What is a Database Management System?

1. Manages very large amounts of data.
2. Supports efficient access to very large amounts of data.
3. Supports concurrent access to v.l.a.d.
   - Example: bank and its ATM machines.
4. Supports secure, atomic access to v.l.a.d.
   - Contrast two people editing the same UNIX file — last to write "wins" — with the problem if two people deduct money from the same account via ATM machines at the same time — new balance is wrong whichever writes last.

Relational Model

- Based on tables, as:

acct#	name	balance
12345	Sally	1000.21
34567	Sue	285.48
...	...	...

- Today used in *most* DBMS's.

The DBMS Marketplace

- Relational DBMS companies — Oracle, Informix, Sybase — are among the largest software companies in the world.
- IBM offers its relational DB2 system. With IMS, a nonrelational system, IBM is by some accounts the largest DBMS vendor in the world.
- Microsoft offers SQL-Server, plus Microsoft *Access* for the cheap DBMS on the desktop, answered by "lite" systems from other competitors.
- Relational companies also challenged by "object-oriented DB" companies.
- But countered with "object-relational" systems, which retain the relational core while allowing type extension as in OO systems.

Three Aspects to Studying DBMS's

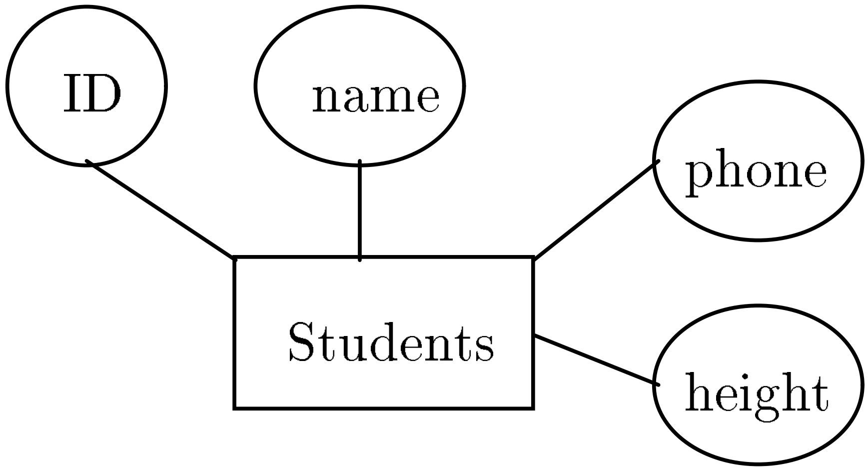
1. Modeling and design of databases.
   - Allows exploration of issues before committing to an implementation.
2. Programming: queries and DB operations like update.
   - SQL = "intergalactic dataspeak."
3. DBMS implementation.

CS145 = (1) + (2), while (3) is covered in CS245, CS346, CS347.

Entity/Relationship Model

Diagrams to represent designs.

- *Entity* like object, = “thing.”
- *Entity set* like class = set of “similar” entities/objects.
- *Attribute* = property of entities in an entity set, similar to fields of a struct.
- In diagrams, entity set → rectangle; attribute → oval.



Relationships

- Connect two or more entity sets.
- Represented by diamonds.

Relationship Set

Think of the "value" of a relationship set as a table.

- One column for each of the connected entity sets.
- One row for each list of entities, one from each set, that are connected by the relationship.

