Topic
Enhanced Database Modelling and Crow's Feet Notation

Outline

- ER model
  - Overview
  - Entity types
    - Attributes, keys
  - Relationship types
  - Weak entity types
- EER model
  - Subclasses
  - Specialization/Generalization
- Schema Design
  - Single DB
  - View integration in IS
- uses Integration DEFinition for Information Modeling (IDEF1X) notation in ERwin

Uses Crows feet notation for ER Diagrams

- This is an alternative to the diamond representation of relationships.
- Diamond icons are replaced with lines, simplifying the ER schema.
- In ERwin , select IE -- “Information Engineering” Notation

Uses Crows feet notation for ER Diagrams

- Intuition

means “Entity”

means “Identifying relationship” (one or zero to many)

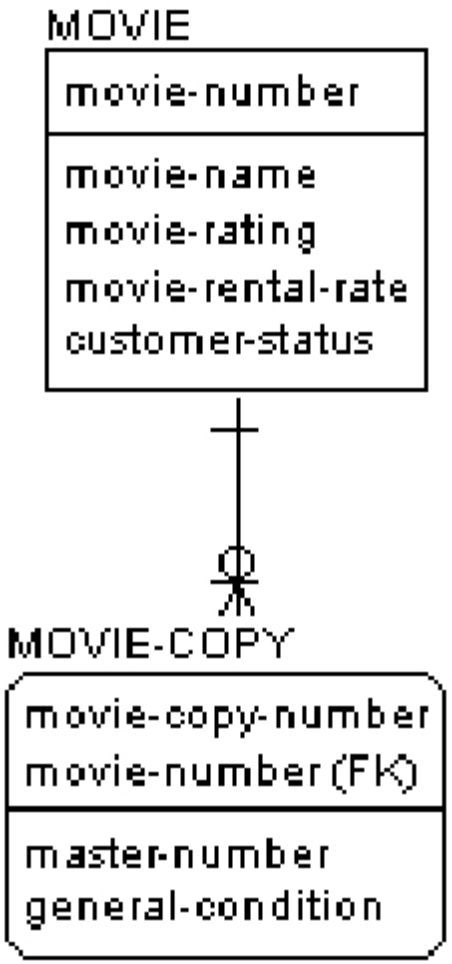
means “Many-to –many relationship”

means “Non-identifying relationship” (one or zero to many)

Uses Crows feet notation for ER Diagrams in ERwin

An identifying relationship is a relationship between two entities in which an instance of a child entity is identified through its association with a parent entity, which means the child entity is dependent on the parent entity for its identify and cannot exist without it. In an identifying relationship, one instance of the parent entity is related to multiple instances of the child.

In IE notation, ERwin draws an identifying relationship line as a solid line with crows feet.

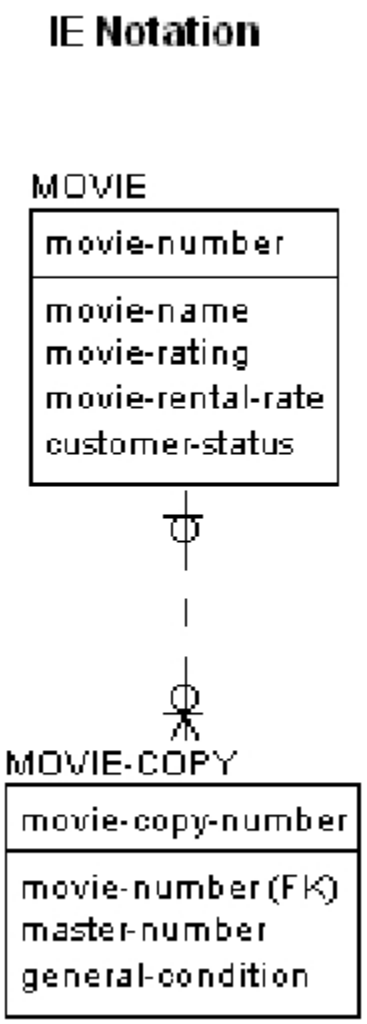


Uses Crows feet notation for ER Diagrams in ERwin

A non-identifying relationship is a relationship between two entities in which an instance of the child entity is not identified through its association with a parent entity, which means the child entity is not dependent on the parent entity for its identify and can exist without it. In a non-identifying relationship, one instance of the parent entity is related to multiple instances of the child.

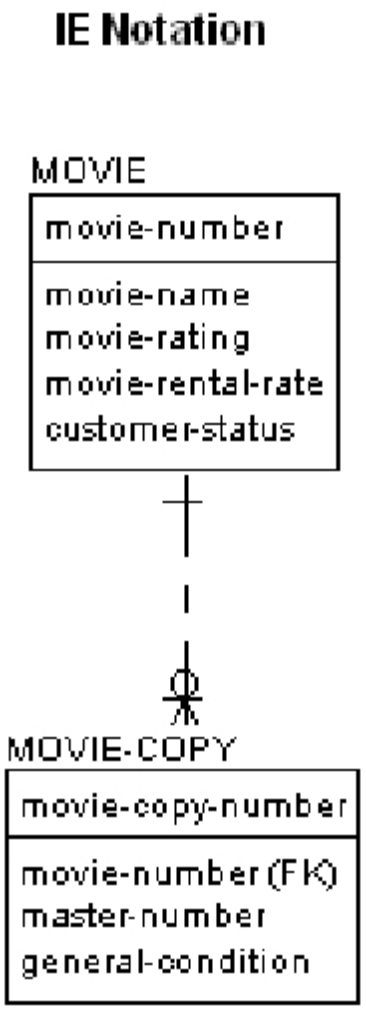
Uses Crows feet notation for ER Diagrams in ERwin

