Logical Database Design: Entity-Relation Models

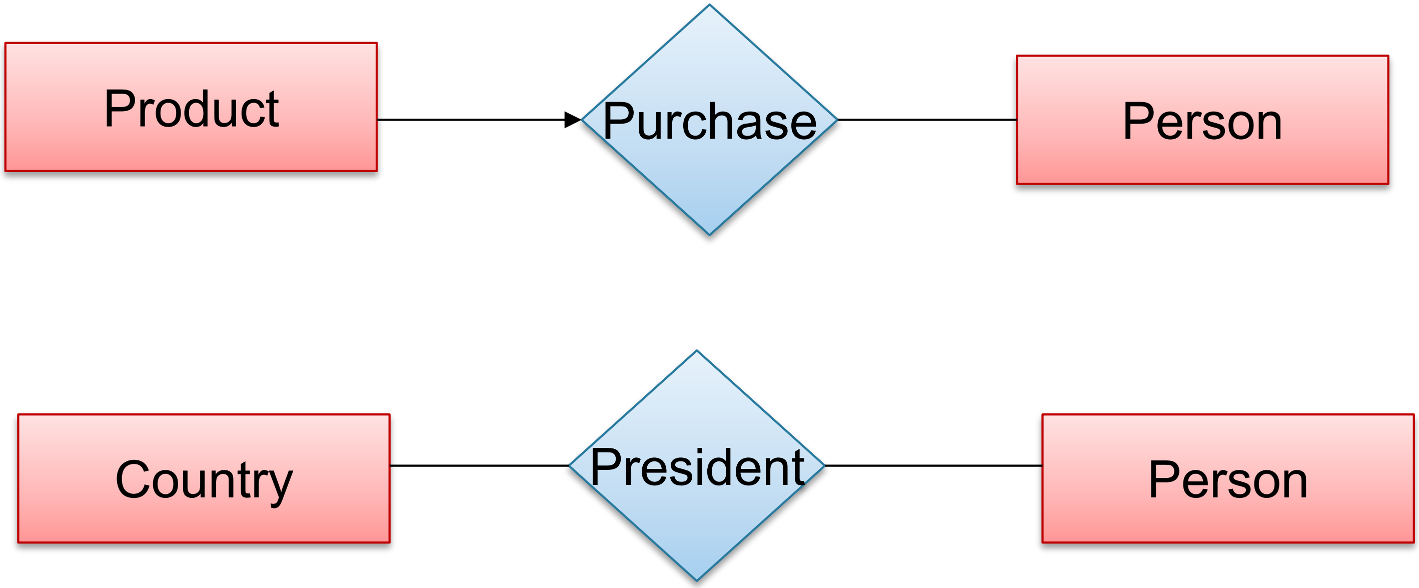
Translating ER diagrams to relations

Alvin Cheung
Aditya Parameswaran

Reading: R & G Chapter 2

