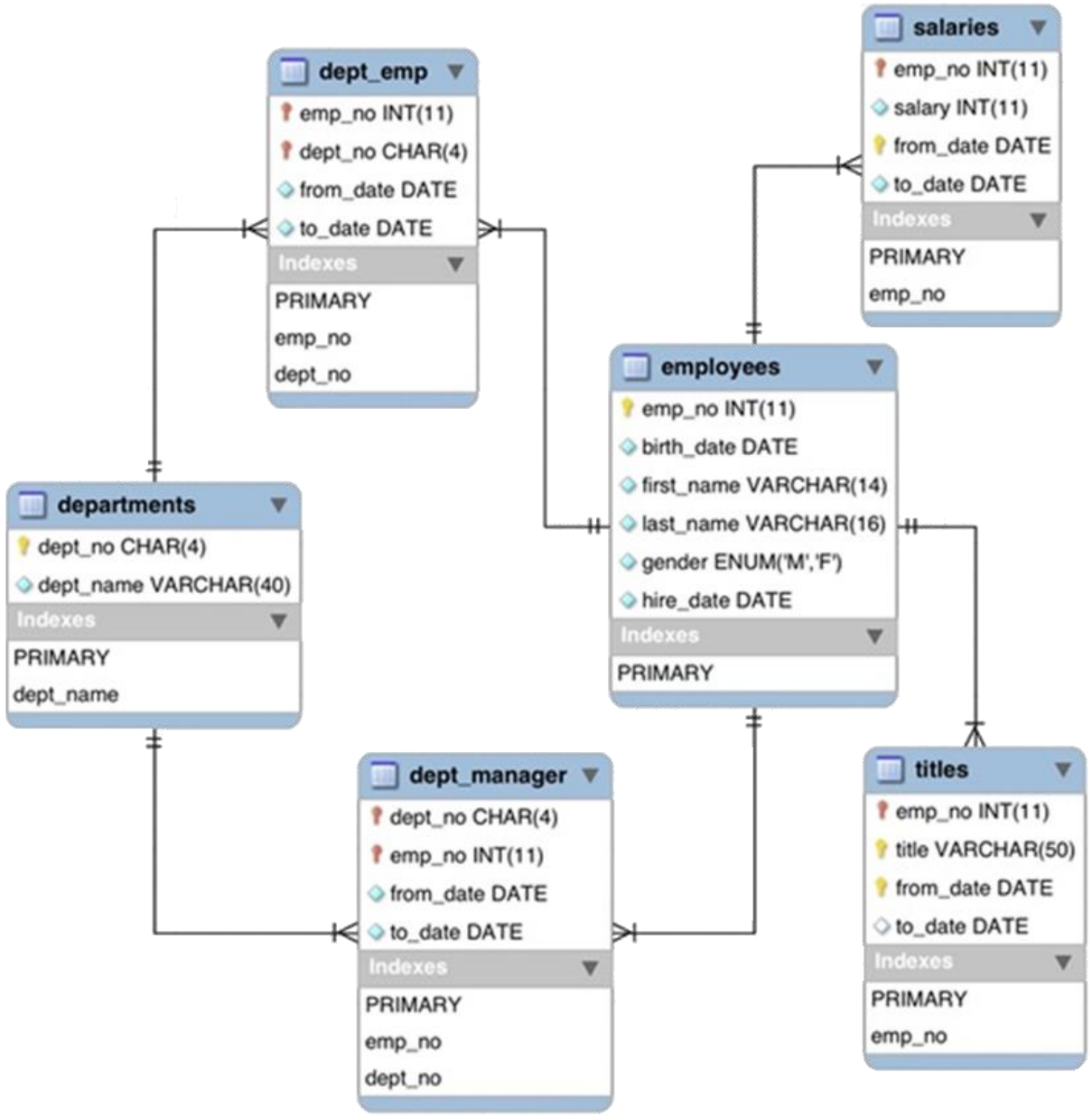


INTRODUCTION TO DATABASES USING ORACLE

420-983-VA

DATABASE DESIGN II

CONCEPTUAL MODELS

Additional Features of the E/R Model

Key Constraints

A key for an entity set **E** is a set **K** of one or more attributes such that, given any two distinct entities e_1 and e_2 in E, e_1 and e_2 cannot have identical values for each of the attributes in the key K. If K consists of more than one attribute, then it is possible for e_1 and e_2 to agree in some of these attributes, but never in all attributes.

- Every entity set must have a key, although in some cases — isa-hierarchies and “weak” entity sets, the key actually belongs to another entity set.
- There can be more than one possible key for an entity set. However, it is customary to pick one key as the “primary key,” and to act as if that were the only key.
- When an entity set is involved in an isa-hierarchy, we require that the root entity set have all the attributes needed for a key, and that the key for each entity is found from its component in the root entity set

Under E/R diagram keys are underline. There is no notation for representing the situation where there are several keys for an entity set; we underline only the primary key.