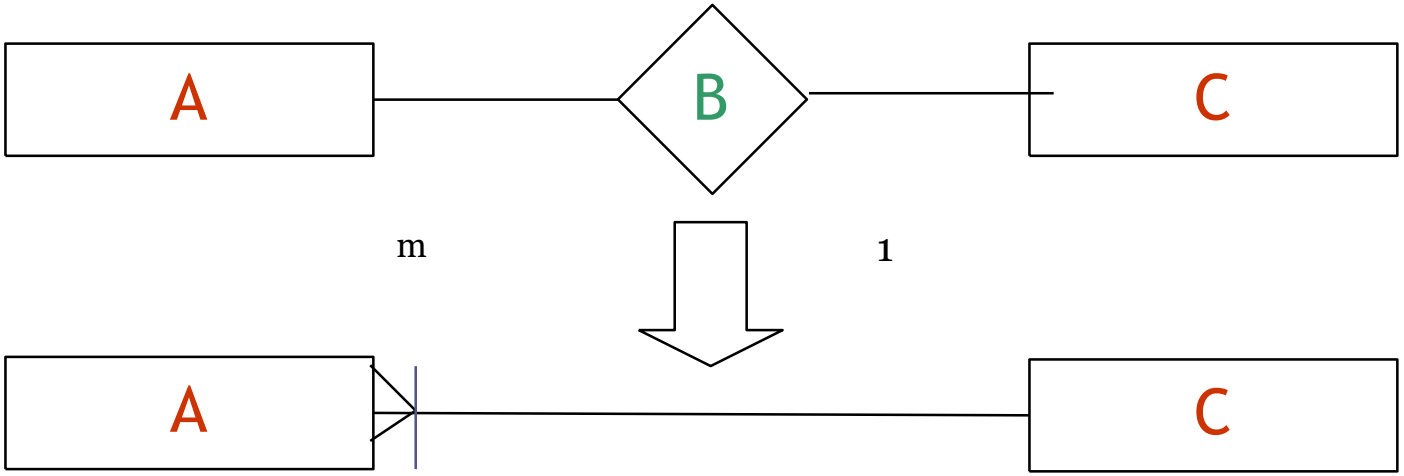
In an optional non-identifying relationship, the attributes that are migrated into the non-key area of the child entity are not required in the child entity. Therefore, nulls are allowed in the foreign key.



Uses Crows feet notation for ER Diagrams in ERwin

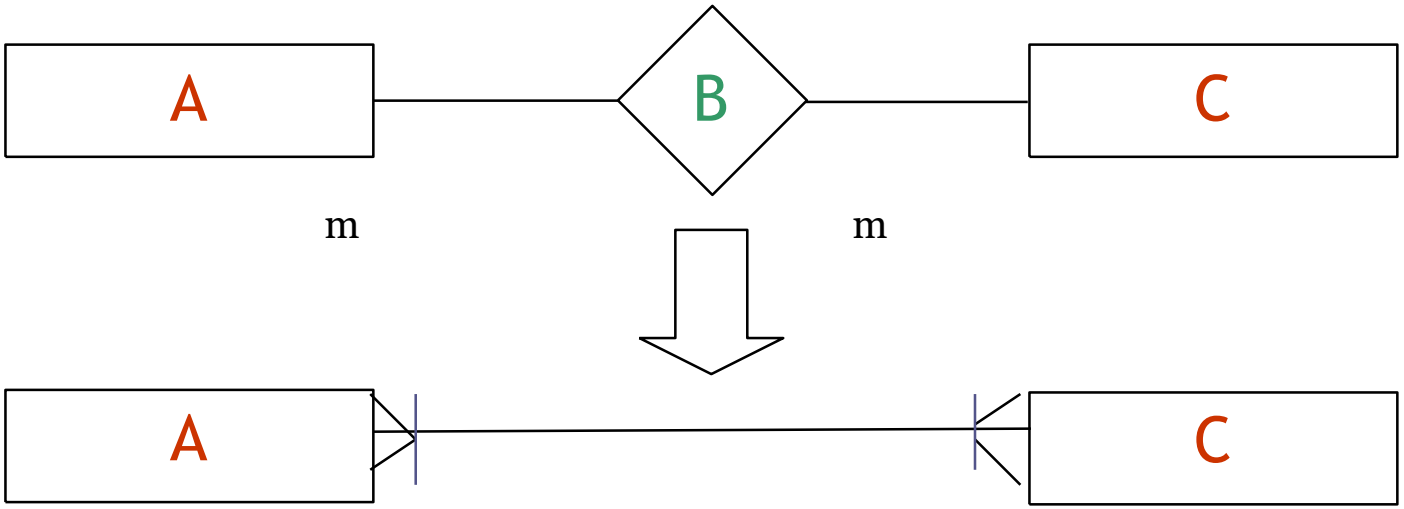
In a mandatory non-identifying relationship, the attributes that are migrated into the non-key area of the child entity are required in the child entity. Therefore, the foreign key cannot be null.





