

CSCI4333 Database Design & Implement

Lecture Five – E-R Model 4

Instructor: Dr. Yifeng Gao

Ternary Relationships

So far, we have only considered binary relationships

Ternary Relationships

So far, we have only considered binary relationships

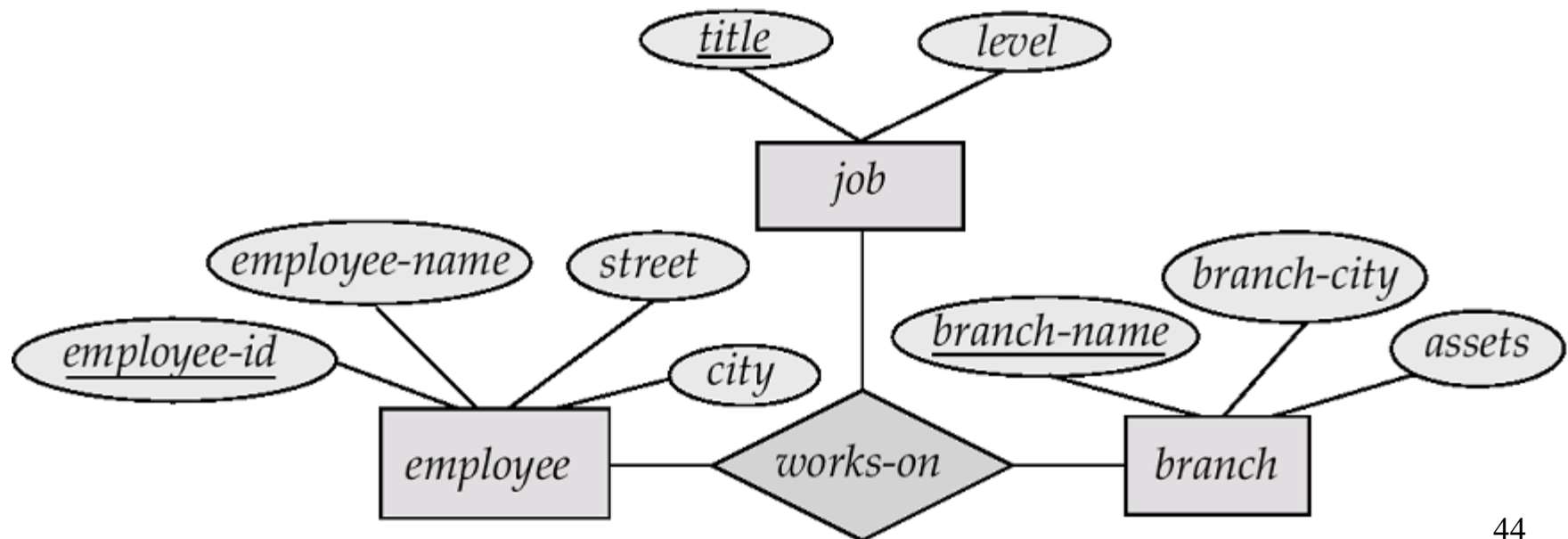
it is possible to have **higher order** relationships, including **ternary** relationships.

Ternary Relationships

So far, we have only considered binary relationships

it is possible to have **higher order** relationships, including **ternary** relationships.

Consider the following example that describes the fact that employees at a bank work in one or more bank branches, and have one or more job descriptions.



Ternary Relationships

Sometimes you have a choice of a single ternary relationship or two binary relationships...

Ternary Relationships

Sometimes you have a choice of a single ternary relationship or two binary relationships...

In general, unless you really need a ternary relationship, use binary relationships.

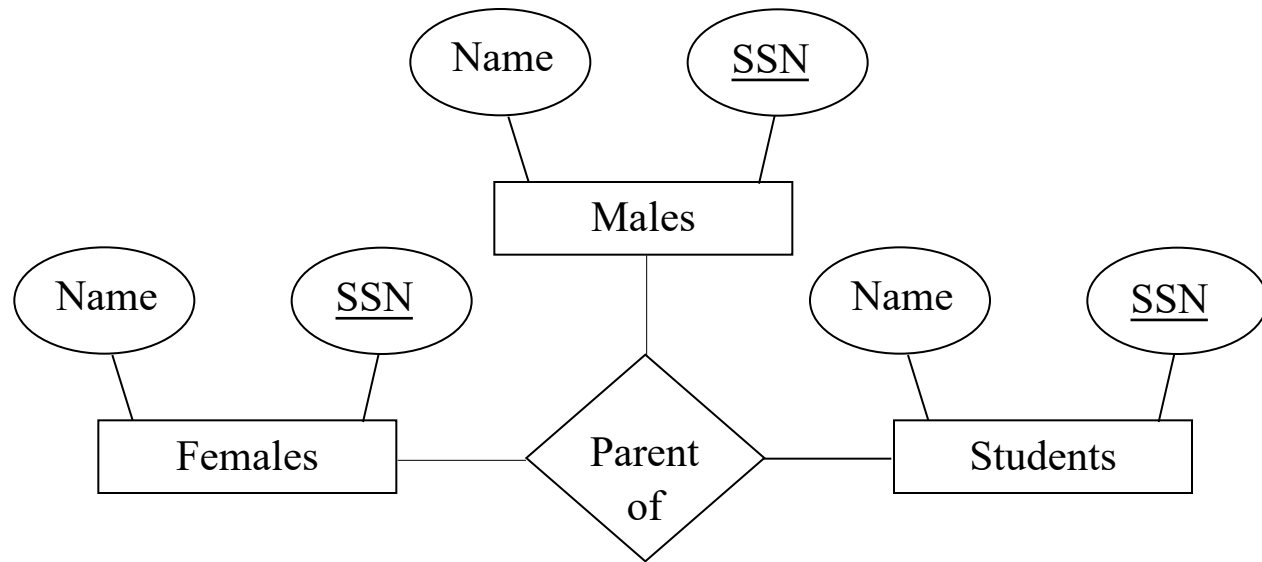
FACT: Every ternary (and higher order) relationship can be converted into a set of binary relationships.