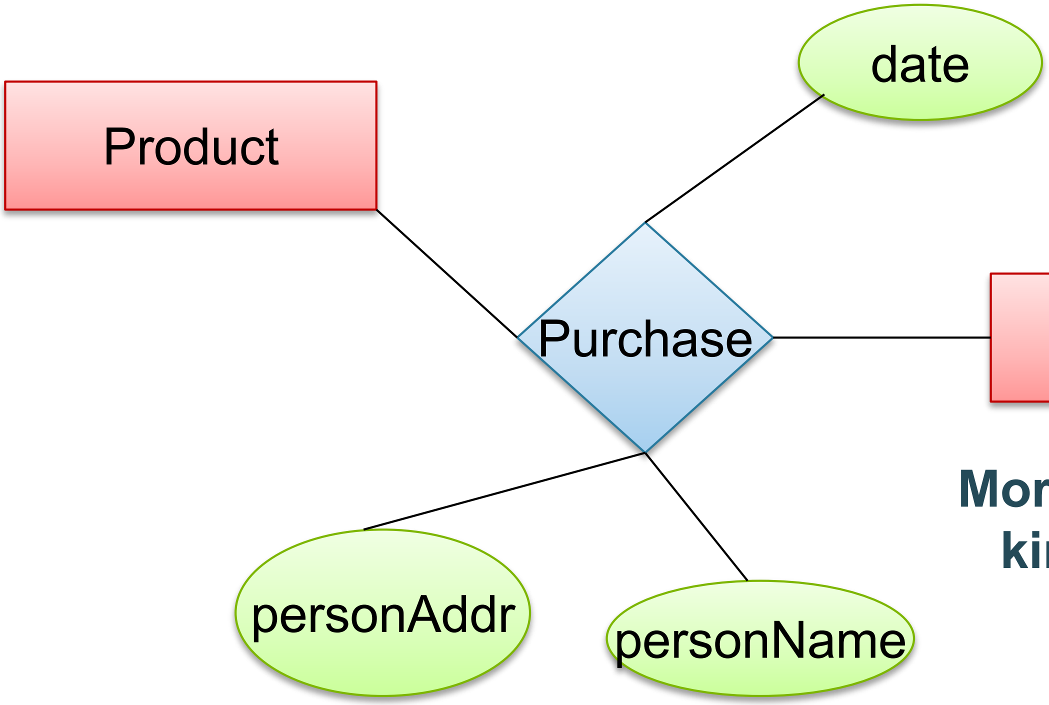
Design Principles

What's wrong?



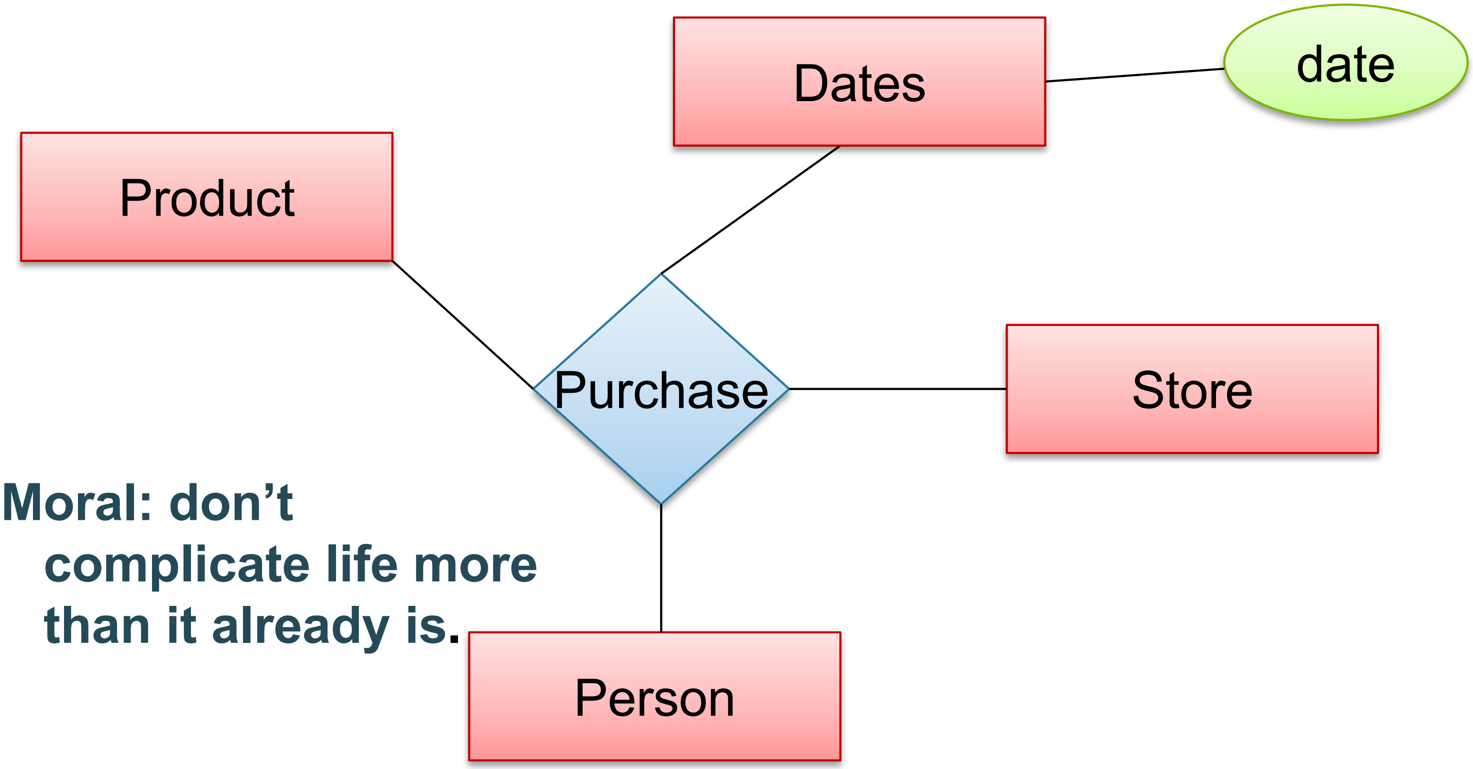
Moral: Be faithful to the specifications of the application!

