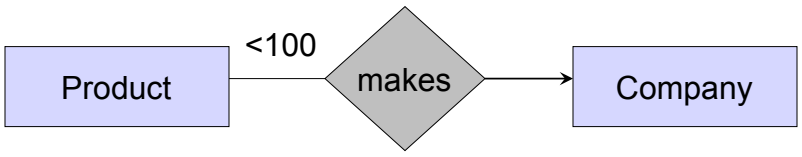
Each product made by at most one company.
Some products made by no company



Each product made by ***exactly*** one company.

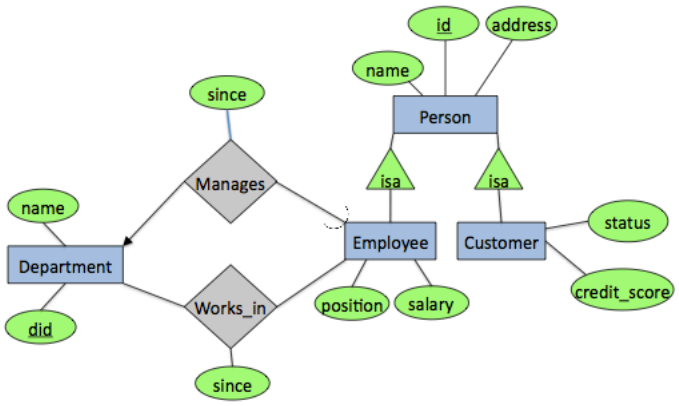
17

Other Constraints



What does this mean?

18



19

E/R to Relations

Person (id, name, address)

Employee (id, position, salary)

Employee.id is a foreign key that references Person.id.

Customer (id, status, credit_score)

Customer.id is a foreign key that references Person.id.

Department(did, name, manager_id, since)

Department.manager_id is a foreign key that references Employee.id

Manager_id is also a key for Department.

Works_in (did, eid, since)

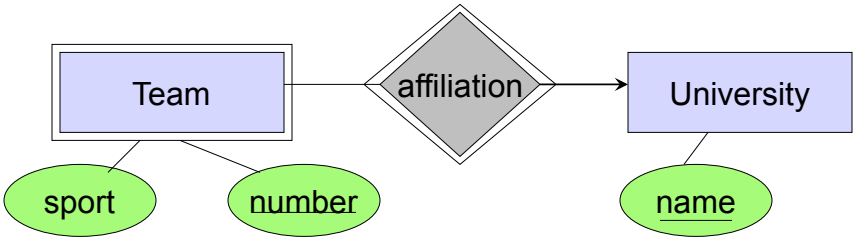
Works_in.did is a foreign key that references Department.did

Works_in.eid is a foreign key that references Employee.id

Weak Entity Sets

Entity sets are weak when their key comes from other classes to which they are related. This happens if:

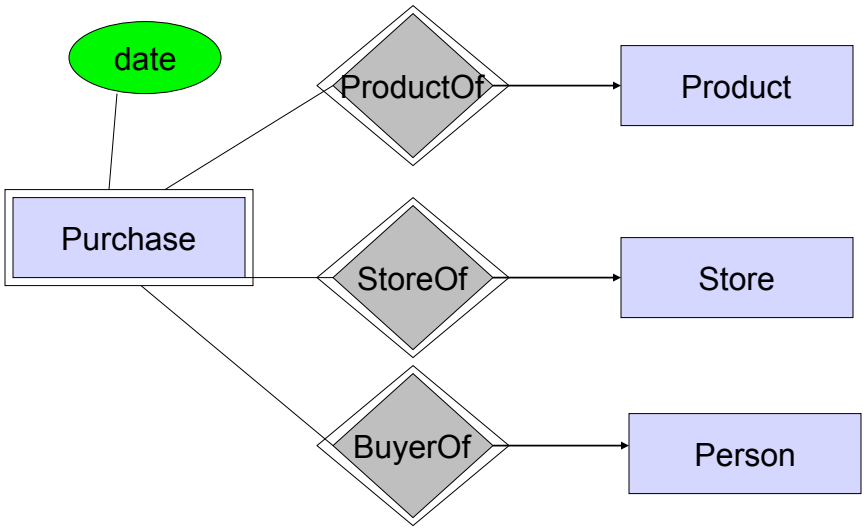
- "part-of" hierarchies
- splitting n-ary relations to binary.