Ternary Relationships

Sometimes you have a choice of a single ternary relationship or two binary relationships...

In general, unless you really need a ternary relationship, use binary relationships.

FACT: Every ternary (and higher order) relationship can be converted into a set of binary relationships.

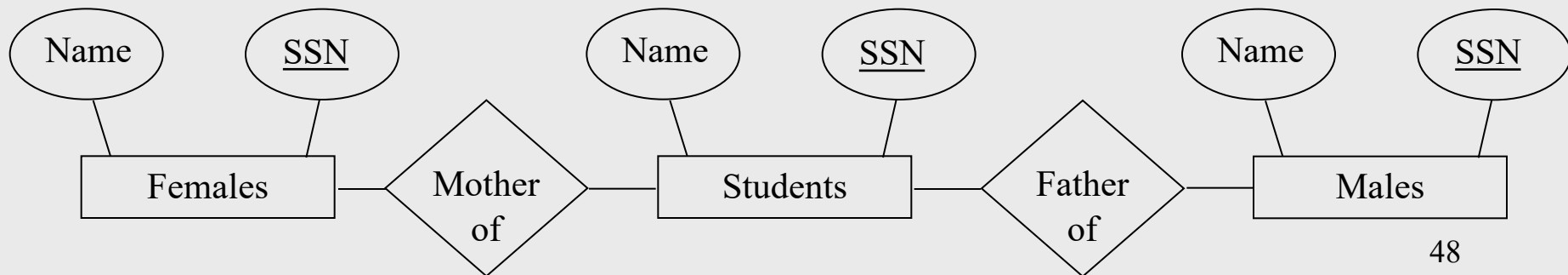
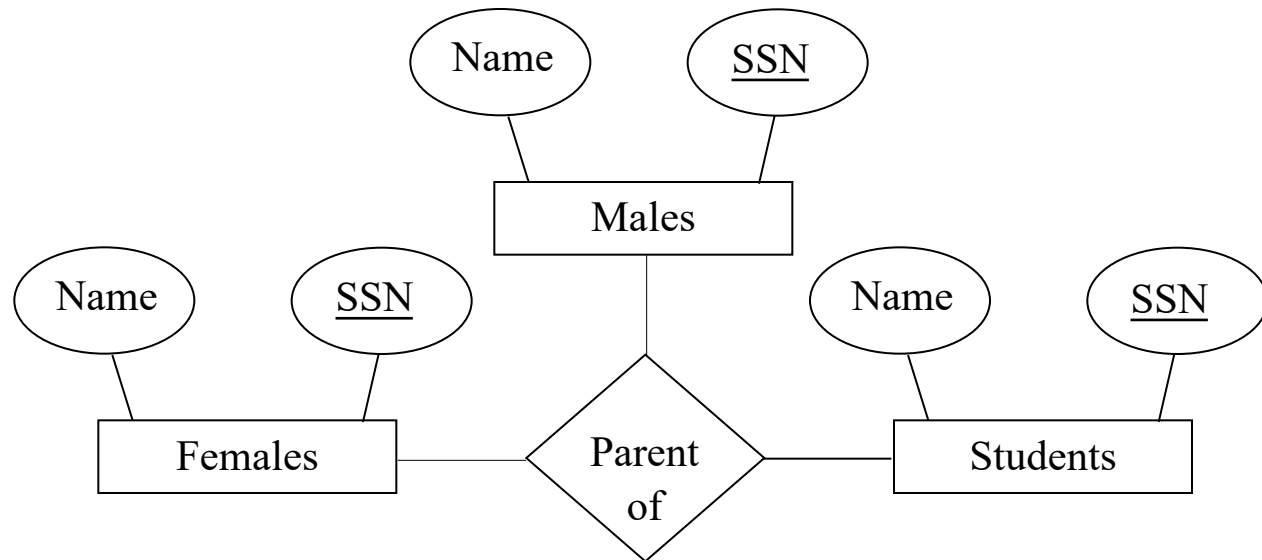


Ternary Relationships

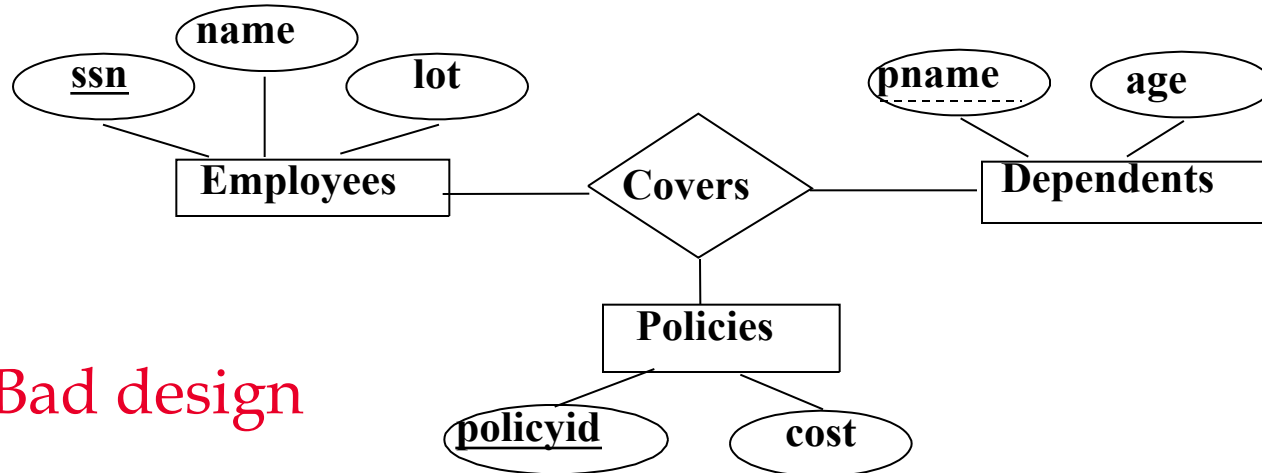
Sometimes you have a choice of a single ternary relationship or two binary relationships...