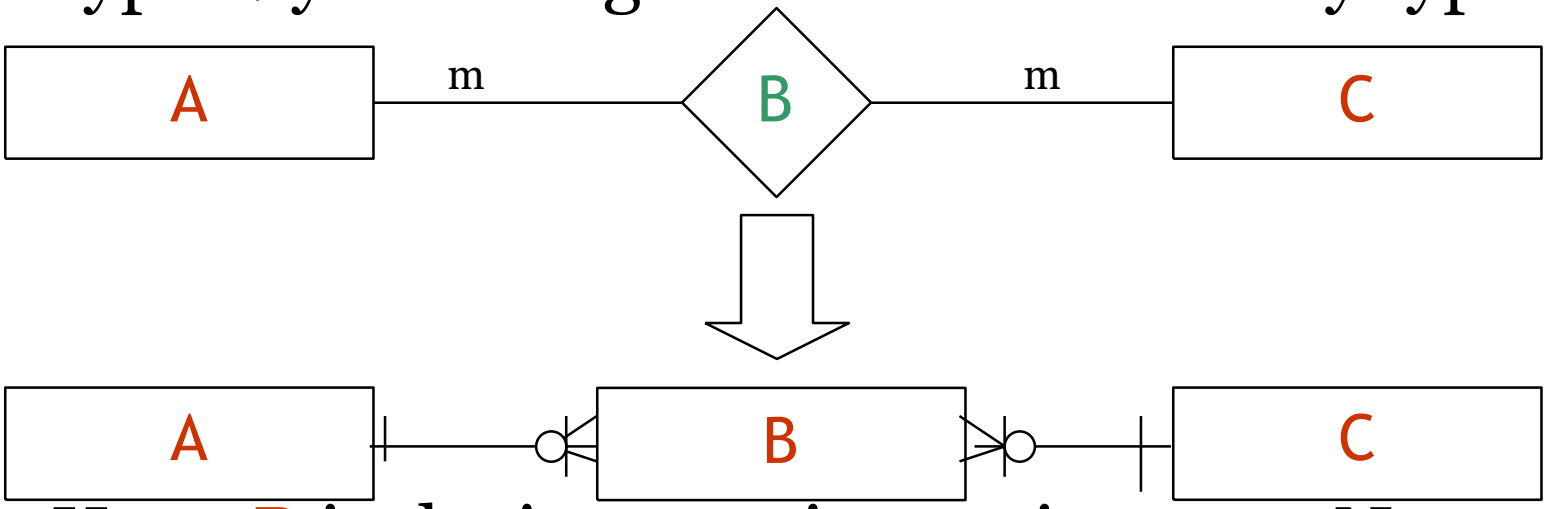
- The crow can be seen as a pictorial representation of "many".
  - Each instance of the entity type A is associated with 0 or 1 instances of the entity type C.
  - Each instance of the entity type C is associated with 0 to many instances of the entity type A.

Many-To-Many Relationship



- An instance of the entity type A is associated with possibly several instances of the entity type C. An instance of the entity type C is associated with possibly several instances of the entity type A.

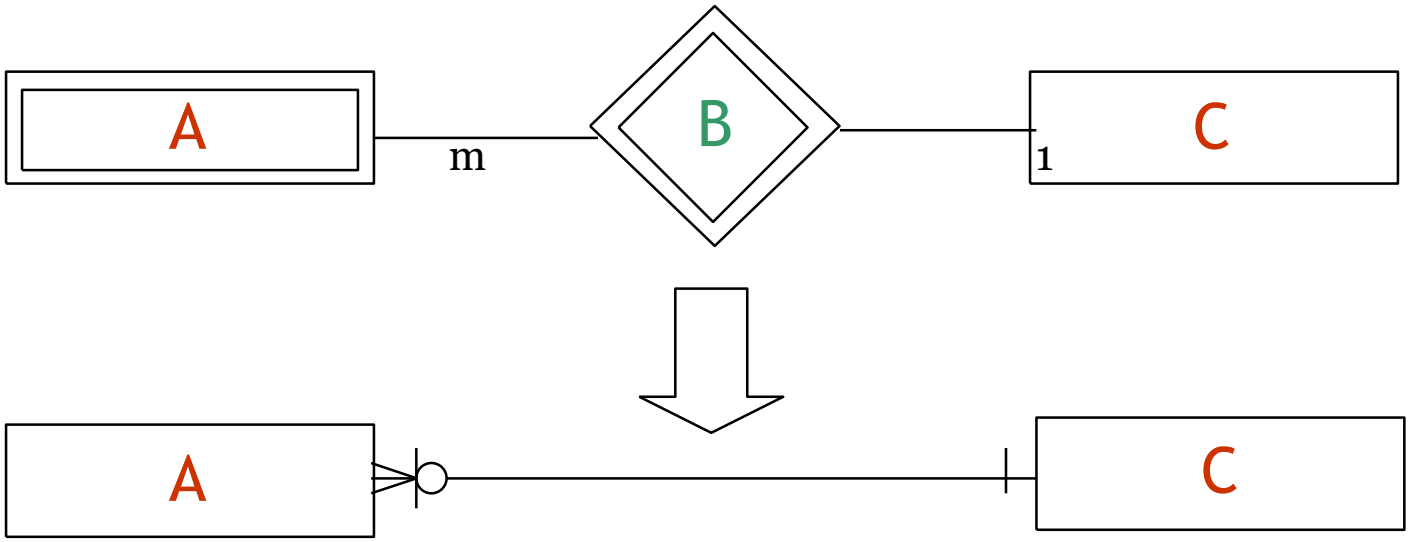
- Often many-to-many relationship types are resolved to two many-to-one relationship types by inserting an intersection entity type.



- Here, B is the intersection entity type. Note, it needs keys!
- This makes the conversion to tables easier, but can confuse the logical design.