CONCEPTUAL MODELS

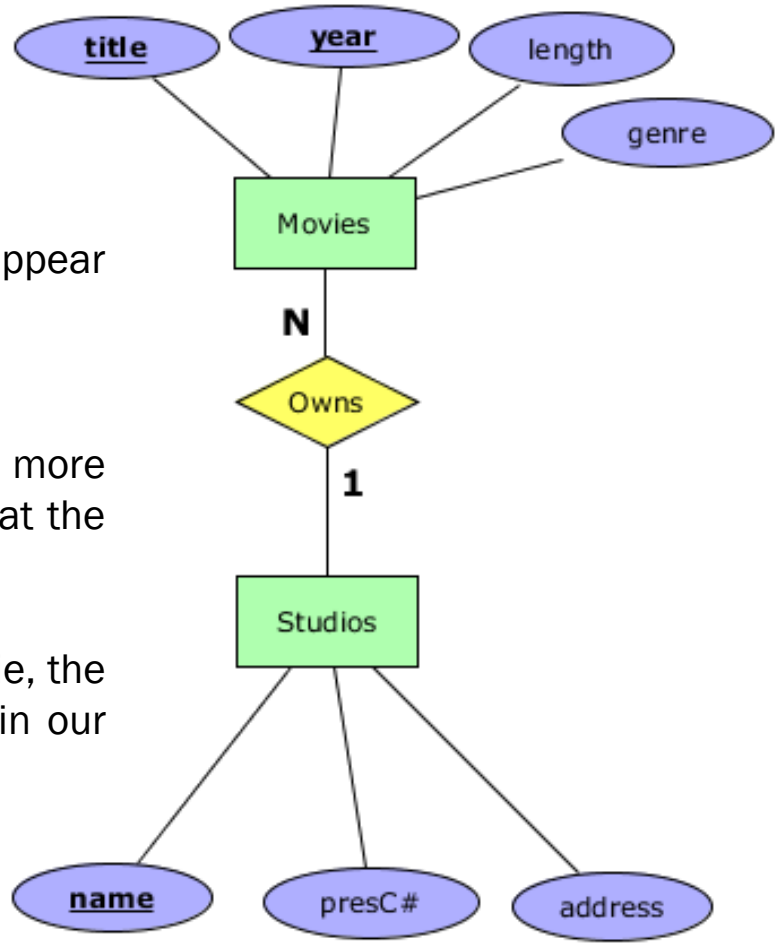
Additional Features of the E/R Model

Referential Integrity

Referential Integrity constraints say that a value appearing in one context must also appear in another.

Example: The many-to-one requirement simply says that no movie can be owned by more than one studio. It does not say that a movie must surely be owned by a studio, or that the owning studio must be present in the Studios entity set, as stored in our database.

An appropriate referential integrity constraint on relationship Owns is that for each movie, the owning studio (the entity "referenced" by the relationship for this movie) must exist in our database.



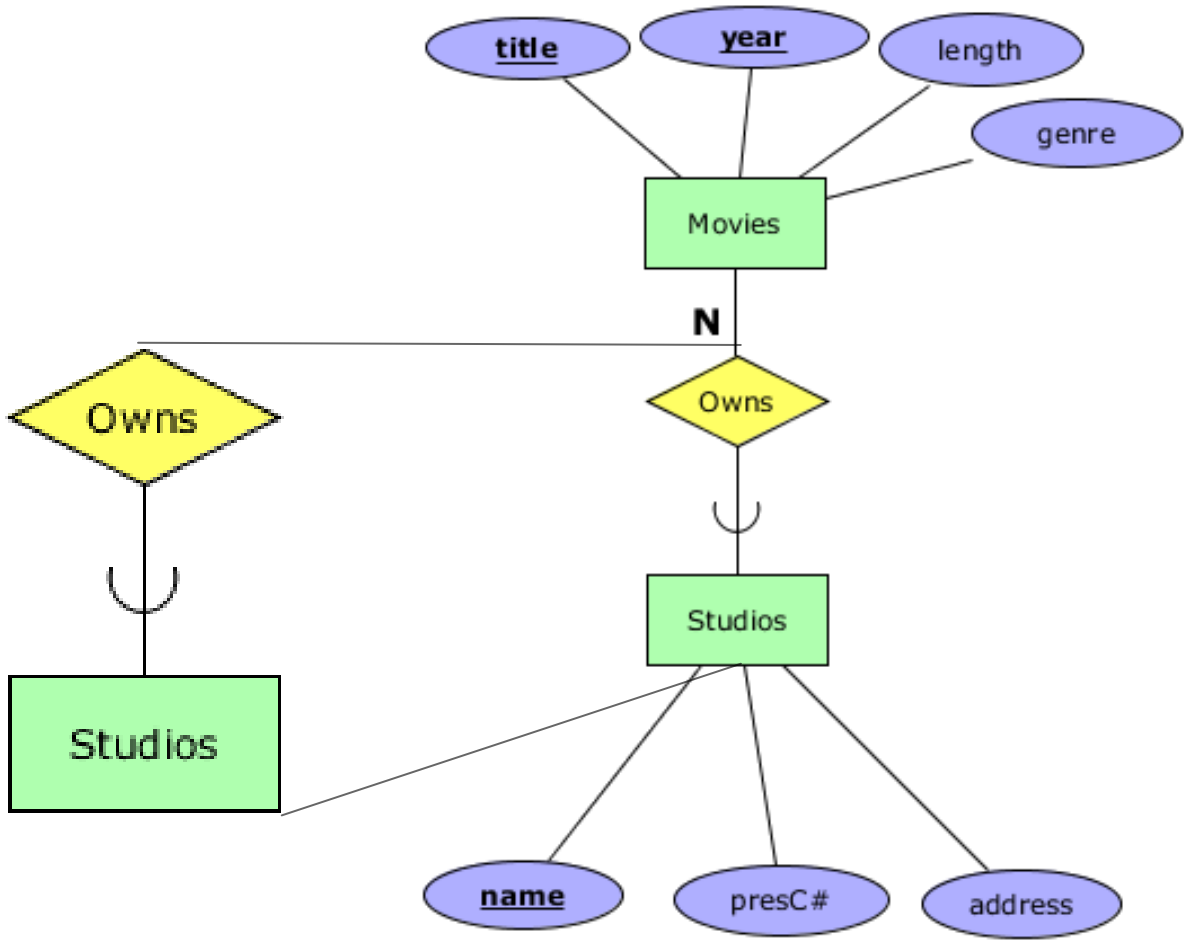
CONCEPTUAL MODELS

Additional Features of the E/R Model

Referential Integrity

Referential Integrity constraints is marked by a rounded arrow.

Denotes not only that the relationship is many-to-one but that the entity of set Studios must exists for each entity of set Movies.



CONCEPTUAL MODELS

Additional Features of the E/R Model

Degree Constraints

In the E/R model, we can attach a bounding number to the edges that connect a relationship to an entity set, indicating limits on the number of entities that can be connected to any one entity of the related entity set.

Example: A movie entity cannot be connected by relationship Stars-in to more than 10 star entities.