In general, unless you really need a ternary relationship, use binary relationships.

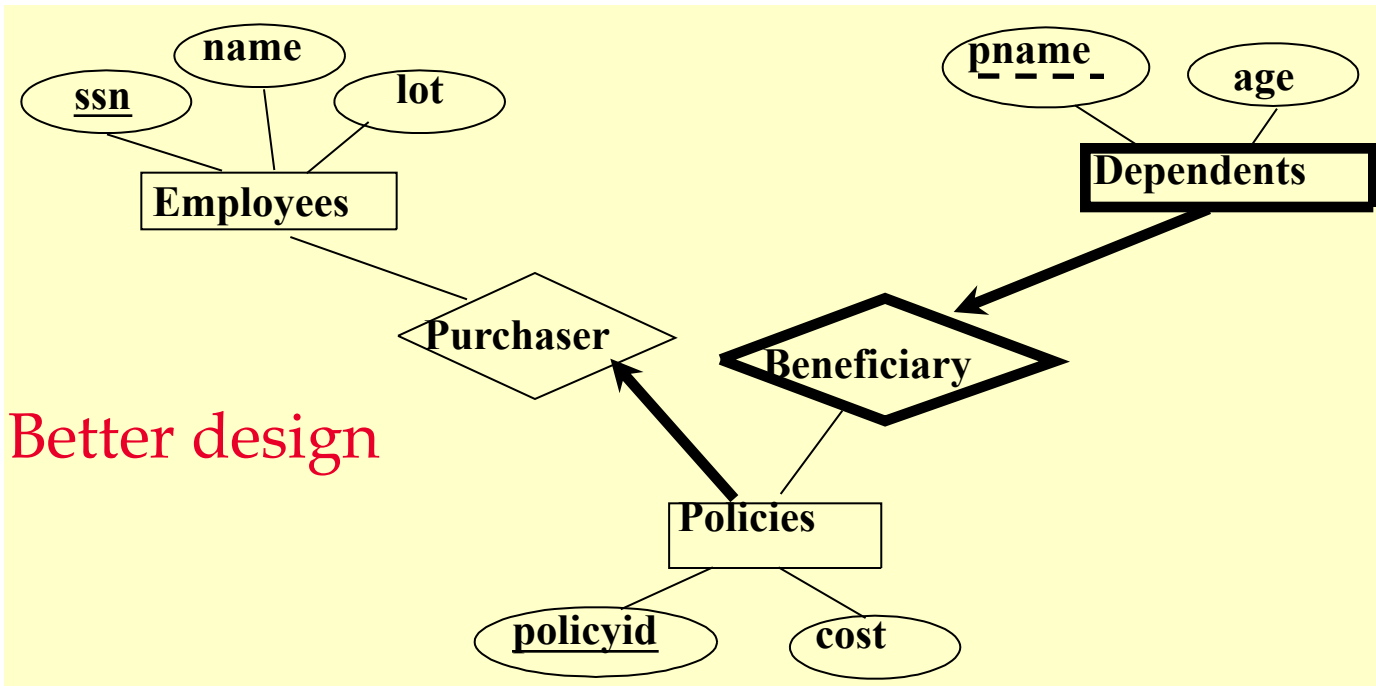
FACT: Every ternary (and higher order) relationship can be converted into a set of binary relationships.