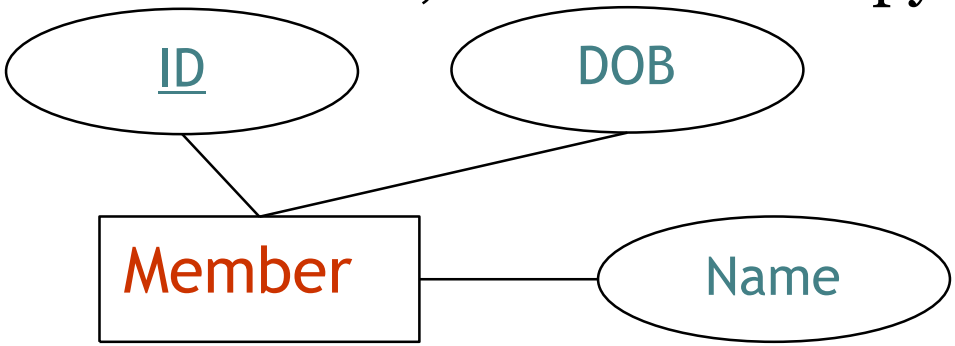
Dependence

- An entity type that *borrows* a key is dependent.
- Needed for weak entity types

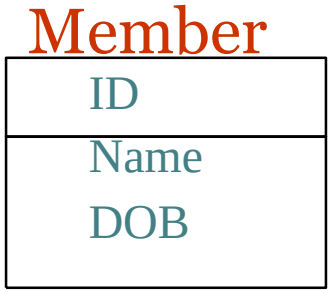


Reducing Clutter on Entity Types

- In diamond notation, attributes occupy much space



- Using ERwin, can extend entity type with attributes



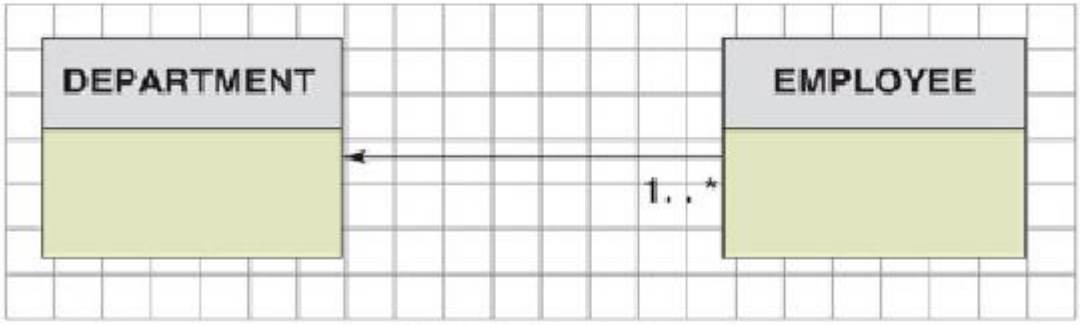
- Note: have lost ability to model multi-valued, derived, and composite attributes explicitly.

Maximum and minimum cardinality

Once Maximum and Minimum cardinality has been figures out Between the entities – then they can be notated in whatever design Tool you're using

(a) Original E-R Model

(b) Crow's Foot Version (ERwin)



(c) Visio Version

Just for reference - Data modelling notation: Erwin

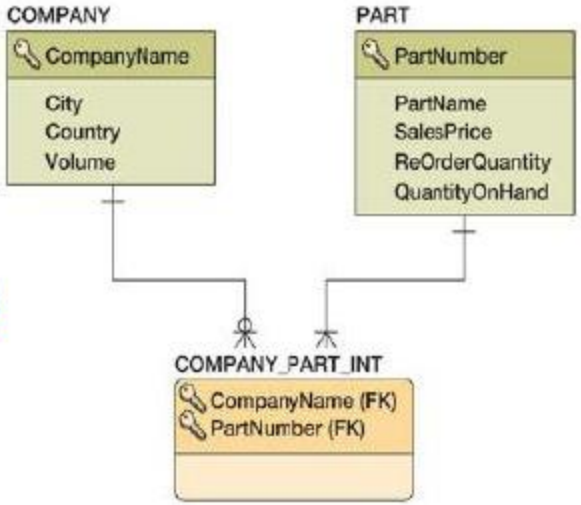
ERwin Symbol Use	Meaning
Oval with hash mark	0 or 1 entities are allowed
Hash mark alone	Exactly 1 entity is allowed
Hash mark with crow's foot	1 or more entities are allowed
Oval, hash mark, and crow's foot	0, 1, or more entities are allowed

Note: Solid line ____ versus Dashed line ---- in Erwin = identifying versus non identifying relationship between two entities

Create relationships: n:m relationships – intersection table

- The solution is to create an **intersection table** that stores data about the corresponding rows from each entity
- The intersection table consists only of the primary keys of each table which form a composite primary key
- Each table's primary key becomes a foreign key linking back to that table

COMPANY_PART_INT (*CompanyName*, *PartNumber*)



Drawing the ERD with what you've learnt so far…

Read the specification and make sure your understand it

Figure out what your entities are e.g. sales, book title, authors

Figure out which ones are related (don't worry yet about the cardinality – helps to put the word in (e.g. "has")

Figure out the attributes of each entity

Identify the primary key for each entity

Any foreign keys?