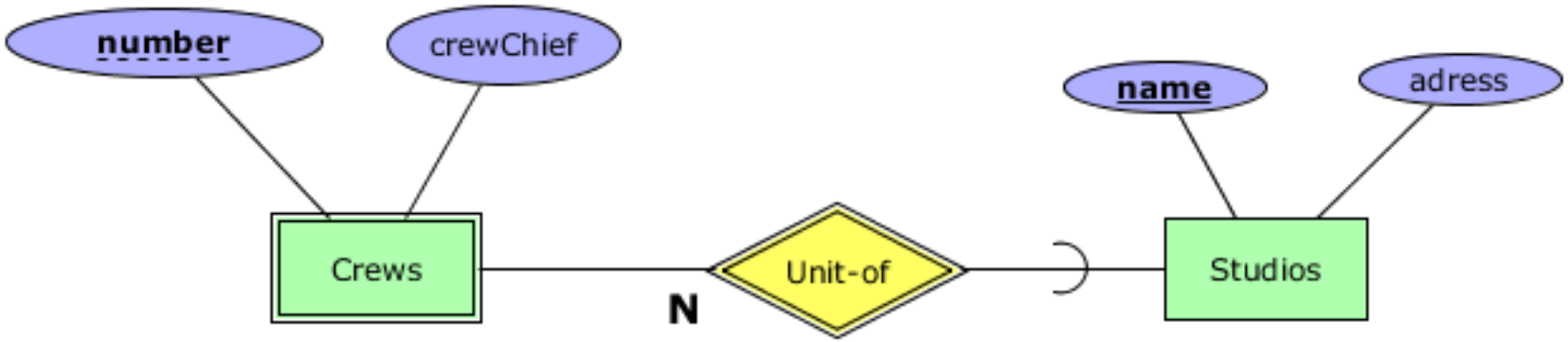
Note that <=1 is equivalent with one-to-many and =1 equivalent with the referential integrity.

CONCEPTUAL MODELS

Weak Entity Sets

It is possible for an entity set's key to be composed of attributes, some or all of which belong to another entity set. Such an entity set is called a weak entity set.

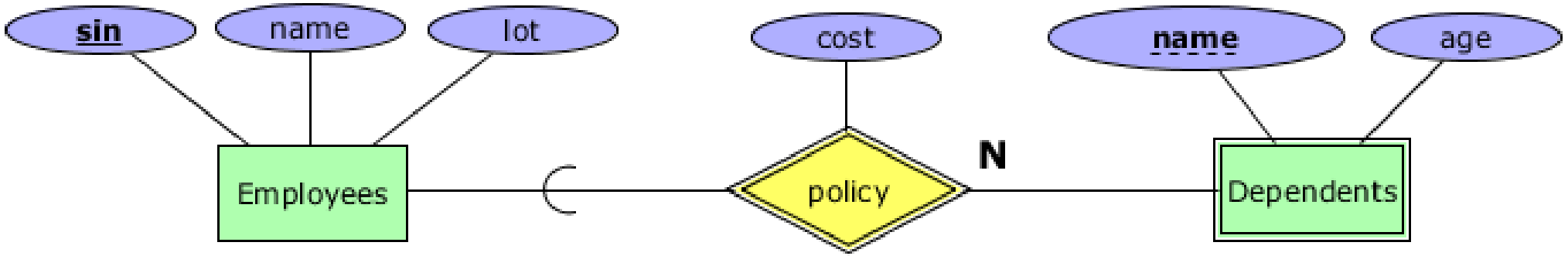
Example 1: A movie studio might have several film crews. The crews might be designated by a given studio as crew 1, crew 2, and so on. However, other studios might use the same designations for crews, so the attribute number is not a key for crews.



CONCEPTUAL MODELS

Weak Entity Sets

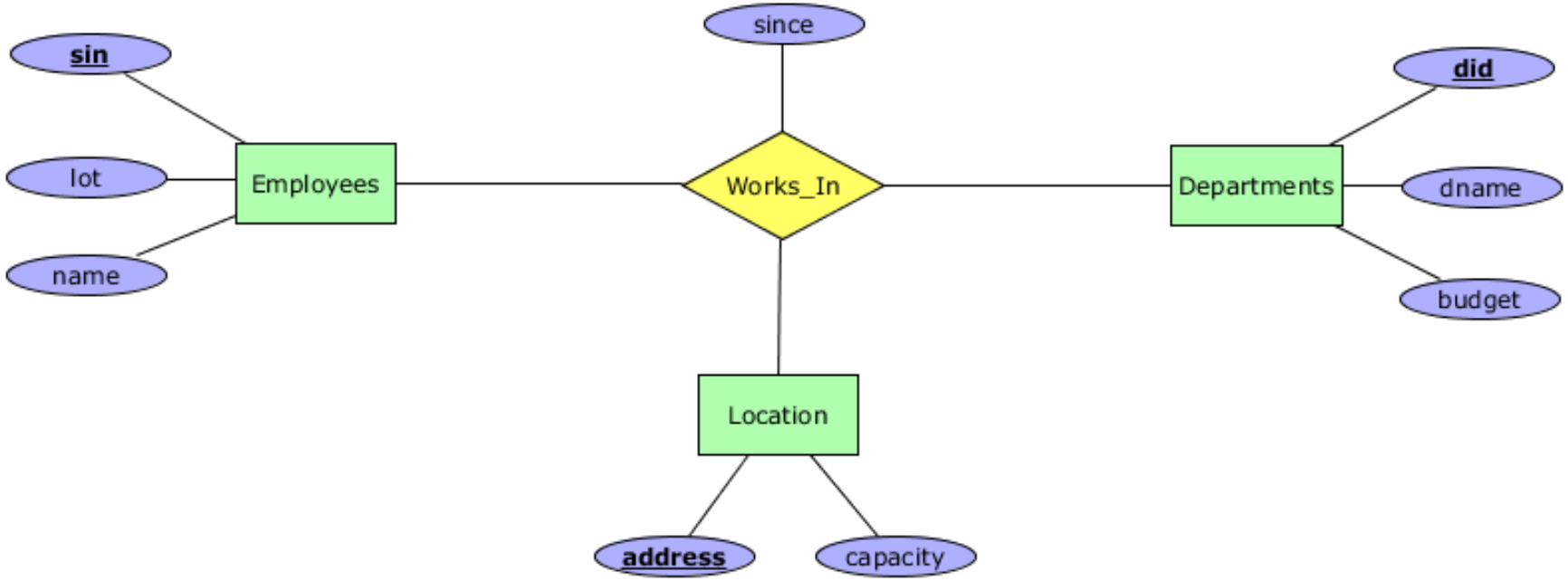
Example 2: A company insurance policy insures an employee and any dependents. For the purpose of describing an insurance policy, an Employee might or might not have a Dependent, but the Dependent must be associated with an Employee. Moreover, the Dependent cannot exist without the Employee; that is, a person cannot get insurance coverage as a dependent unless the person is a dependent of an employee. Dependent is the weak entity in the relationship "Employee–policy–Dependent".