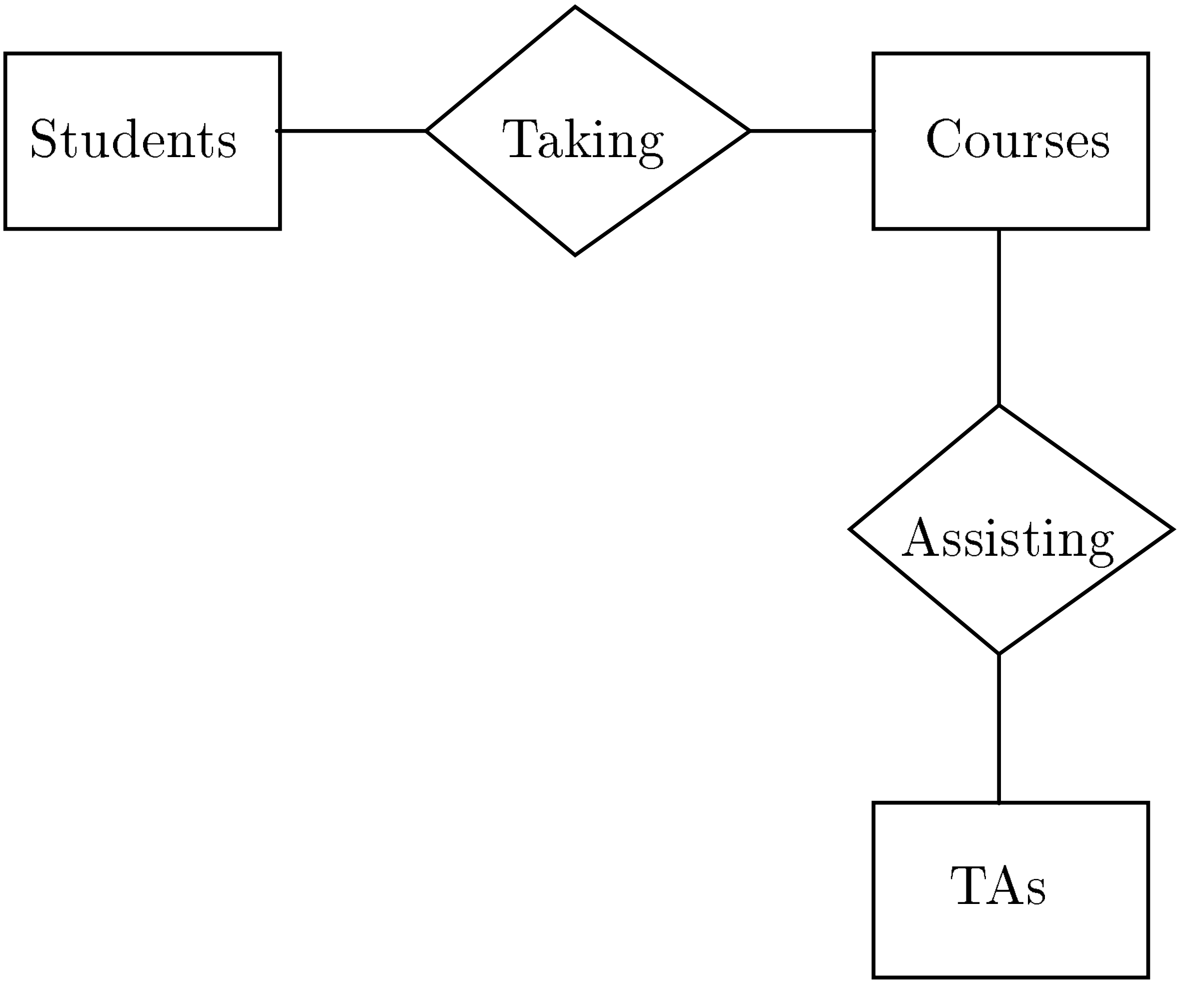
Students	Courses
Sally	CS145
Sally	CS244
Joe	CS145
...	...