Identify the cardinality of the relationship

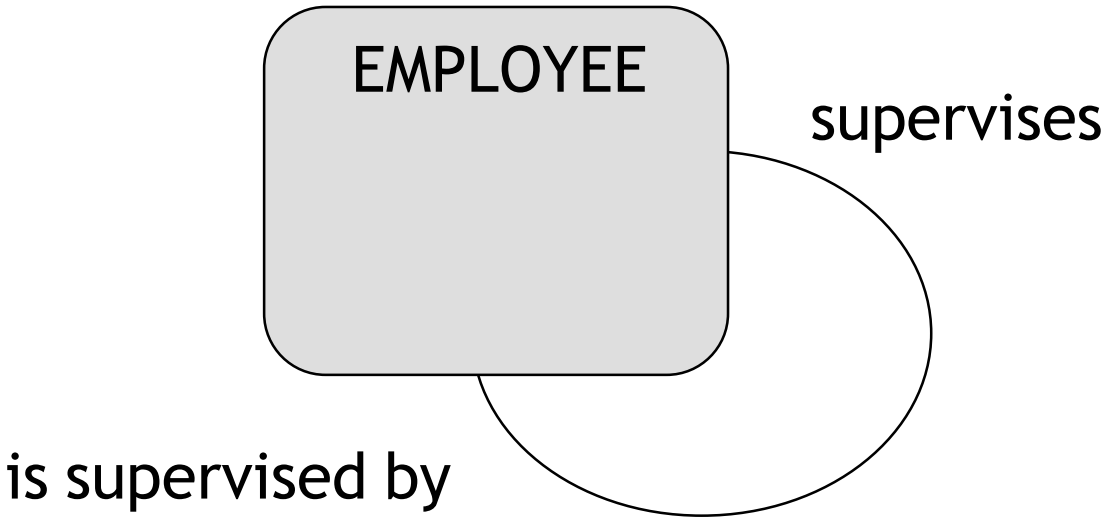
Discussion

A furniture company needs to develop a database to store information about its customers and sales

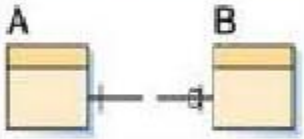
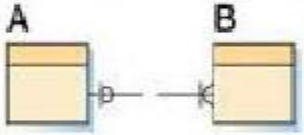
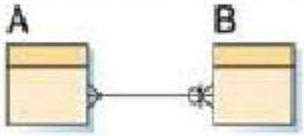
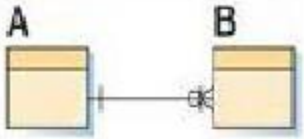
Customer details need to be captured, such as name, address.
The company wants to be able to report on what their sales – what each customer bought (i.e. what furniture items). They would also like to know what the items are.

Try and sketch out the entities involved.

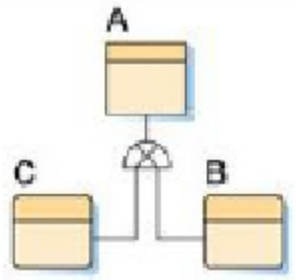
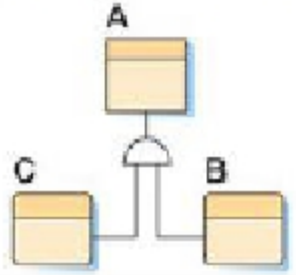
Recursive relationships



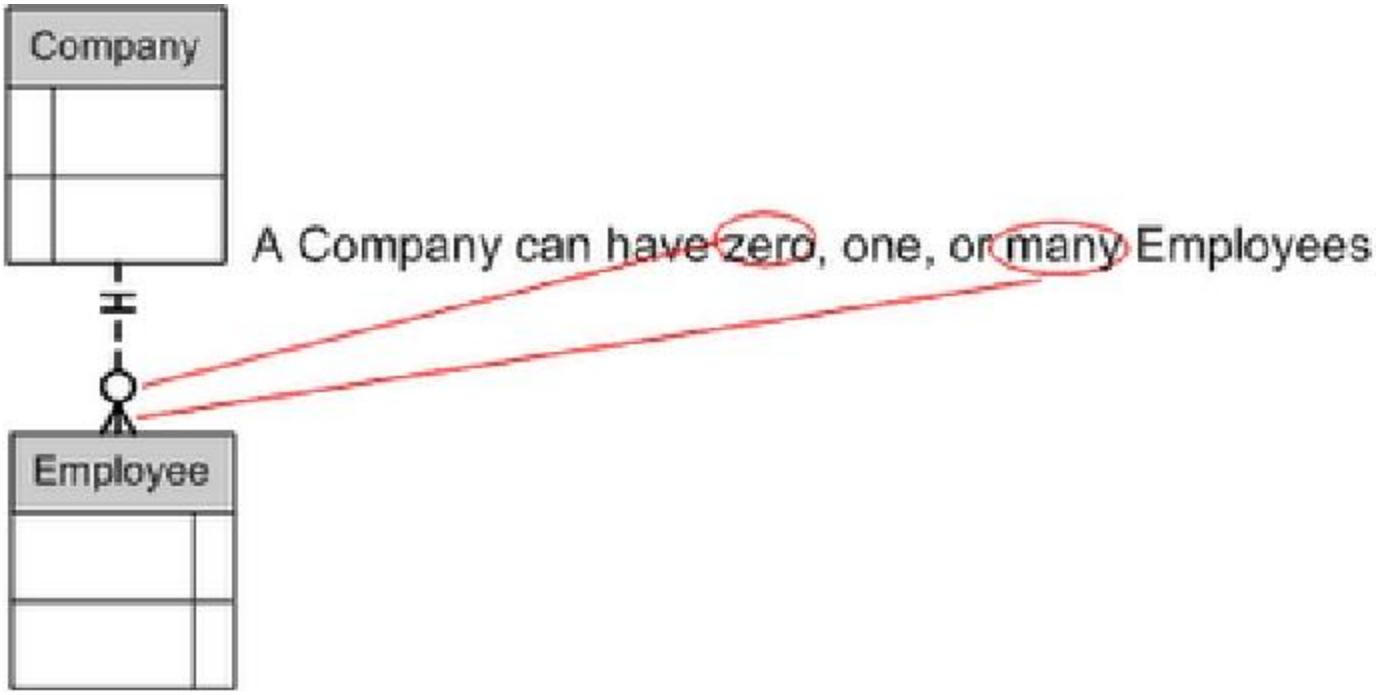
For reference.. Summary of Erwin symbols

DEPARTMENT DepartmentName BudgetCode OfficeNumber	DEPARTMENT entity; DepartmentName is identifier; BudgetCode and OfficeNumber are attributes.
	1:1, nonidentifying relationship. A relates to zero or one B; B relates to exactly one A. Relationship shown using a dashed line.
	1:N, nonidentifying relationship. A relates to one or many Bs; B relates to zero or one A. Relationship shown using a dashed line.
	Many-to-many, nonidentifying relationship. Minimum cardinality must be shown in notes or annotation. Relationship shown with a solid line.
	1:N identifying relationship. A relates to zero, one, or many Bs. B relates to exactly one A. Relationship shown with a solid line. For identifying relationships, the child must always relate to exactly one parent. The parent may relate to zero, one, many, or a combination of these minimum cardinalities.