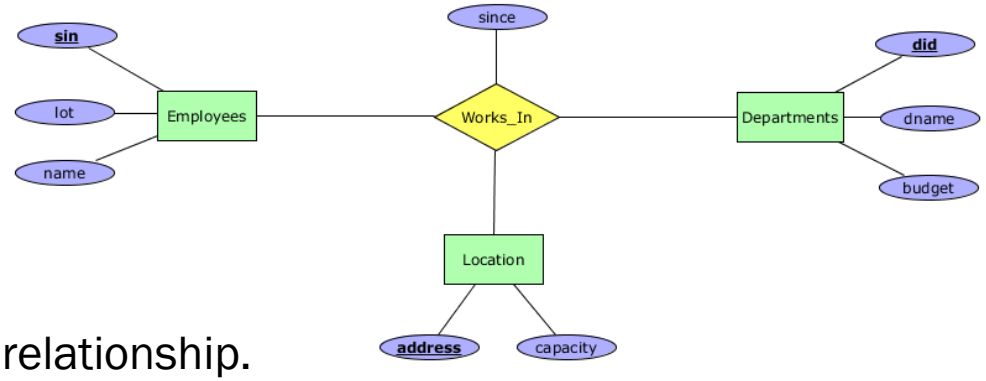
CONCEPTUAL MODELS

Weak Entity Sets

Let's review the transformation from multiway relationships to binary relationship.

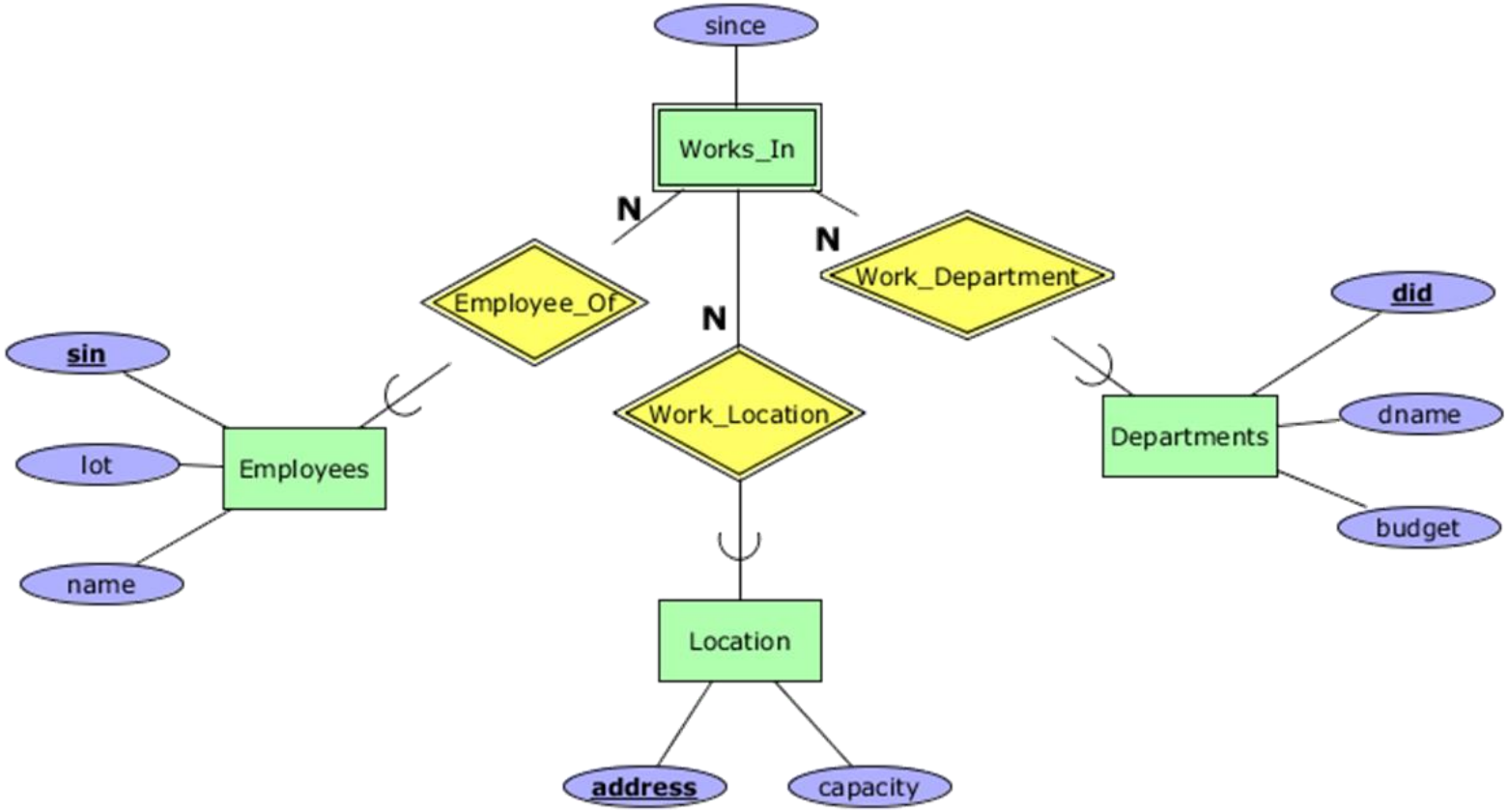


CONCEPTUAL MODELS



Weak Entity Sets

Let's review the transformation from multiway relationships to binary relationship.