Multiway Relationships

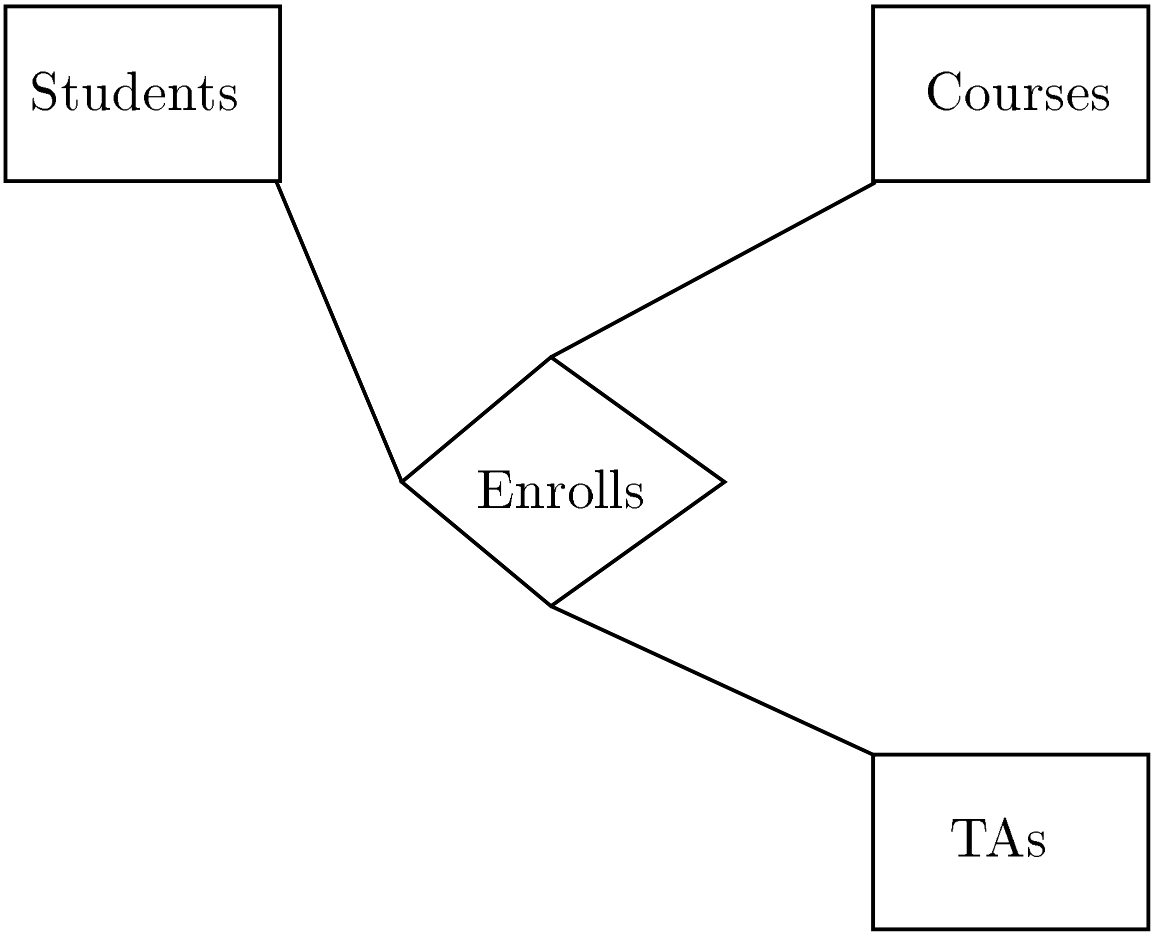
Usually *binary* relationships (connecting two E.S.) suffice.

- However, there are some cases where three or more E.S. must be connected by one relationship.

- Example: relationship among students, courses, TA's. Possibly, this E/R diagram is OK:



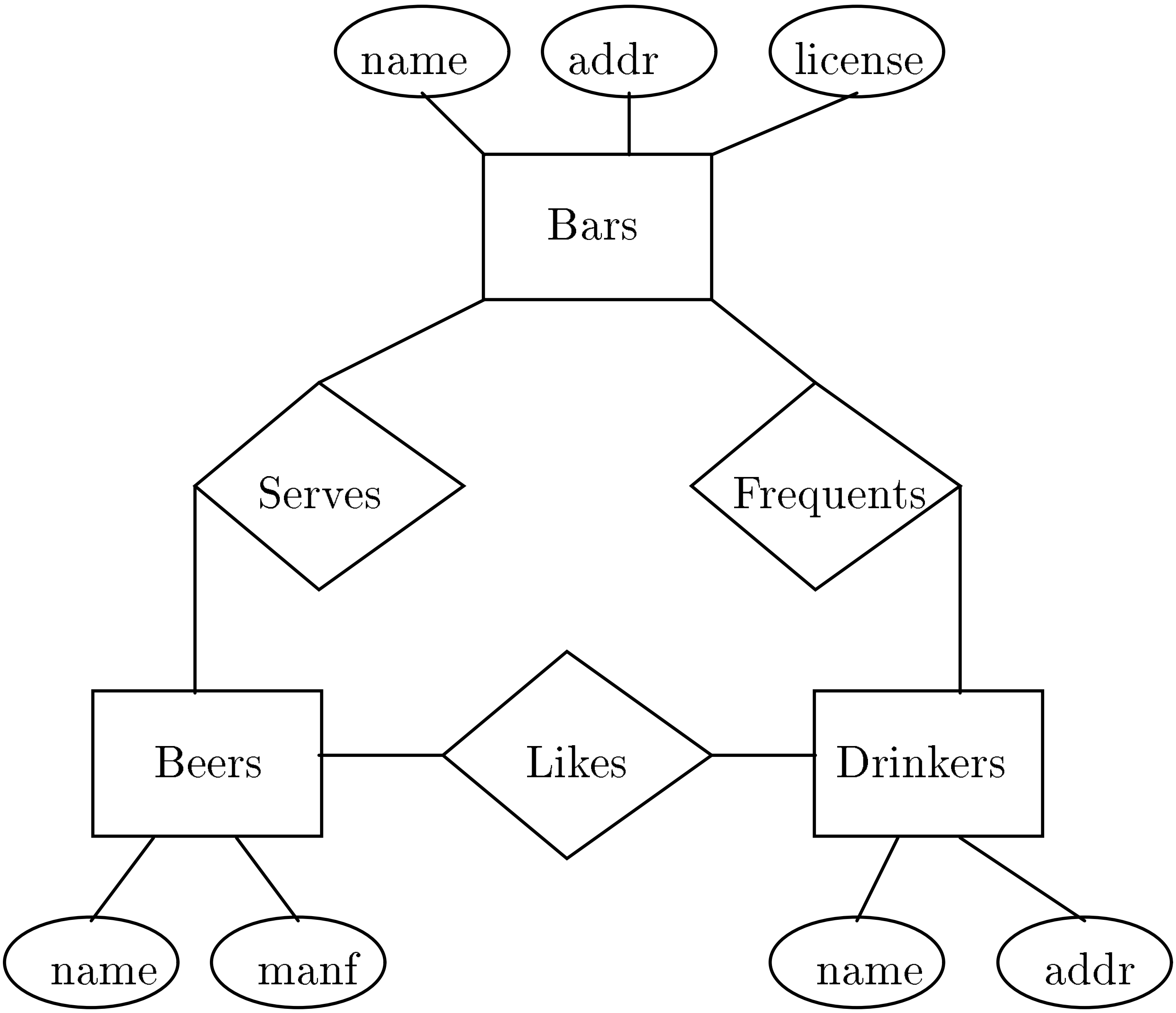
- Works in CS145, because each TA is a TA of all students. Connection student-TA is *only* via the course.
- But what if students were divided into sections, each headed by a TA?
  - Then, a student in CS145 would be related to only one of the TA's for CS145. Which one?
- Need a 3-way relationship to tell.



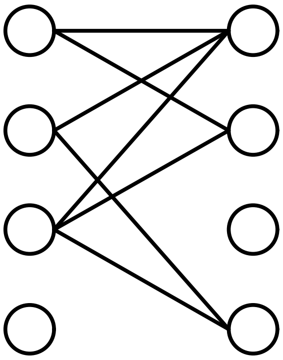
Students	Courses	TAs
Ann	CS145	Jim
Sue	CS145	Roy
Bob	CS145	Jim
...	...	...

Beers-Bars-Drinkers Example

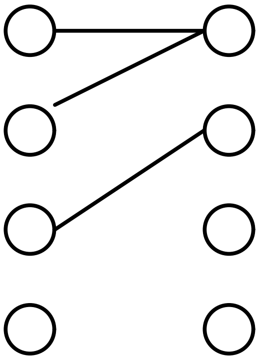
- Our running example for the course.