Binary vs. Ternary Relationships

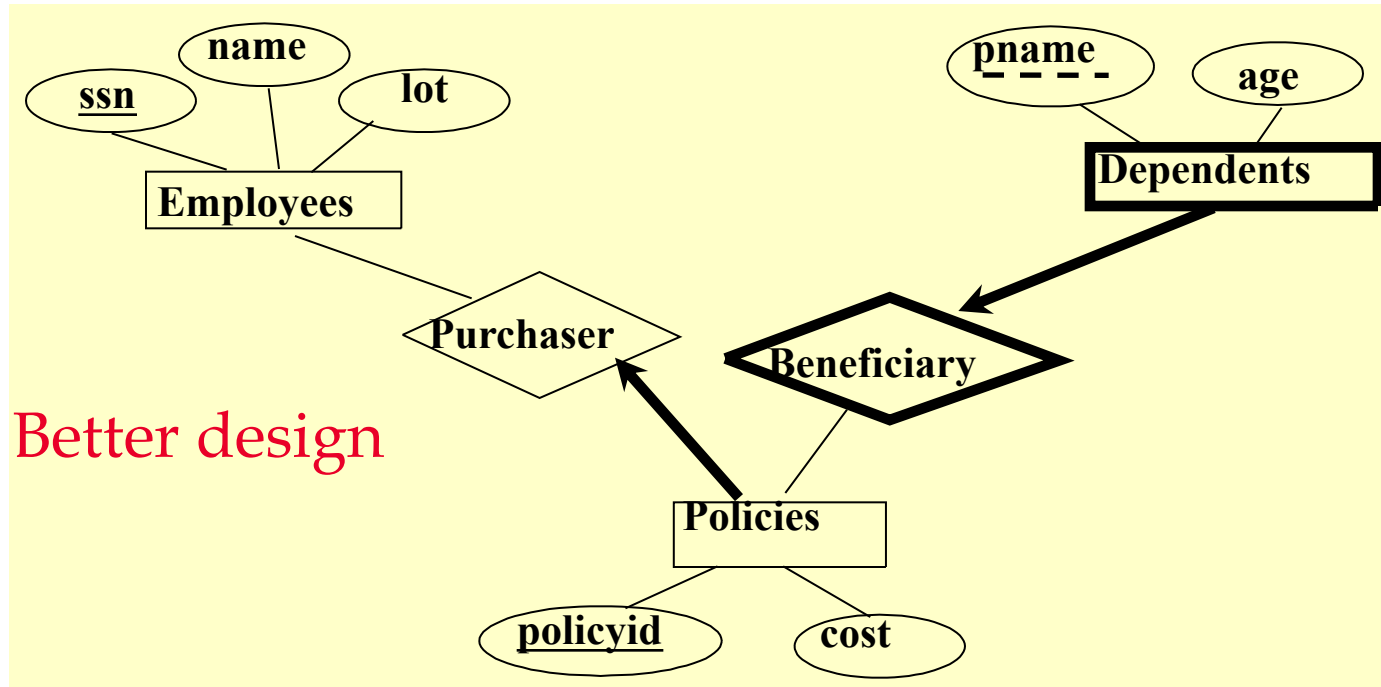


Bad design



Better design

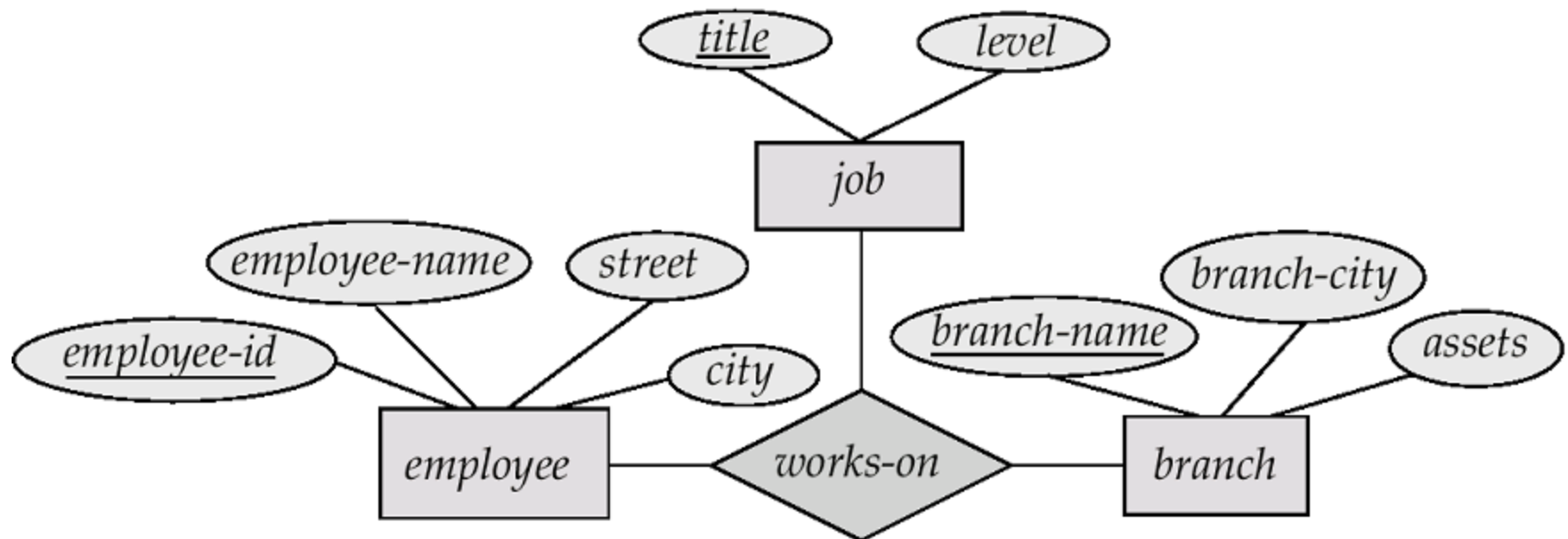
Binary vs. Ternary Relationships



1. Every policy must be owned by some employee.
2. Each policy covers at least one dependent.
3. A policy cannot be owned by two or more employees (one-to-many).
4. A dependent must be matched with some policyid.

Aggregation

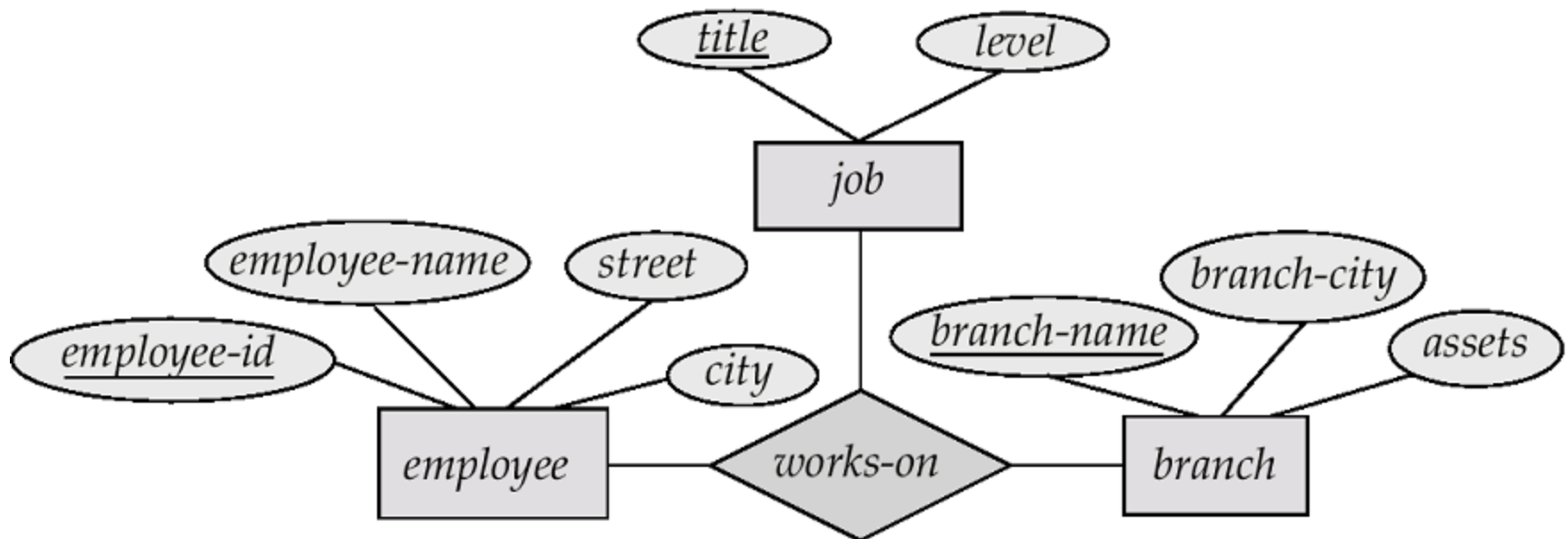
Consider this ER model, which we have seen before...