CONCEPTUAL MODELS

Weak Entity Sets

Requirements for Weak Entity Sets

We cannot obtain key attributes for a weak entity set indiscriminately. Rather, if E is a weak entity set then its key consists of:

1. Zero or more of its own attributes, and

2. Key attributes from entity sets that are reached by certain many-one relationships from E to other entity sets. These many-one relationships are called supporting relationships for E, and the entity sets reached from E are supporting entity sets.

Which are the key attributes for the Works_In entity set?

CONCEPTUAL MODELS

Weak Entity Sets

We shall adopt the following conventions to indicate that an entity set is weak and to declare its key attributes.

1. If an entity set is weak, it will be shown as a rectangle with a double border. **Example**:

2. The supporting relationships will be shown as diamonds with a double border. **Example**:

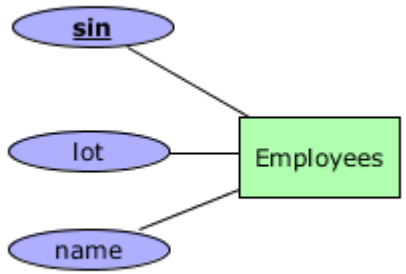
3. If an entity set supplies any attributes for its own key, then those attributes will be underlined.

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

For each non-weak entity set, we shall create a relation of the same name and with the same set of attributes. This relation will not have any indication of the relationships in which the entity set participates

E/R Model



Relational Model

Employees(sin:string, lot:integer, name: string)

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

From E/R Relationships to Relations

Relationships in the E/R model are also represented by relations. The relation for a given relationship R has the following attributes:

1. For each entity set involved in relationship R, we take its key attribute or attributes as part of the schema of the relation for R.

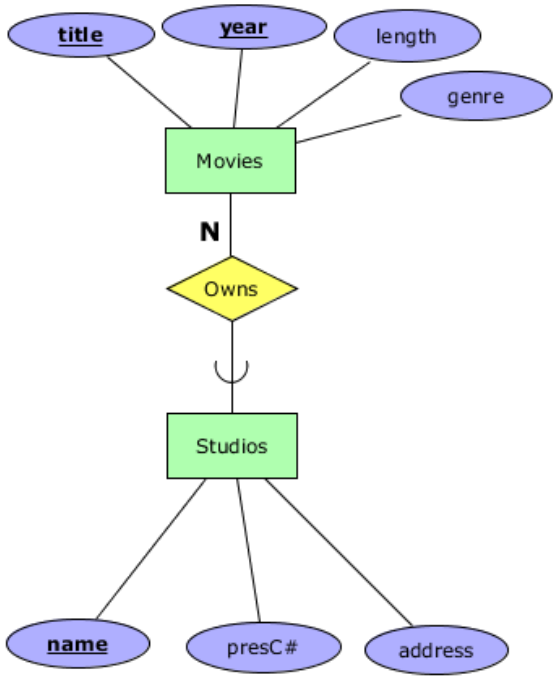
2. If the relationship has attributes, then these are also attributes of relation R.

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

From E/R Relationships to Relations

E/R Model



Relational Model

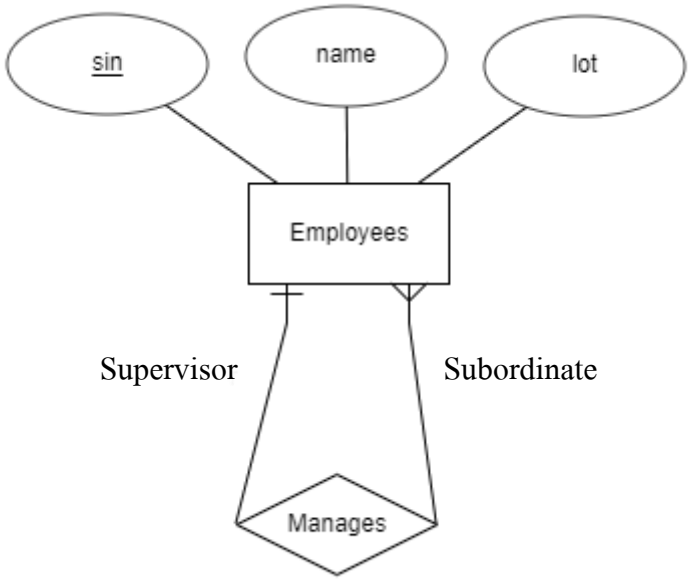
Owns(movieTitle:string, movieYear:integer, studioName: string)

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

From E/R Relationships to Relations

E/R Model



Relational Model

Manages(supervisorSIN:string, subordinateSIN:integer)

Example 2 (relationships with roles)