Summary of Erwin symbols (cont.)

	A is supertype, C and D are exclusive subtypes. Discriminator not shown. Relationships shown with a solid line.
	A is supertype, C and D are inclusive subtypes. Relationships shown with a solid line.

Some examples

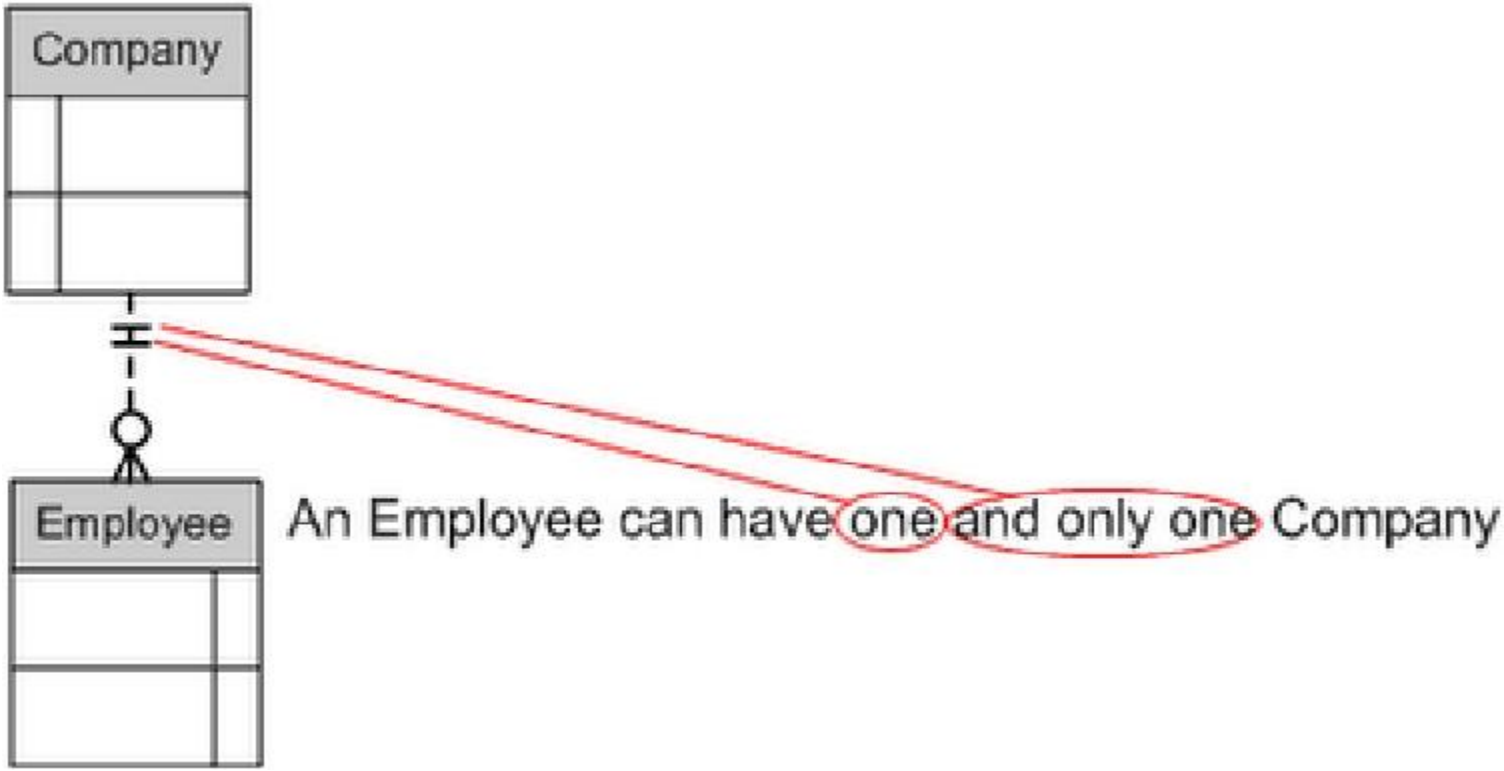


Note.. The symbols right beside the entity tell you how many of "it" can and must apply to the other entity.