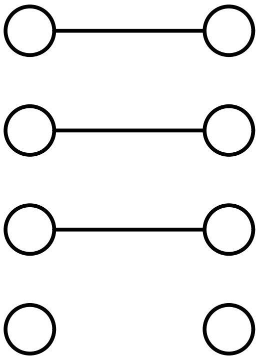
Multiplicity of Relationships



Many-many



Many-one

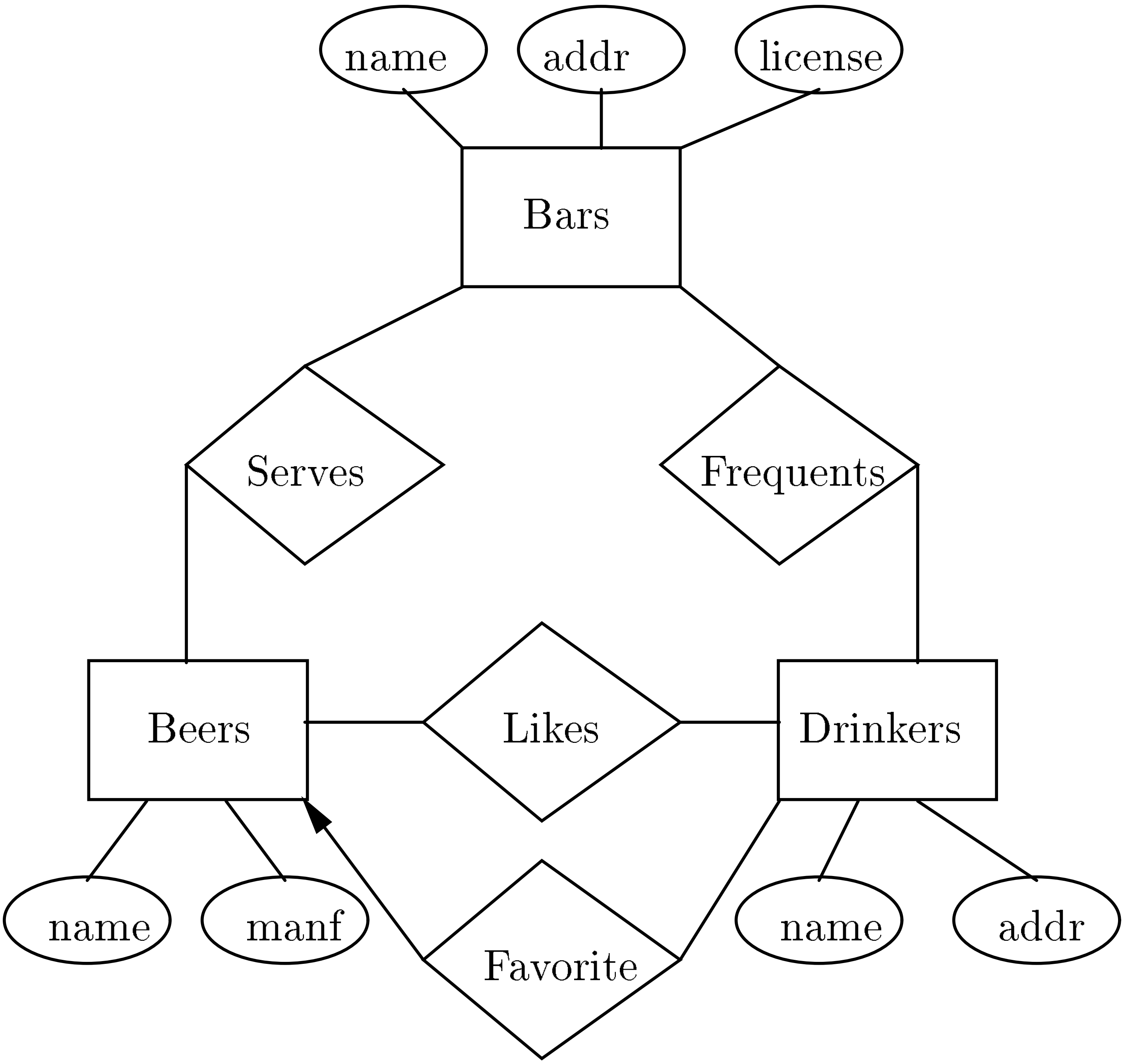


One-one

Representation of Many-One

- E/R: arrow pointing to "one."
    - ❖ Rounded arrow = "exactly one."