Design Principles: What's Wrong?



Moral: pick the right kind of entities.

Design Principles: What’s Wrong?



Moral: don’t complicate life more than it already is.

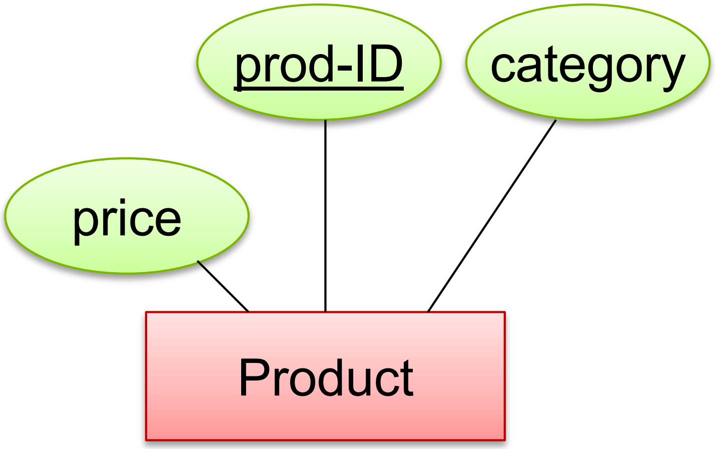
Steps in Database Design, Part 4

- Requirements Analysis
  - user needs; what must database do?
- Conceptual Design
  - *high level description (often done w/ER model)* ← Completed
  - ORM encourages you to program here
- **Logical Design**
  - **translate ER into DBMS data model** ← You are here
  - **ORMs often require you to help here too**
- Schema Refinement
  - consistency, normalization
- Physical Design - indexes, disk layout
- Security Design - who accesses what, and how

Converting ER to Relations

- Fairly analogous structure
- But many simple concepts in ER are subtle to specify in relations