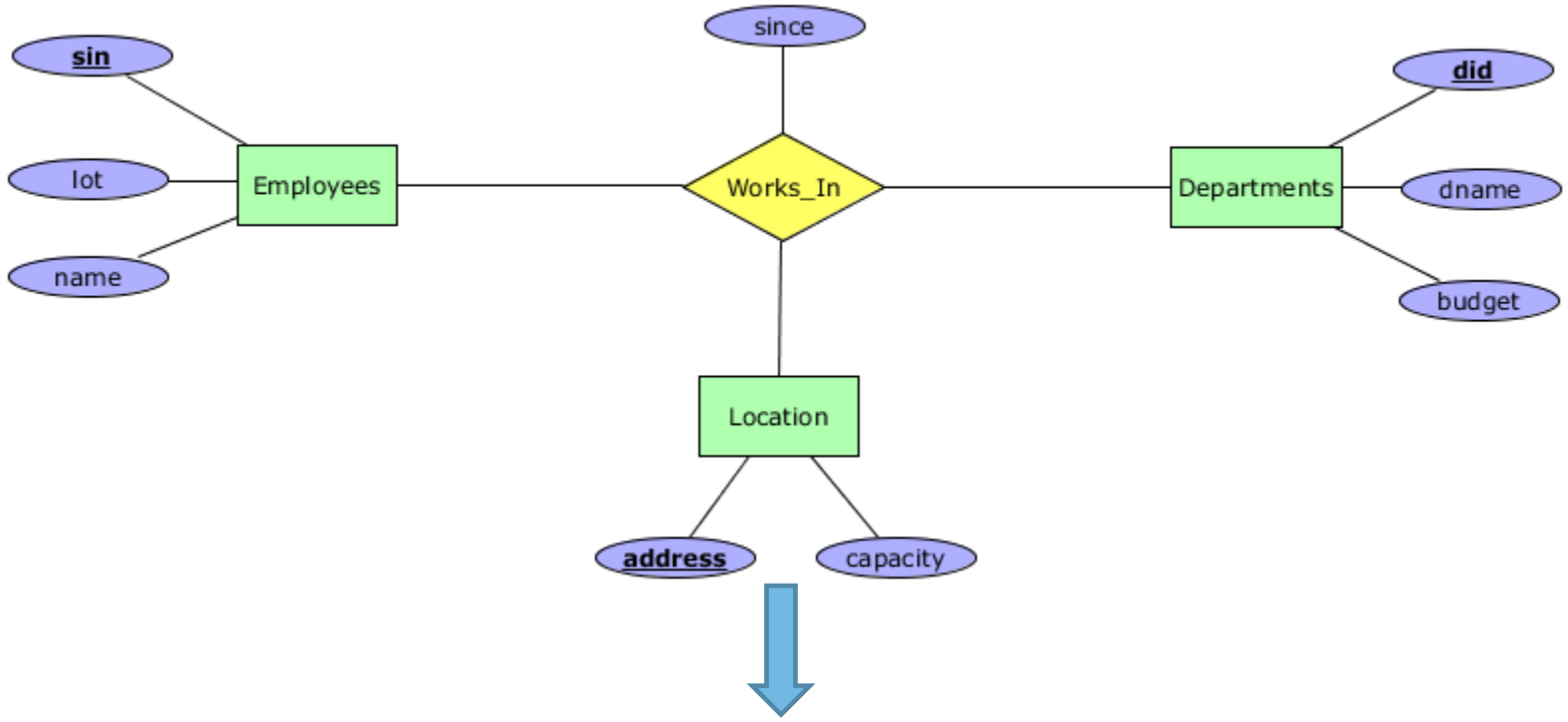
CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

From E/R Relationships to Relations

Example 3 (multiway relationships)

E/R Model



Relational Model

WorksIn(empSIN:string, deptDID:integer, locAddress:string, since:date)

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Combining relations

Sometimes, the relations that we get from converting entity sets and relationships to relations are not the best possible choice of relations for the given data. One common situation occurs when there is an entity set **E** with a many-one relationship **R** from **E** to **F**. The relations from **E** and **R** will each have the key for **E** in their relation schema. In addition, the relation for **E** will have in its schema the attributes of **E** that are not in the key, and the relation for **R** will have the key attributes of **F** and any attributes of **R** itself. Because **R** is many-one, all these attributes are functionally determined by the key for **E**, and we can combine them into one relation with a schema consisting of:

1. All attributes of **E**.
2. The key attributes of **F**.
3. Any attributes belonging to relationship **R**.

For an entity e of **E** that is not related to any entity of **F**, the attributes of types (2) and (3) will have null values in the tuple for e.

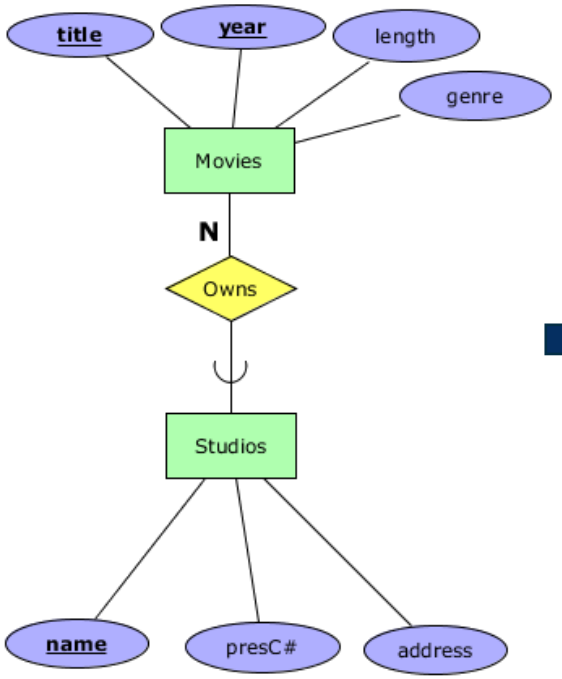
CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Combining relations

Example (without combining relations):

E/R Model



Relational Model

Movies(title:string, year:integer, length: string, genre:string)

Owns(movieTitle:string, movieYear:integer, studioName: string)

Studios(name:string, presC#: integer, address:string)