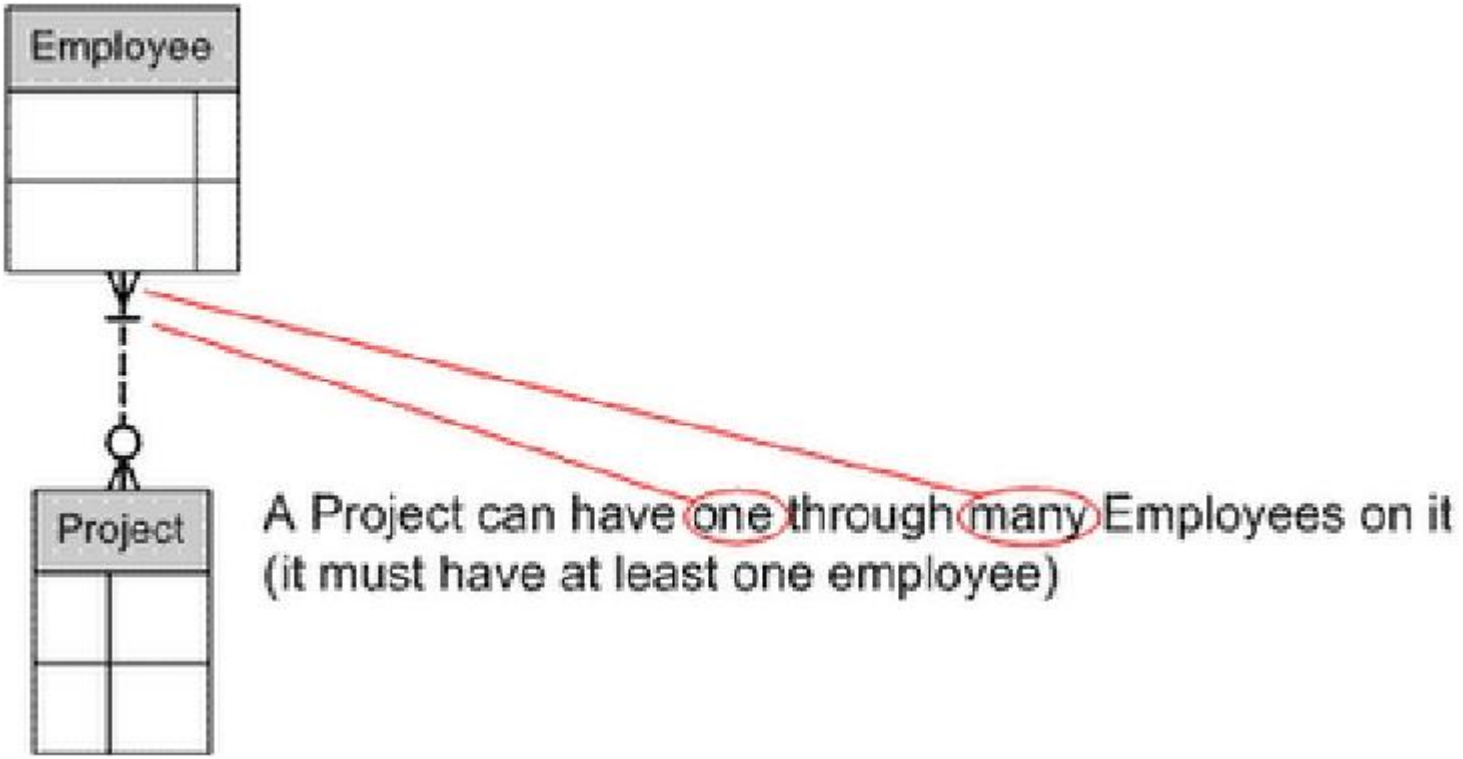
How many companies can and must an employee belong to?

(note: sometimes I I shown, sometimes I

Some examples (continued)



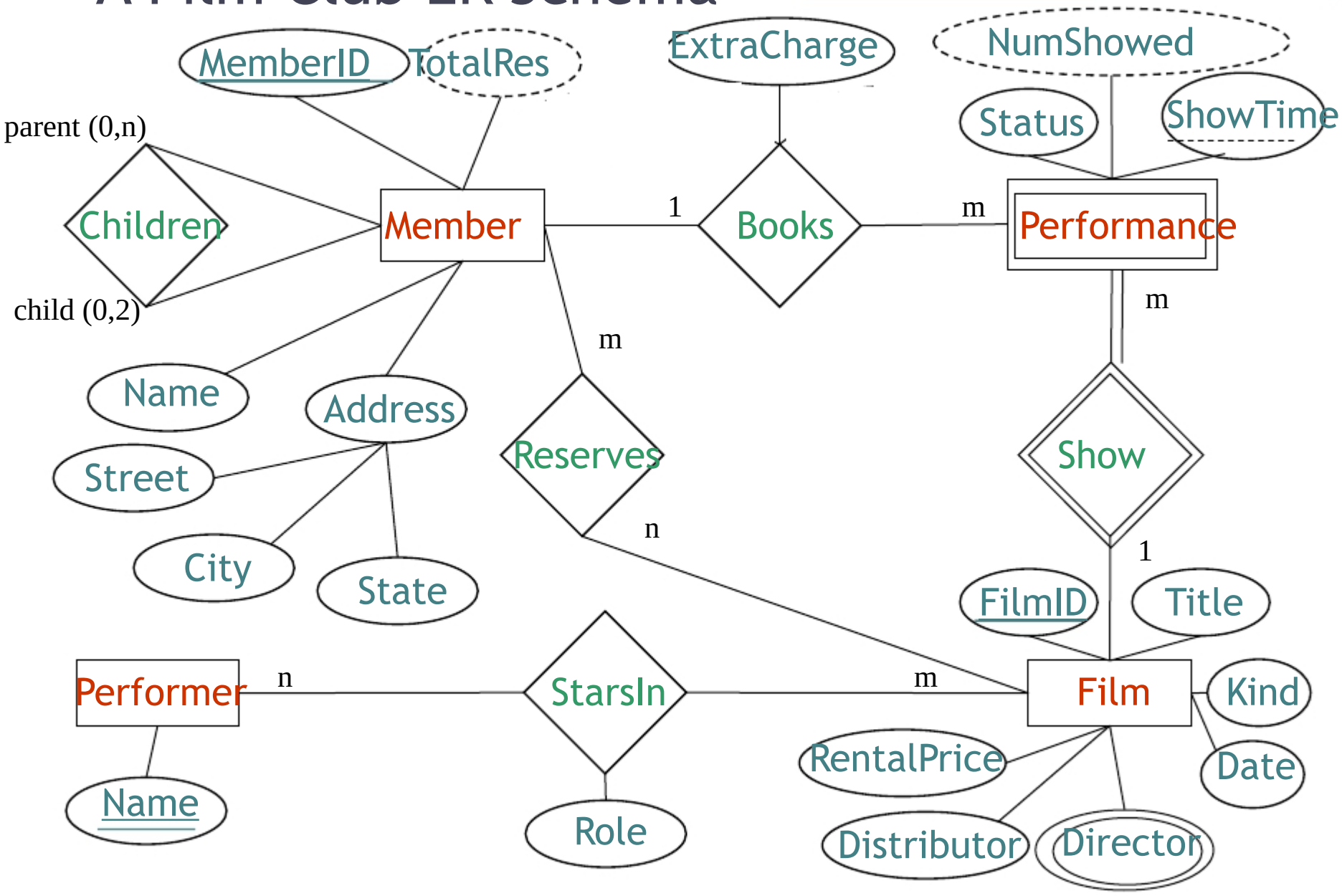
Some examples (continued)