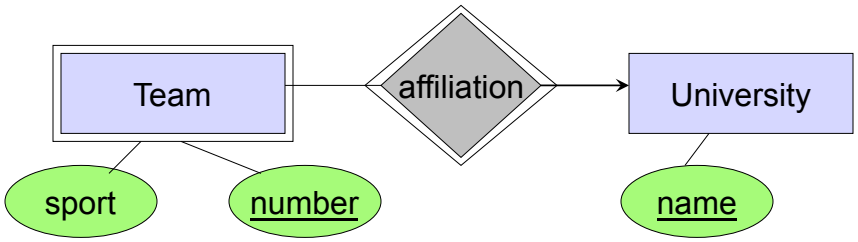
Notice: we encountered this when converting multiway relationships to binary relationships

21

Weak Entity Sets



Handling Weak Entity Sets



How do we represent this with relations ?

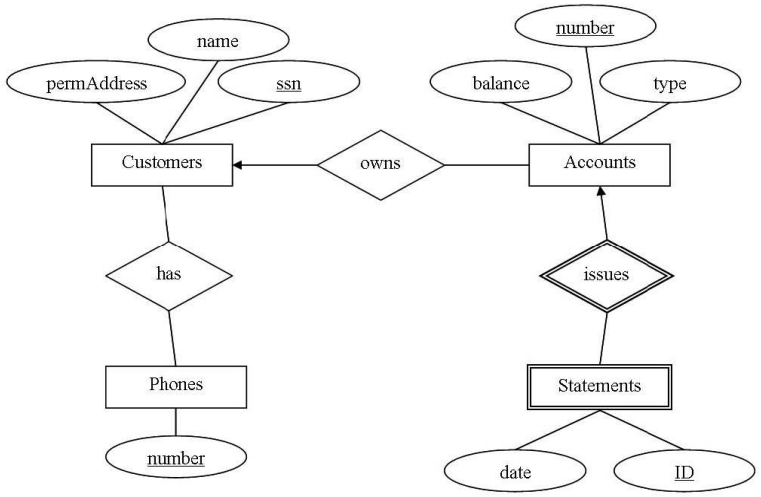
Team(sport, <u>number</u>, <u>name</u>)

23

Exercise

Draw an ER diagram to design a database for a banking system, which maintains information about customers and their accounts with the following assumptions:

- Each customer has a name, a permanent address, and a social security number.
- Each customer can have multiple phone numbers, and the same phone number may be shared by multiple customers.
- A customer can own multiple accounts, but each account is owned by a single customer.
- Each account has an account number, a type (such as saving, checking, etc.), and a balance.
- The bank issues an account statement for each account and mails it to its account owner every month. As time goes on, there will be multiple statements of the same account.
- Each statement has an issued date and a statement ID. All the statements of the same account have different statement IDs, but two different accounts could have statements with the same statement ID. For example, it is possible that account A has a statement with ID ‘123’, while account B has another statement with the same ID ‘123’.



25

FROM E/R to Relations?

Translate your ER diagram into a set of relations. Select approaches that yield the smallest number of relations; merge relations where appropriate.

Customers(ssn, name, permAddress)

Accounts(number, type, balance, ssn)

- Accounts.ssn is a foreign key that references Customers.ssn
- Accounts.ssn is also a key for Accounts.

Has(ssn, number)

- Has.ssn is a foreign key that references Customers.ssn

Statements(number, ID, date)

- Statements.number is a foreign key for Accounts.number