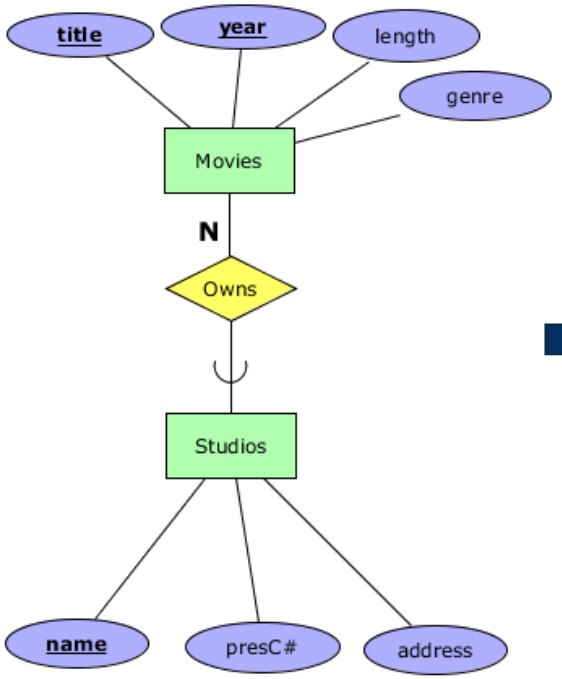
CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Combining relations

Example (combining relations):

E/R Model



Relational Model

Movies(title:string, year:integer, length: string, genre:string, studioName:string)

Studios(name:string, presC#: integer, address:string)

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Combining relations

Whether or not we choose to combine relations in this manner is a matter of judgement. However, there are some advantages to having all the attributes that are dependent on the key of entity set E together in one relation, even if there are a number of many-one relationships from E to other entity sets. For example, it is often more efficient to answer queries involving attributes of one relation than to answer queries involving attributes of several relations.

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Weak Entity Sets

When a weak entity set appears in an E/R diagram, we need to do three things differently.

The relation for the weak entity set **W** itself must include not only the attributes of **W** but also the key attributes of the supporting entity sets. The supporting entity sets are easily recognized because they are reached by supporting (double-diamond) relationships from **W**.

The relation for any relationship in which the weak entity set **W** appears must use as a key for **W** all of its key attributes, including those of other entity sets that contribute to **W**'s key.

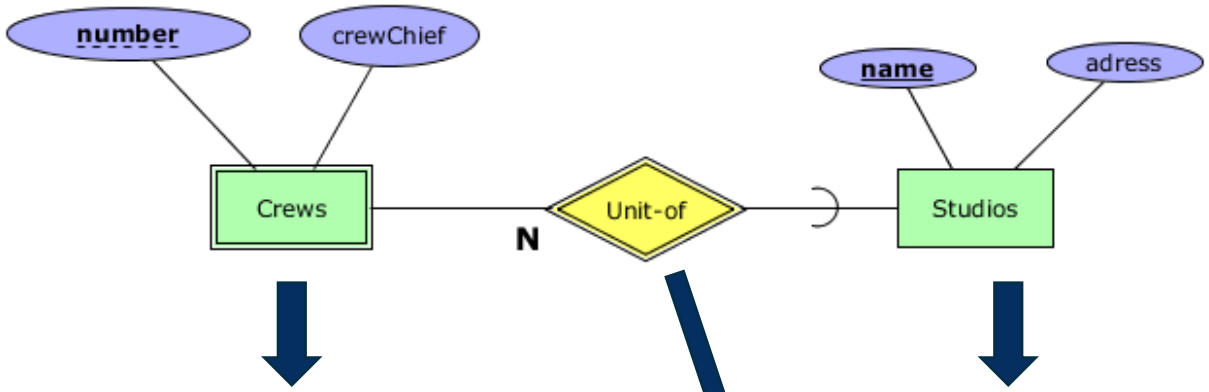
However, a supporting relationship **R**, from the weak entity set **W** to a supporting entity set, need not be converted to a relation at all. The justification is that the attributes of many-one relationship **R**'s relation will either be attributes of the relation for **W**, or (in the case of attributes on **R**) can be added to the schema for **W**'s relation.

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Weak Entity Sets

Example:



Crews(number:integer, crewChief:string, studioName:string)

Studios(name:string, address:string)

Relationship does not need to be converted as it is covered by Crews relation.

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Weak Entity Sets

We can change previous rule as follows. When a weak entity set appears in an E/R diagram, we need to do the following.

- If **W** is a weak entity set, construct for **W** a relation whose schema consists of:
    1. All attributes of **W**.
    2. All attributes of supporting relationships for **W**.
    3. For each supporting relationship for **W**, say a many-one relationship from **W** to entity set **E**, all the key attributes of **E**.
    4. Rename attributes, if necessary, to avoid name conflicts.

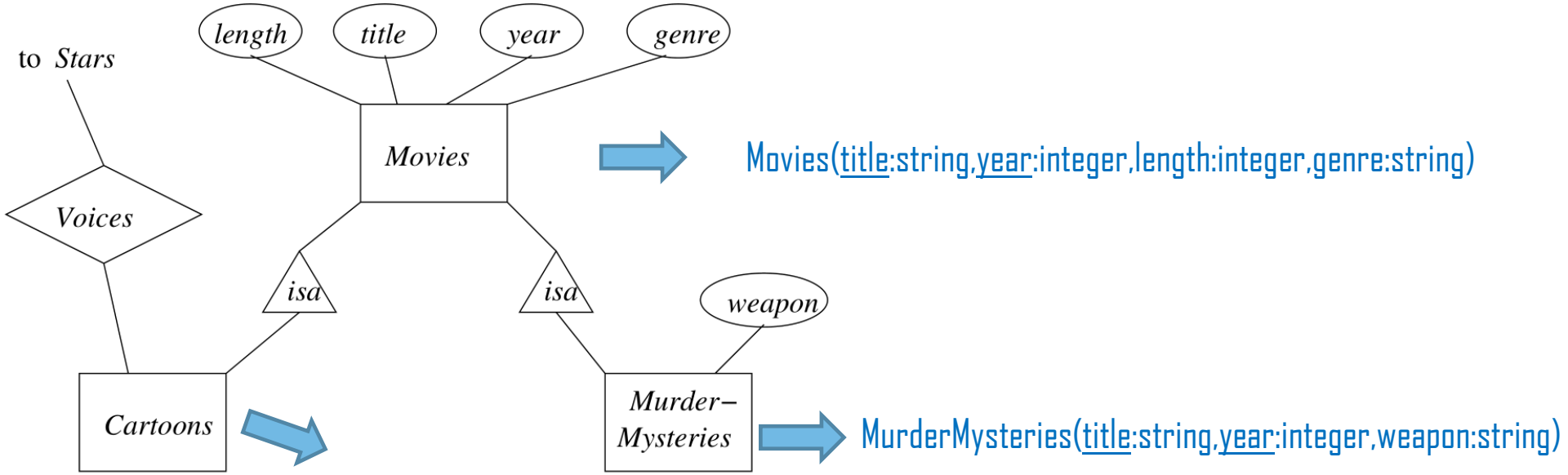
- Do not construct a relation for any supporting relationship for **W**.