Aggregation

Consider this ER model, which we have seen before...

We need to add to it, to reflect that **managers manage the various tasks performed by an employee at a branch**



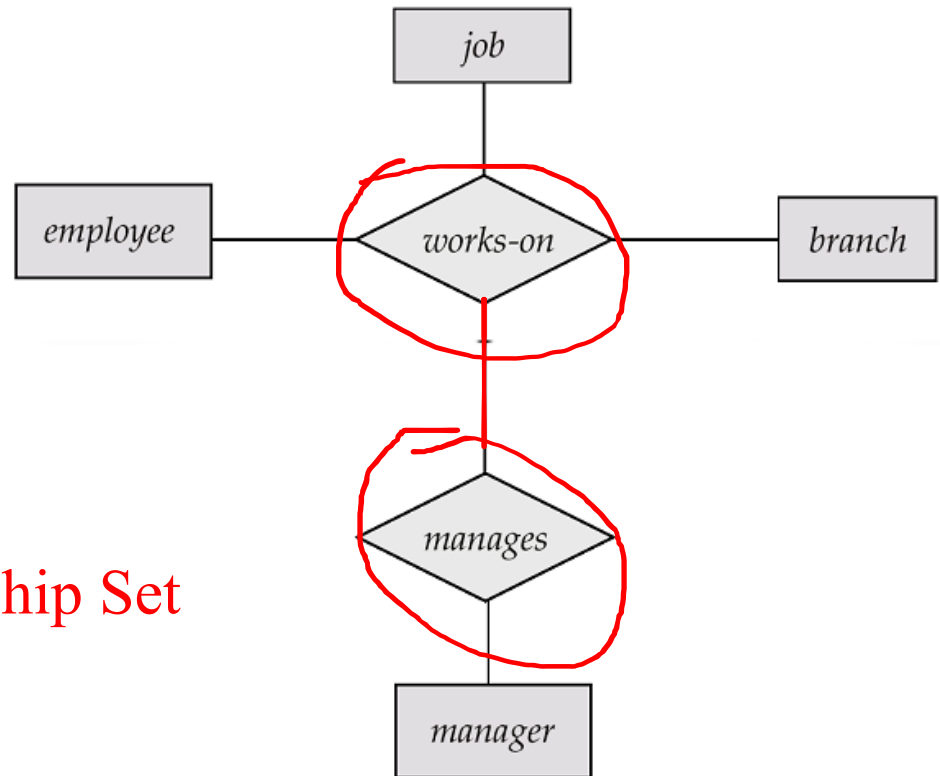
Common Error

Note that the attributes are omitted for graphical simplicity.

This is a wrong ER-diagram!

Do not directly connect

1. Entity Set to Entity Set
2. Relationship Set to Relationship Set



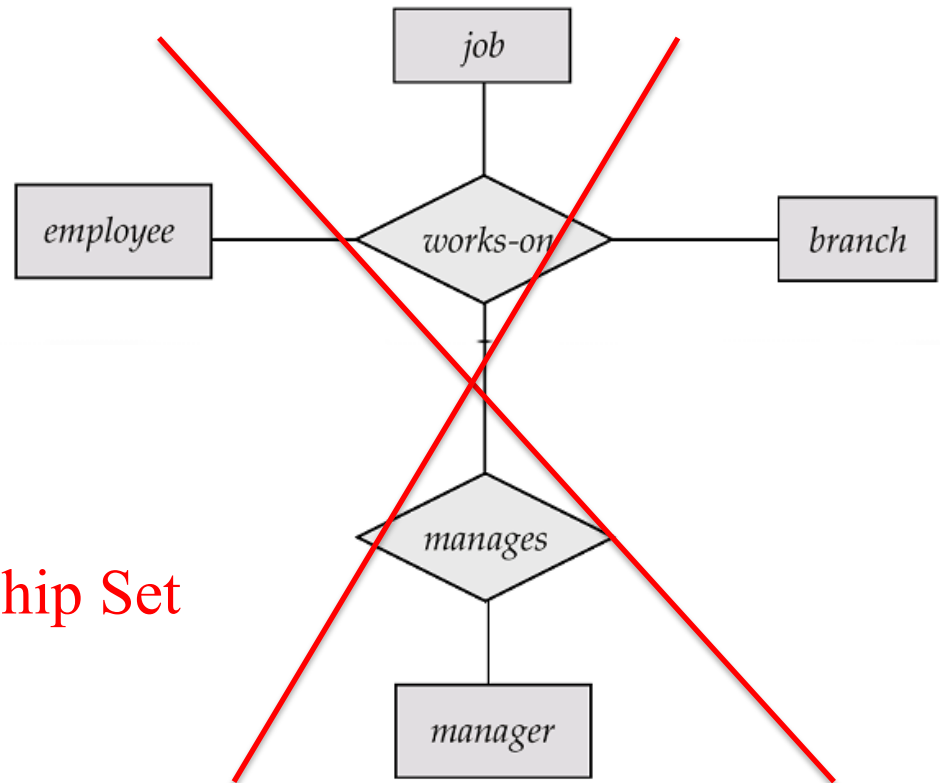
Common Error

Note that the attributes are omitted for graphical simplicity.