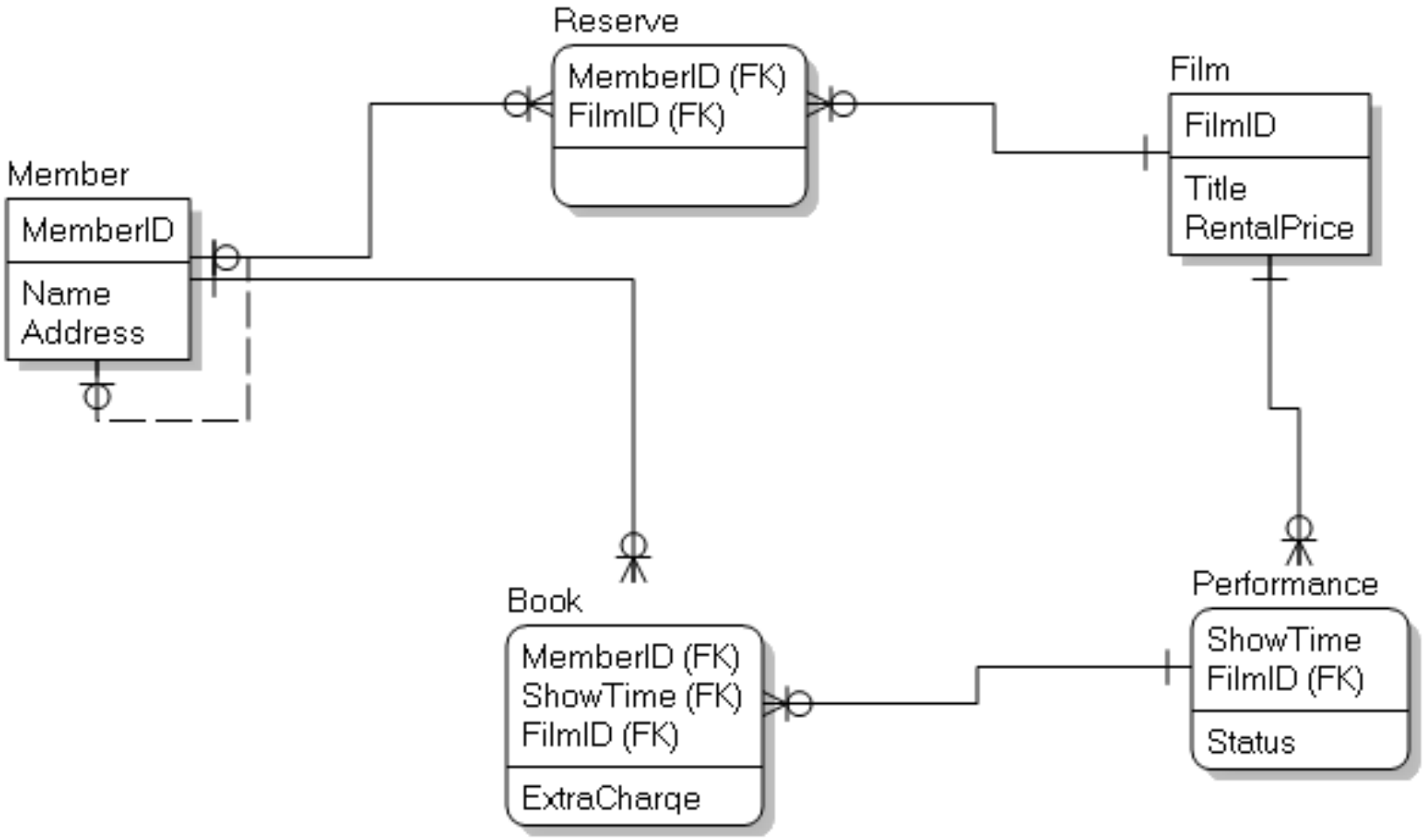
Other Notational Aspects

- Relationship types that have associated attributes must be represented with intersection entity types.
- Details differ among the various tools supporting variants of the Entity-Relationship schemas. For example,
  - Sometimes optional a dashed line is denoted with ("zero or") a circle.
  - Cardinalities can sometimes be placed at either end of a relationship arc.
  - Other icons, such as small diamonds, have specialized meanings.

A Film Club ER Schema



Same Schema convert to Erwin style



Outline

- ER model
  - Overview
  - Entity types
    - Attributes, keys
  - Relationship types
  - Weak entity types
- Schema Design
  - Single DB
  - View integration in IS
- Uses Crows feet notation for ER Diagrams in ERwin