Entity Set to Relation



prod-ID

category

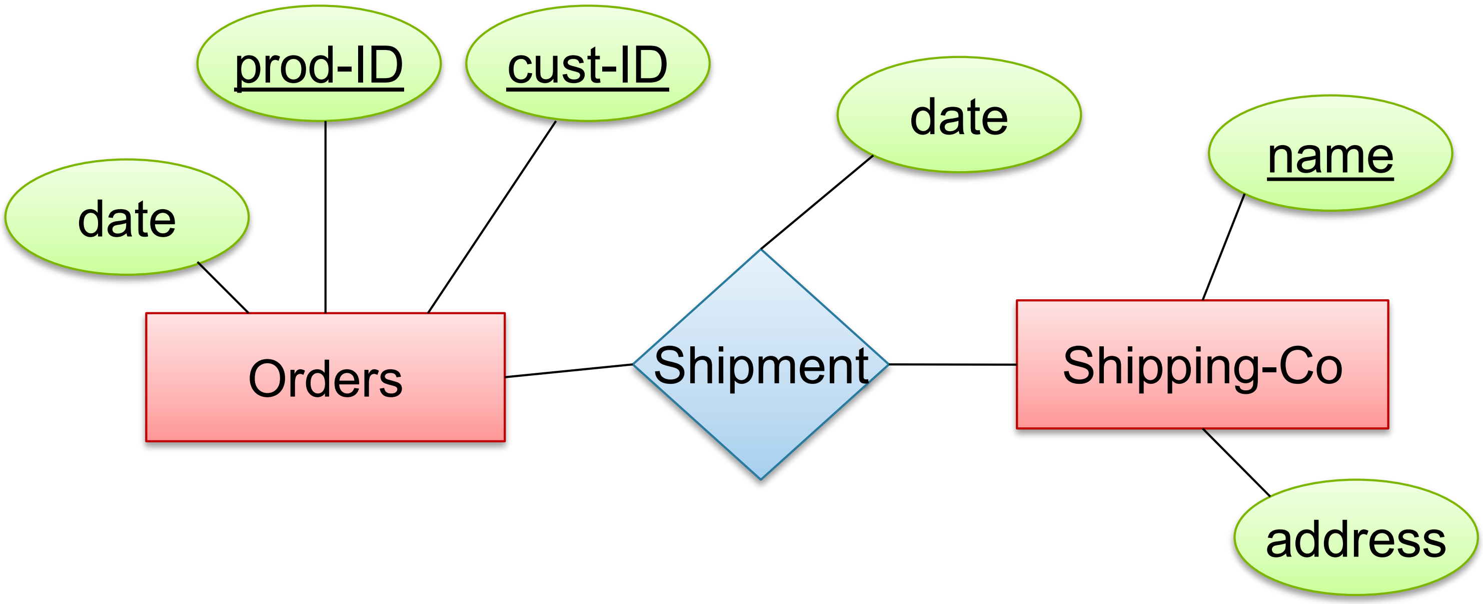
price

Product

Product(prod-ID, category, price)

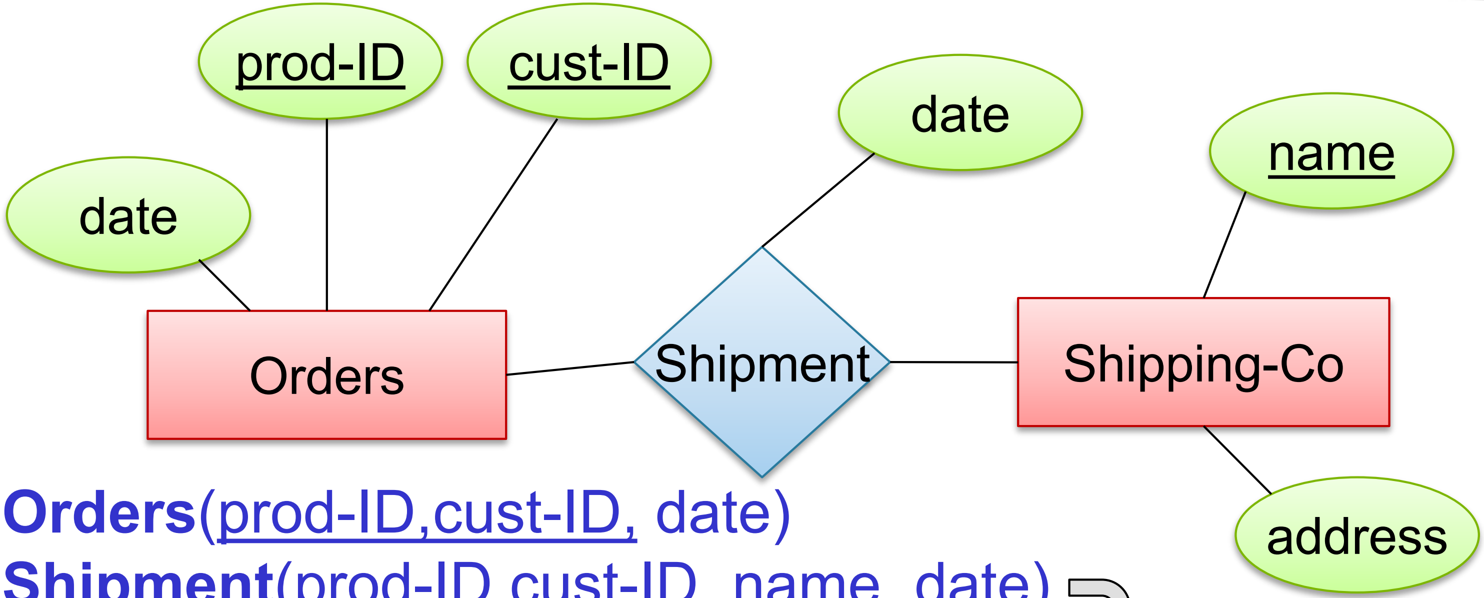
prod-ID	category	price
Gizmo55	Camera	99.99
Pokemn19	Toy	29.99