Example: Drinkers Have Favorite Beers



One-One Relationships

Put arrows in both directions.

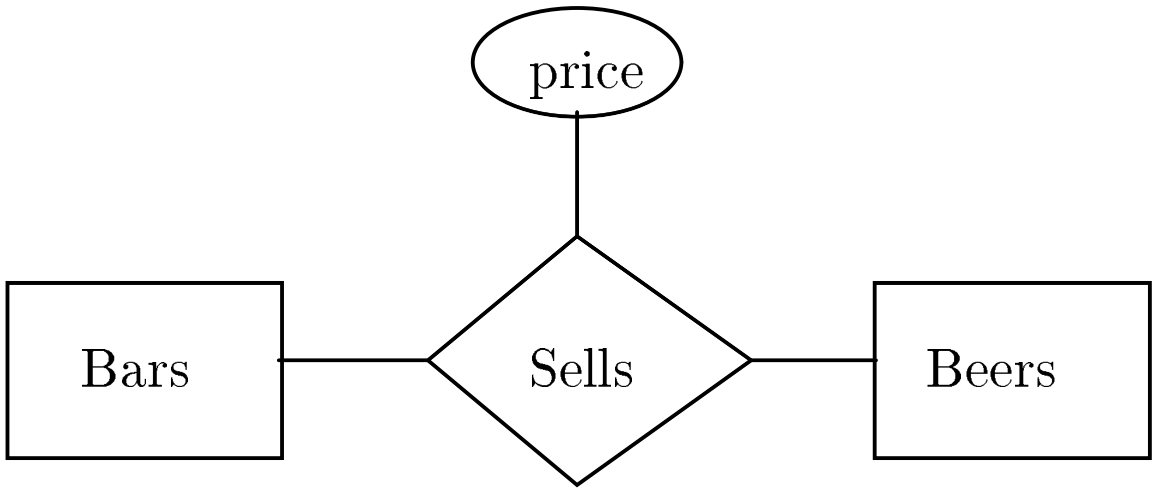
Design Issue:

Is the rounded arrow justified?

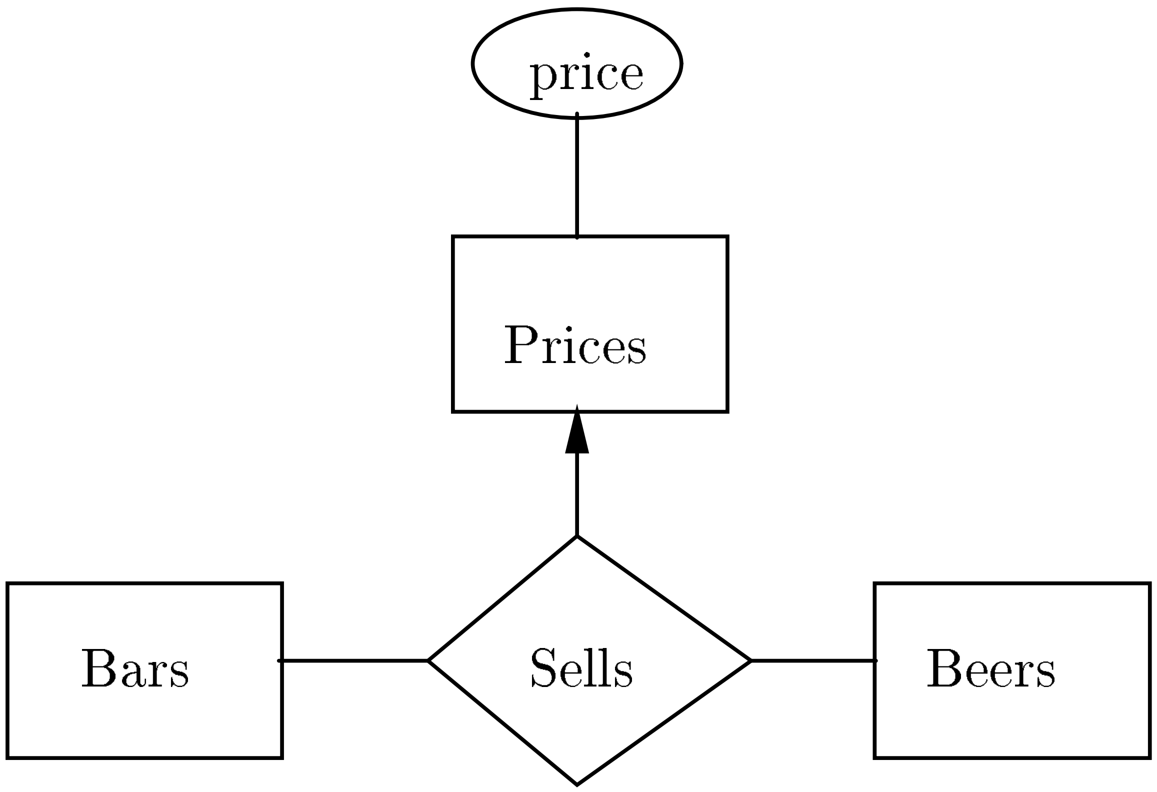
Design Issue:

Here, manufacturer is an E.S.; in earlier diagrams it is an attribute. Which is right?

Attributes on Relationships



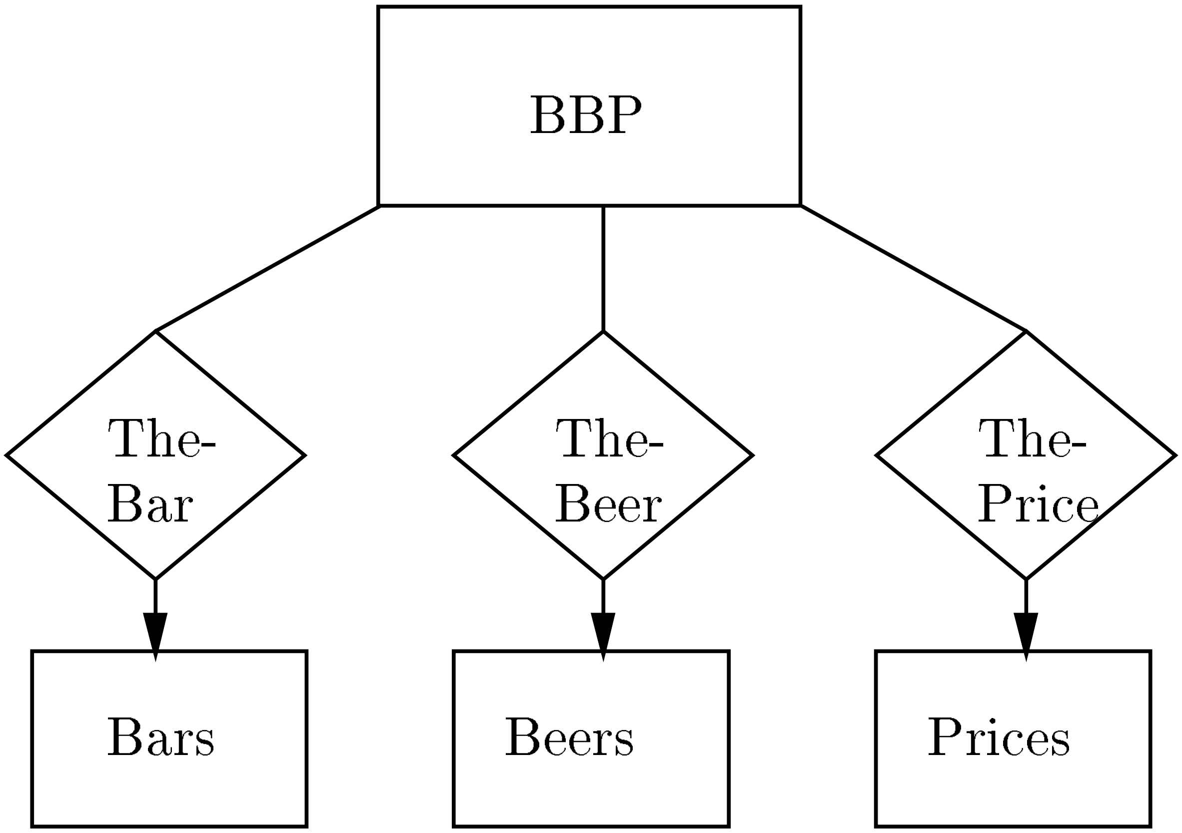
- Shorthand for 3-way relationship:



- A true 3-way relationship.
  - Price depends jointly on beer and bar.
- Notice arrow convention for multiway relationships: "all other E.S. determine one of these."
  - Not sufficiently general to express any possibility.
  - However, if price, say, depended only on the beer, then we could use two 2-way relationships: price-beer and beer-bar.
  - Or better: just make price an attribute of beer.

Converting Multiway to 2-Way

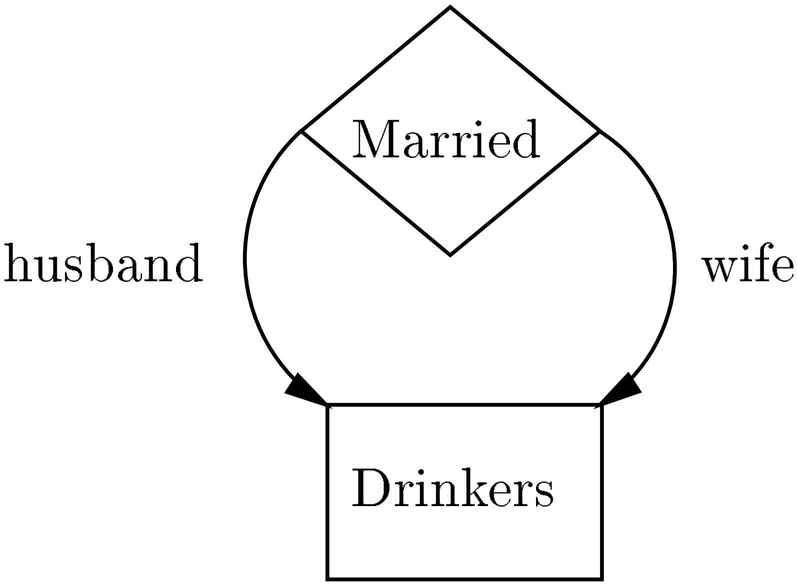
- Baroque in E/R, but necessary in certain "object-oriented" models.
- Create a new *connecting* E.S. to represent rows of a relationship set.
  - E.g., (Joe's Bar, Bud, \$2.50) for the *Sells* relationship.
- Many-one relationships from the connecting E.S. to the others.



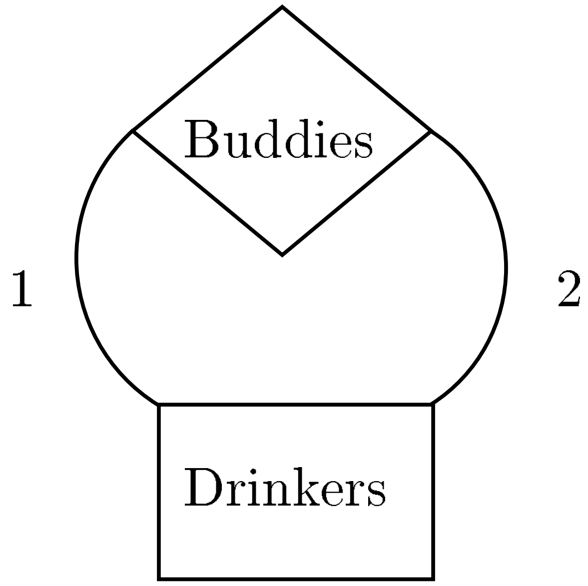
Roles

Sometimes an E.S. participates more than once in a relationship.

- Label edges with *roles* to distinguish.



Husband	Wife
d_1	d_2
d_3	d_4
$\ldots$	$\ldots$



Buddy1	Buddy2
d_1	d_2
d_1	d_3
d_2	d_1
d_2	d_4
$\ldots$	$\ldots$

- Notice *Buddies* is symmetric, *Married* not.
  - ✦ No way to say "symmetric" in E/R.

Design Question

Should we replace `husband` and `wife` by one relationship `spouse`?