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Strategy I: E/R-Style Conversion

First approach is to create a relation for each entity set, as usual. If the entity set **E** is not the root of the hierarchy, then the relation for **E** will include the key attributes at the root, to identify the entity represented by each tuple, plus all the attributes of **E**. In addition, if **E** is involved in a relationship, then we use these key attributes to identify entities of E in the relation corresponding to that relationship.



Movies(title:string,year:integer,length:integer,genre:string)

Cartoons(title:string,year:integer)

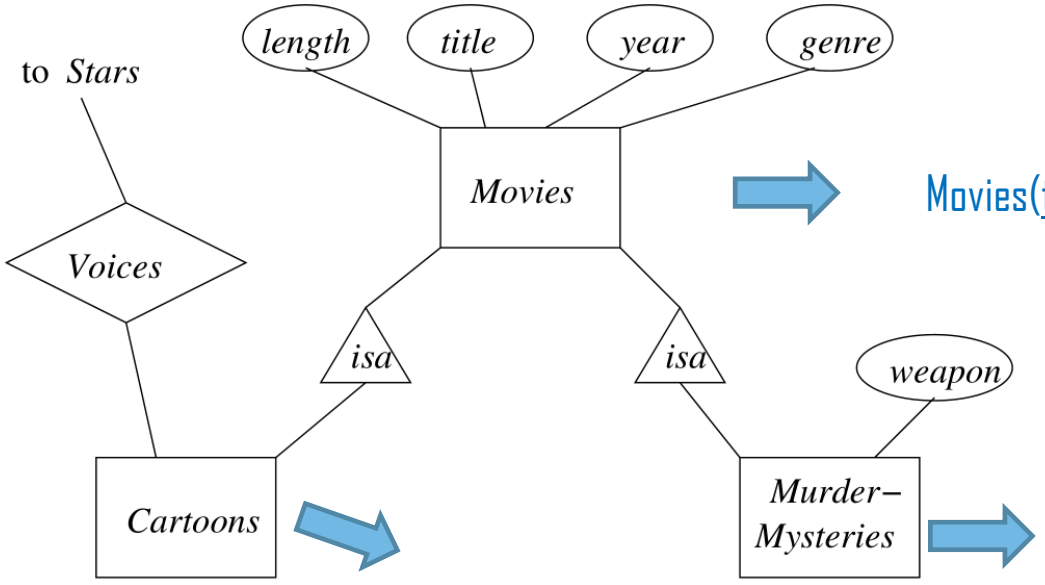
MurderMysteries(title:string,year:integer,weapon:string)

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Strategy II: An Object-Oriented Approach

An alternative strategy for converting isa-hierarchies to relations is to enumerate all the possible subtrees of the hierarchy. For each, create one relation that represents entities having components in exactly those subtrees. The schema for this relation has all the attributes of any entity set in the subtree.



Movies(title:string,year:integer,length:integer,genre:string)

MoviesMM(title:string,year:integer,length:integer,genre:string,weapon:string)

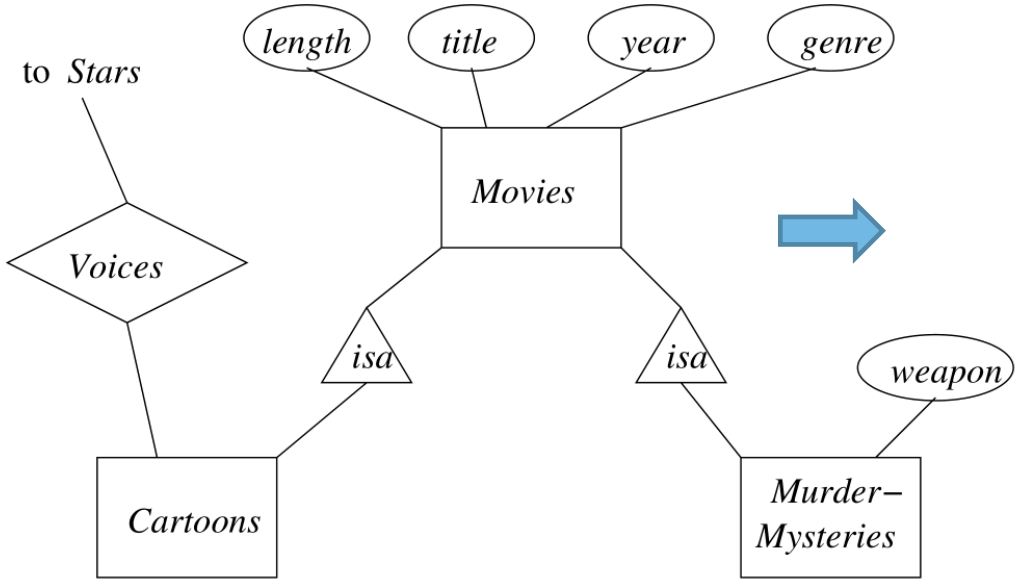
MoviesC(title:string,year:integer,length:integer,genre:string)

CONCEPTUAL MODELS

From E/R Diagrams to Relational Designs

Strategy III: Using Null Values to Combine Relations

If we are allowed to use NULL (the null value as in SQL) as a value in tuples, we can handle a hierarchy of entity sets with a single relation. This relation has all the attributes belonging to any entity set of the hierarchy. An entity is then represented by a single tuple. This tuple has NULL in each attribute that is not defined for that entity.



Movies(title:string,year:integer,length:integer,genre:string,weapon:string)