This is a wrong ER-diagram!

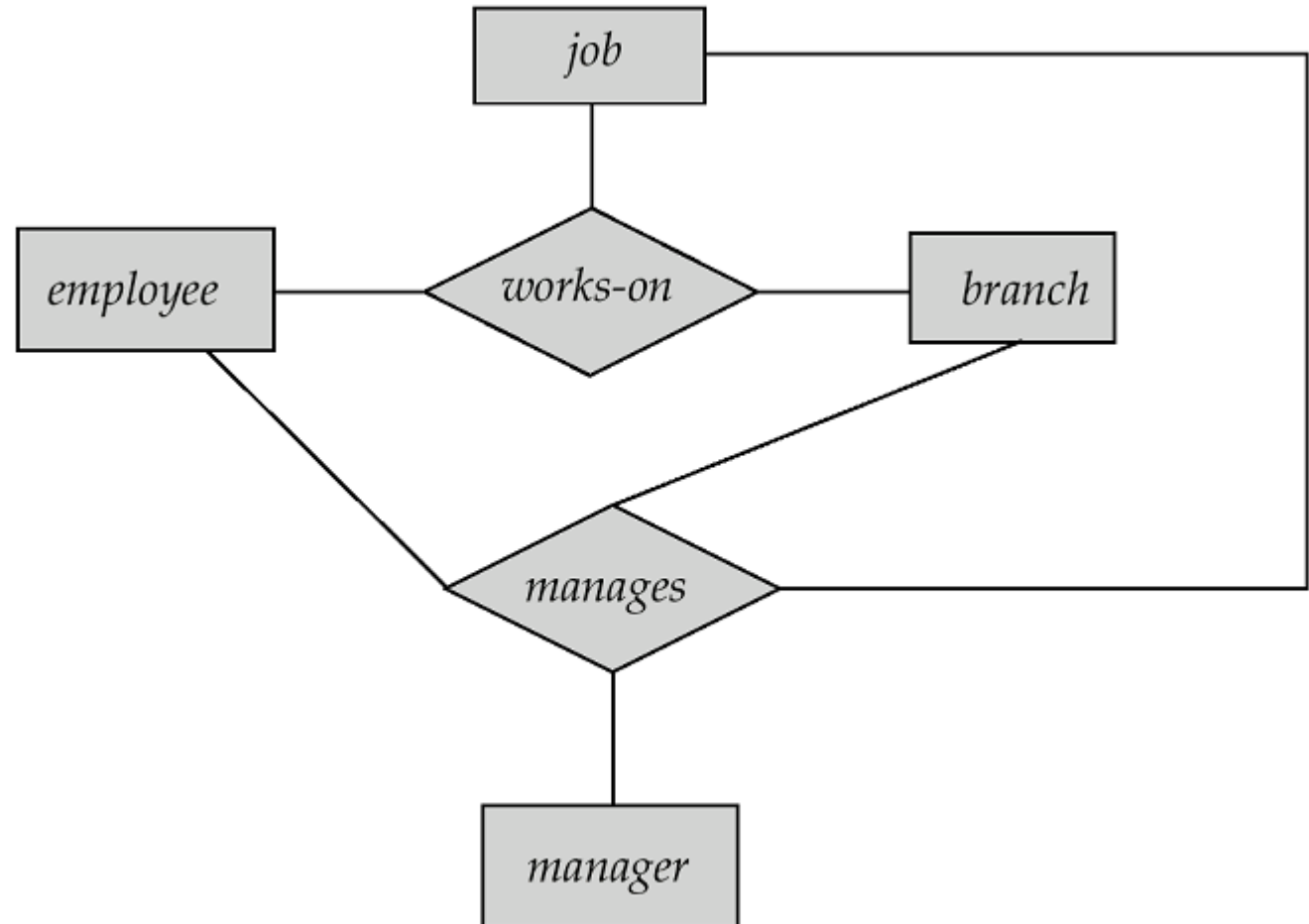
Do not directly connect

1. Entity Set to Entity Set
2. Relationship Set to Relationship Set



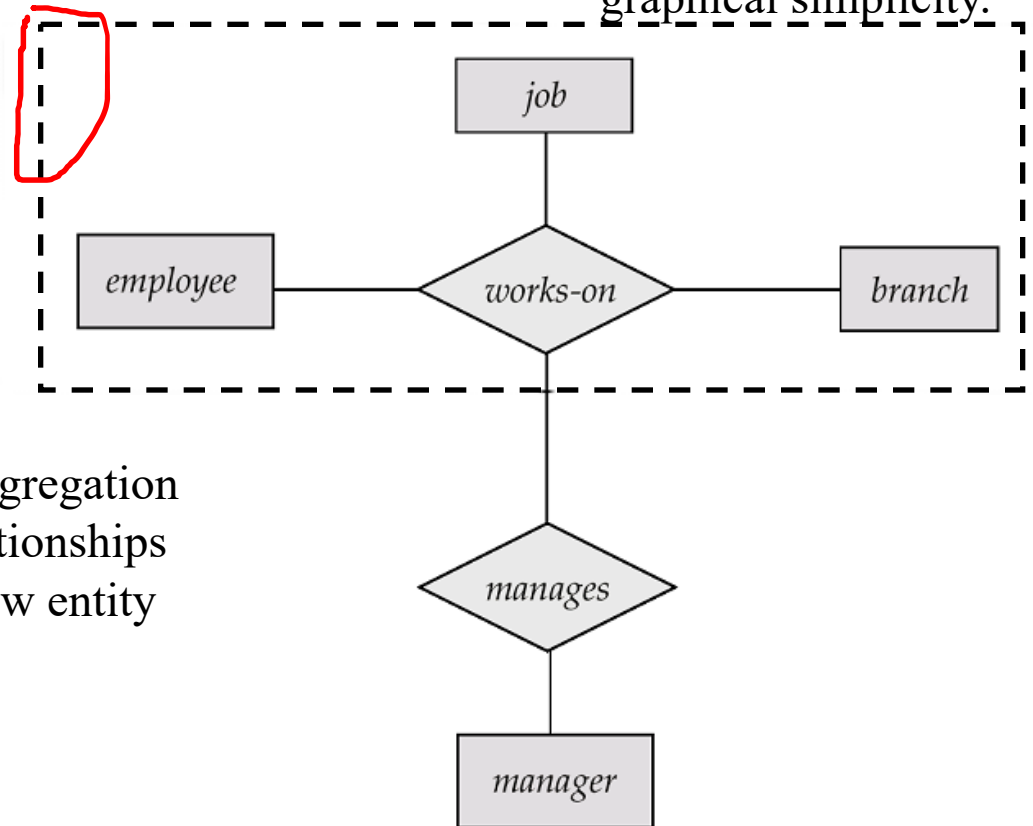
Aggregation Cont.