N-N Relationships to Relations



Represent this in relations

N-N Relationships to Relations



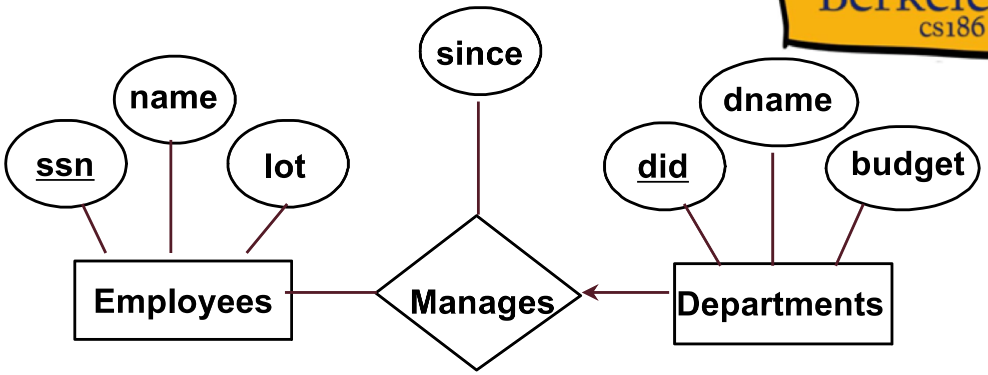
Orders(prod-ID,cust-ID, date)
Shipment(prod-ID,cust-ID, name, date)
Shipping-Co(name, address)

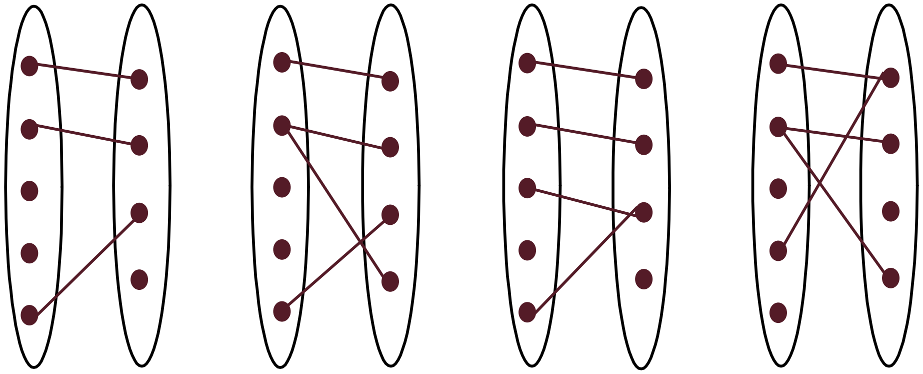
prod-ID	cust-ID	name	date
Gizmo55	Joe12	UPS	4/10/2011
Gizmo55	Joe12	FEDEX	4/9/2011

Note that keys from each participating entity set appear as foreign keys. This set of attributes forms a ***superkey*** for the relation.

Review: Key Constraints

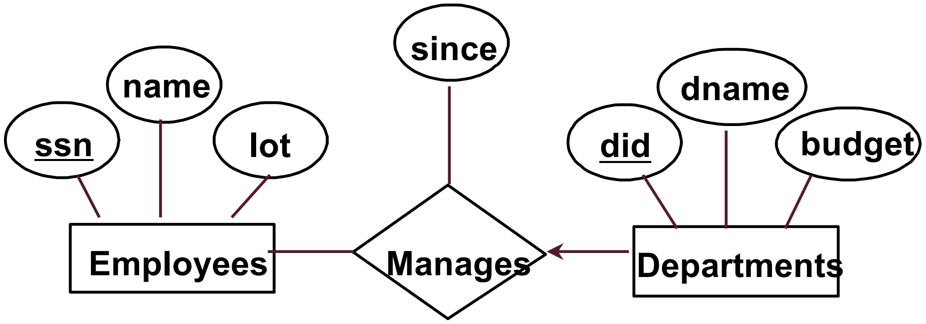
Each dept has at most one manager, according to the **key constraint** on Manages.





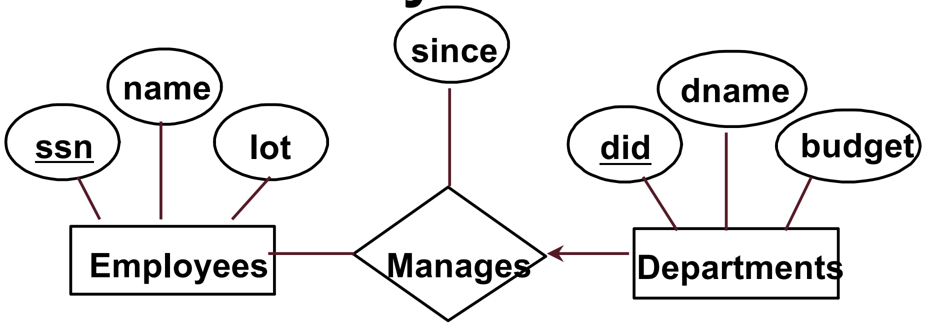
1-to-1 1-to Many Many-to-1 Many-to-Many

Translating ER with Key Constraints



```
CREATE TABLE  Manages(
 ssn  CHAR(11),
 did  INTEGER,
 since  DATE,
 PRIMARY KEY (did),
 FOREIGN KEY (ssn)
   REFERENCES Employees,
 FOREIGN KEY (did)
   REFERENCES Departments)
```

Translating ER with Key Constraints



Since each department has a unique manager, we could instead combine Manages and Departments.