Note that the attributes are omitted for graphical simplicity.



A Simplify Expression - Aggregation

Note that the attributes are omitted for graphical simplicity.

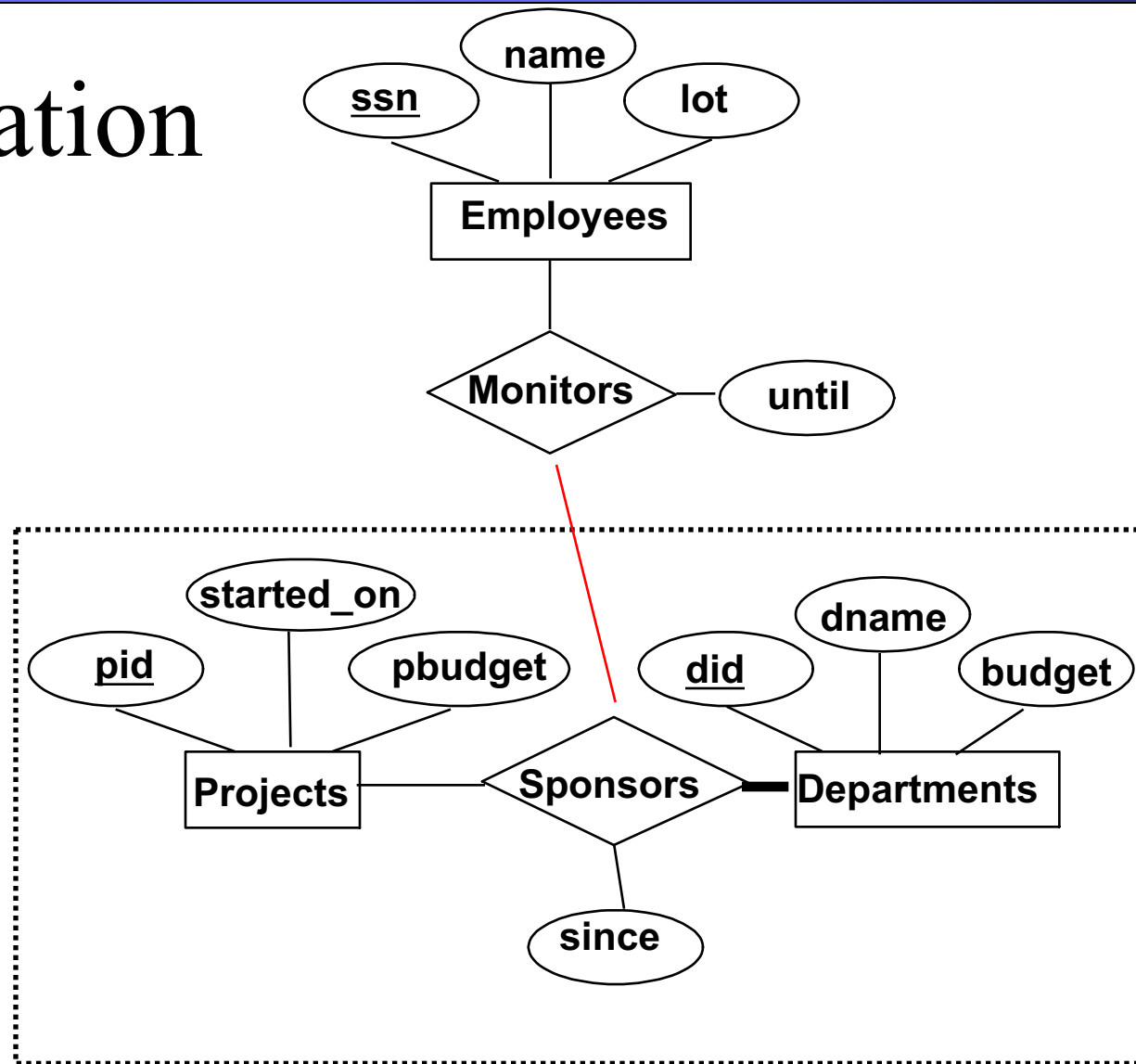


We can eliminate this redundancy via aggregation

- Allows relationships between relationships
- Abstraction of relationship into new entity

Aggregation

- Used when we have to model a relationship involving (entity sets and) a *relationship set*.
 - Aggregation allows us to treat a relationship set as an entity set for purposes of participation in (other) relationships.