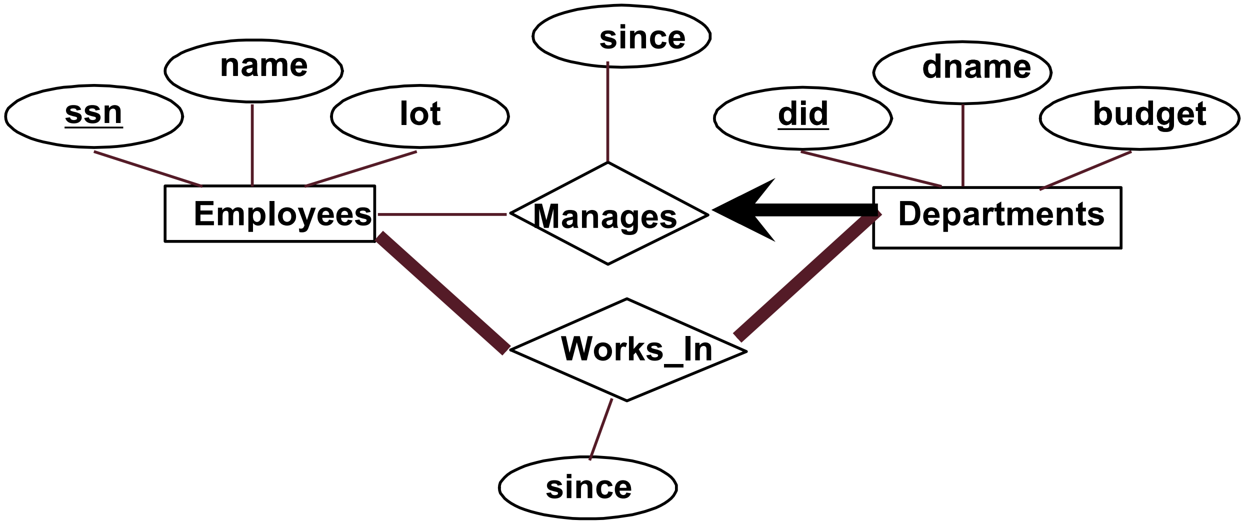
```
CREATE TABLE  Manages(
 ssn  CHAR(11),
 did  INTEGER,
 since  DATE,
 PRIMARY KEY (did),
 FOREIGN KEY (ssn)
   REFERENCES Employees,
 FOREIGN KEY (did)
   REFERENCES Departments)
```

Vs.

```
CREATE TABLE  Dept_Mgr(
 did  INTEGER,
 dname  CHAR(20),
 budget  REAL,
 ssn  CHAR(11),
 since  DATE,
 PRIMARY KEY (did),
 FOREIGN KEY (ssn)
   REFERENCES Employees)
```

Review: Key+Participation Constraints

- Every department has one manager.
  - Every `did` value in `Departments` must appear in a row of `Manages` (with a non-null `ssn`!)