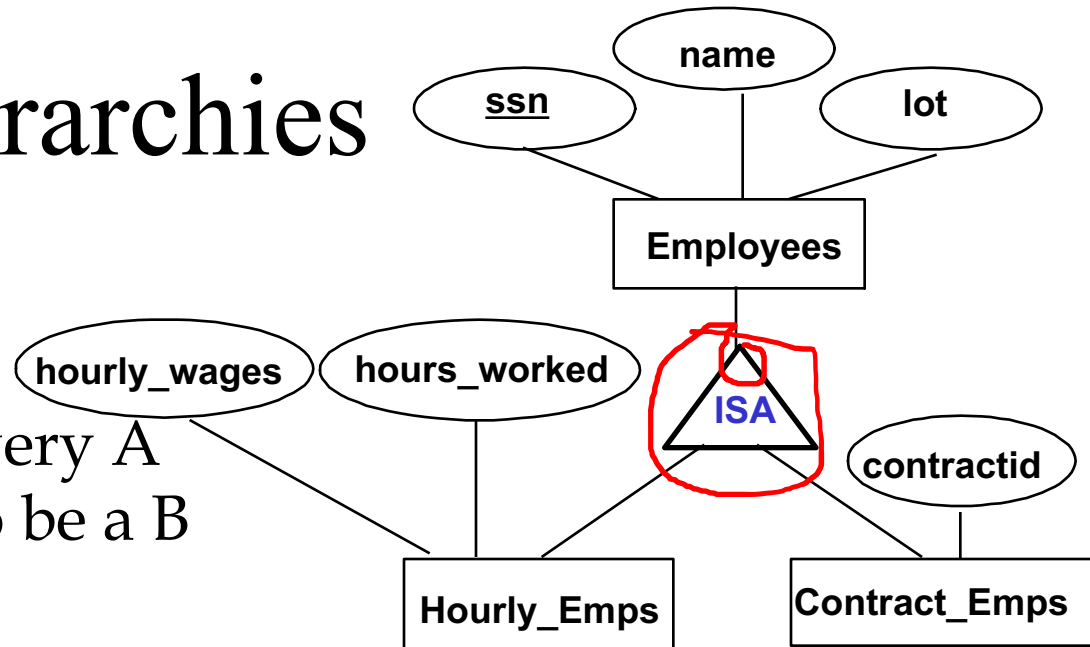


ISA ('is a') Hierarchies

- ❖ As in C++, or other PLs, attributes are inherited.
- ❖ If we declare A **ISA** B, every A entity is also considered to be a B entity.

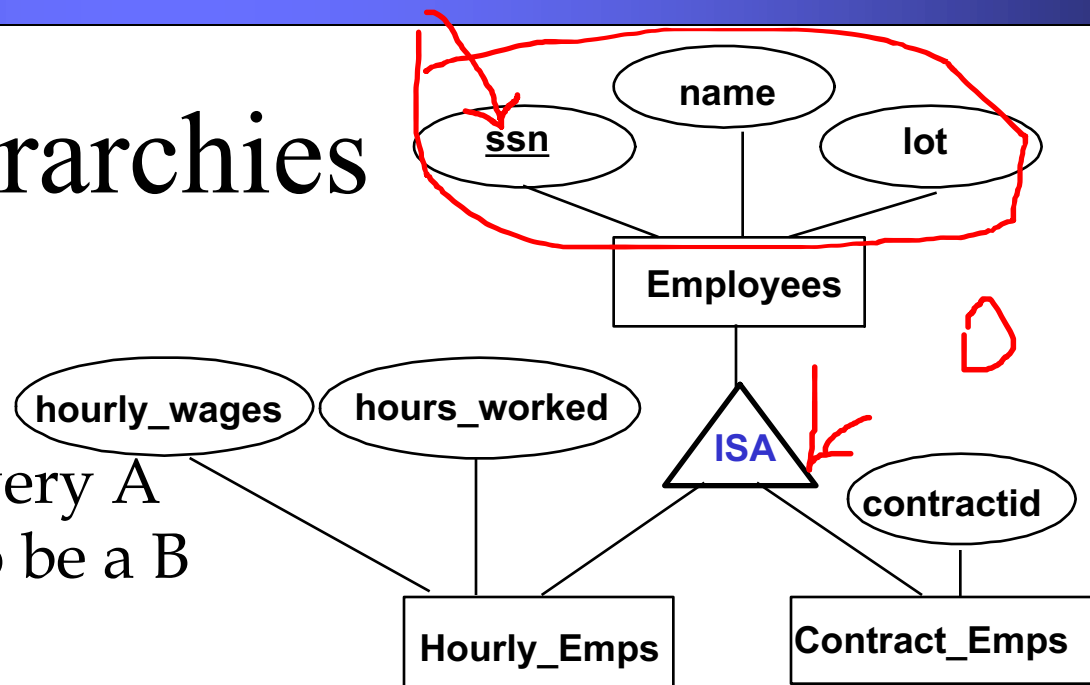
ISA ('is a') Hierarchies

- ❖ As in C++, or other PLs, attributes are inherited.
- ❖ If we declare A **ISA** B, every A entity is also considered to be a B entity.



ISA ('is a') Hierarchies

- ❖ As in C++, or other PLs, attributes are inherited.
- ❖ If we declare A **ISA** B, every A entity is also considered to be a B entity.



- Reasons for using ISA:
 - To add descriptive attributes specific to a subclass.
 - To identify entities that participate in a relationship.
- *Overlap constraints*: Can Joe be an Hourly_Emps as well as a Contract_Emps entity?
- *Covering constraints*: Does every Employees entity also have to be an Hourly_Emps or a Contract_Emps?

ER Design Decisions

- The use of an attribute or entity set to represent an object.
- Whether a real-world concept is best expressed by an entity set or a relationship set.
- The use of a ternary relationship versus a pair of binary relationships.
- The use of a strong or weak entity set.
- The use of aggregation – can treat the aggregate entity sets as a single unit without concern for the details of its internal structure.