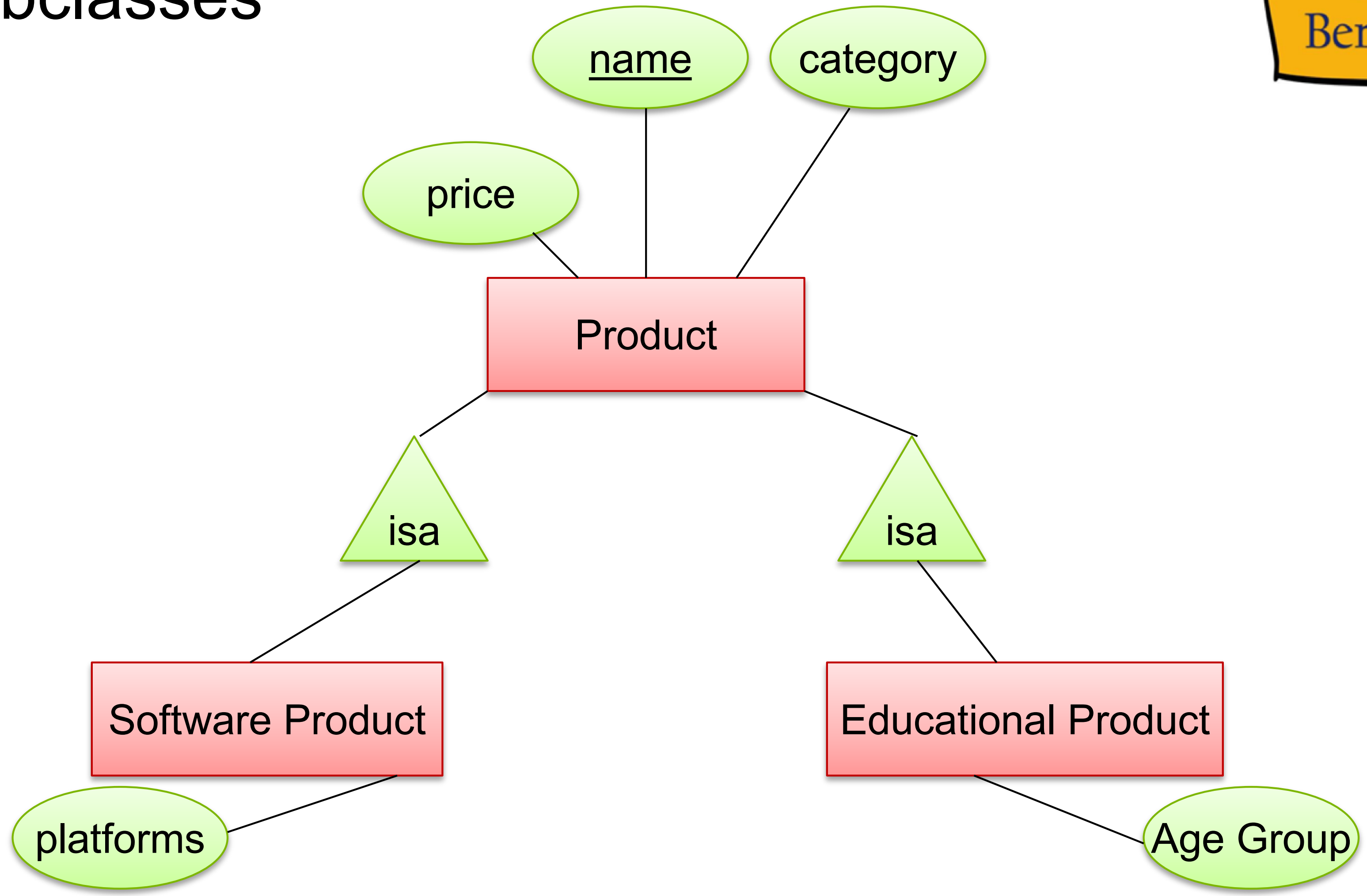


Participation Constraints in SQL

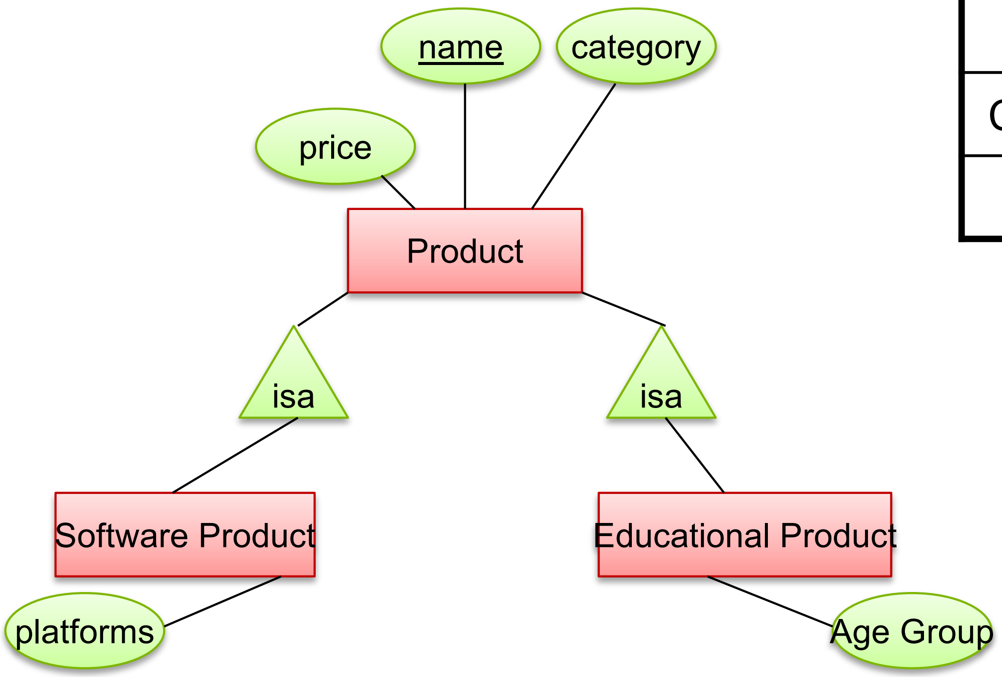
- Participation constraints with one entity set is translated as a binary relationship
- Hard to model more complicated constraints
  - Need to CHECK constraints which we'll learn later.

```
CREATE TABLE  Dept_Mgr(
   did  INTEGER,
   dname  CHAR(20),
   budget  REAL,
   ssn  CHAR(11) NOT NULL, -- total participation!
   since  DATE,
   PRIMARY KEY  (did),
   FOREIGN KEY  (ssn) REFERENCES Employees
      ON DELETE NO ACTION)
```

Subclasses


name
category
price
Product
isa
isa
Software Product
Educational Product
platforms
Age Group

Subclasses to Relations



Other ways to convert are possible

Product

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

Sw.Product

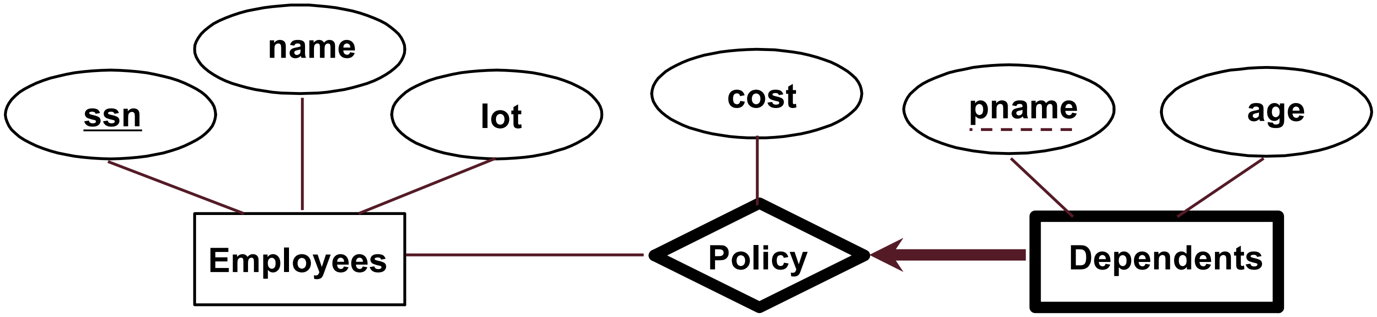
Name	platforms
Gizmo	unix

Ed.Product

Name	Age Group
Gizmo	toddler
Toy	retired

Review: Weak Entities

- A **weak entity** can be identified uniquely only by considering the primary key of another (owner) entity.
  - Owner entity set and weak entity set must participate in a one-to-many relationship set (1 owner, many weak entities).
  - Weak entity set must have total participation in this **identifying** relationship set.



Translating Weak Entity Sets

- Weak entity set and identifying relationship set are translated into a single table.
  - When the owner entity is deleted, all owned weak entities must also be deleted.

```
CREATE TABLE  Dep_Policy (
   pname  CHAR(20),
   age  INTEGER,
   cost  REAL,
   ssn  CHAR(11) NOT NULL,
   PRIMARY KEY (pname, ssn),
   FOREIGN KEY (ssn) REFERENCES Employees
      ON DELETE CASCADE)
```

Summary of ER

- ER design is **subjective.** Many ways to model a given scenario!
- Analyzing alternatives can be tricky! Common choices include:
  - Entity vs. attribute, entity vs. relationship, binary or n-ary relationship, whether or not to use aggregation
- For good DB design: resulting relational schema should be easily analyzable:
  - Capture functional dependencies
  - Can be *normalized*