Relationships and Relationship Types

Concepts of the ER Model (Recap)

- Entities and Entity types
- Relationships and Relationship types
- Attributes

Topics List

- Relationships and Relationship types
- Degree of a Relationship
- Structural Constraints
- Relationship Type Attributes

Relationships

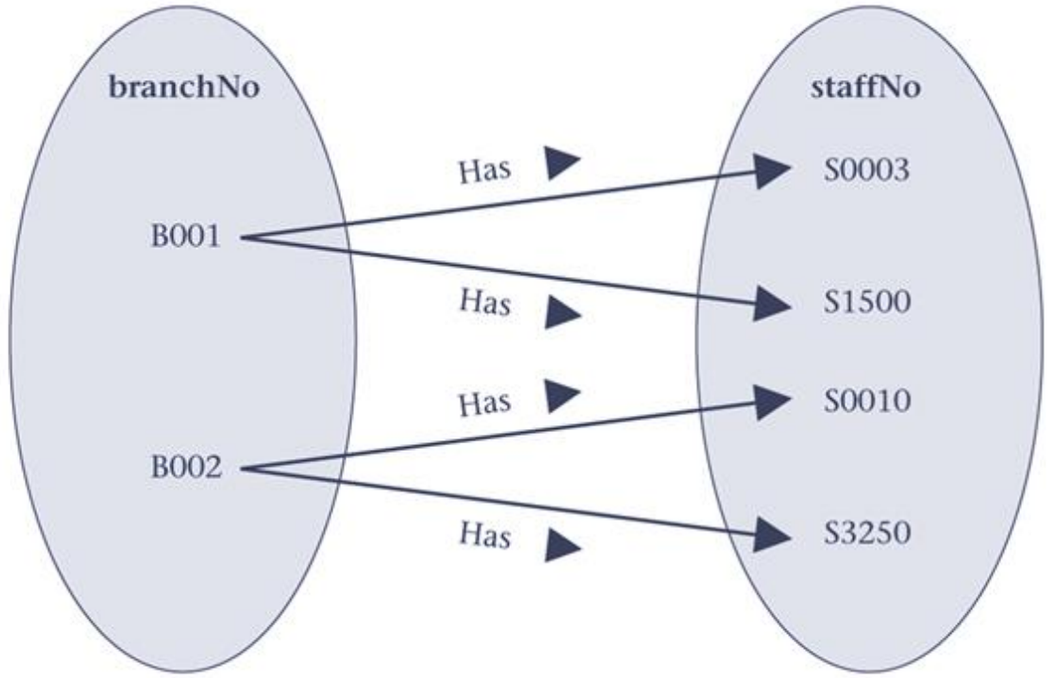
- **Relationship:** The association among entities is called a relationship.

- For example: one particular Student identified by 20081285 *enrolls_on* one particular Course identified by wd155.

Relationships

- In the example, we can see from the occurrence diagram that Branch (identified by branchNo) B001 has 2 Staff members (identified by staffNo) S0003 and S1500.

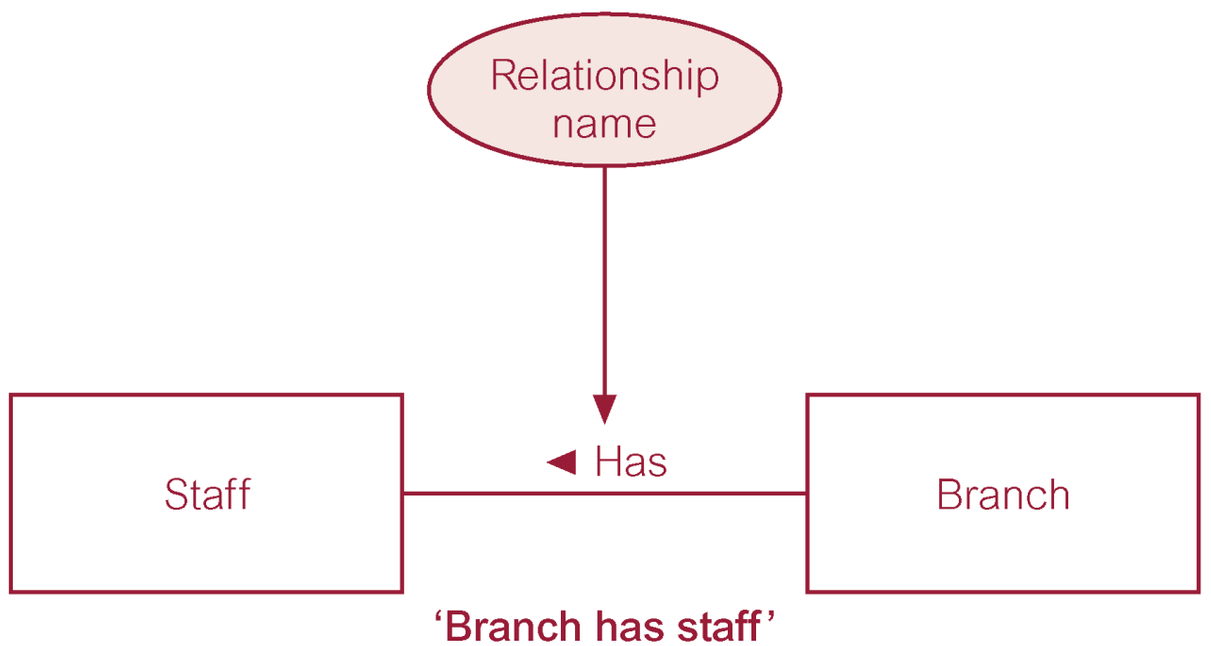
- We can also see that Staff member (identified by staffNo) S0003 works at Branch (identified by branchNo) B001.



Relationship Type

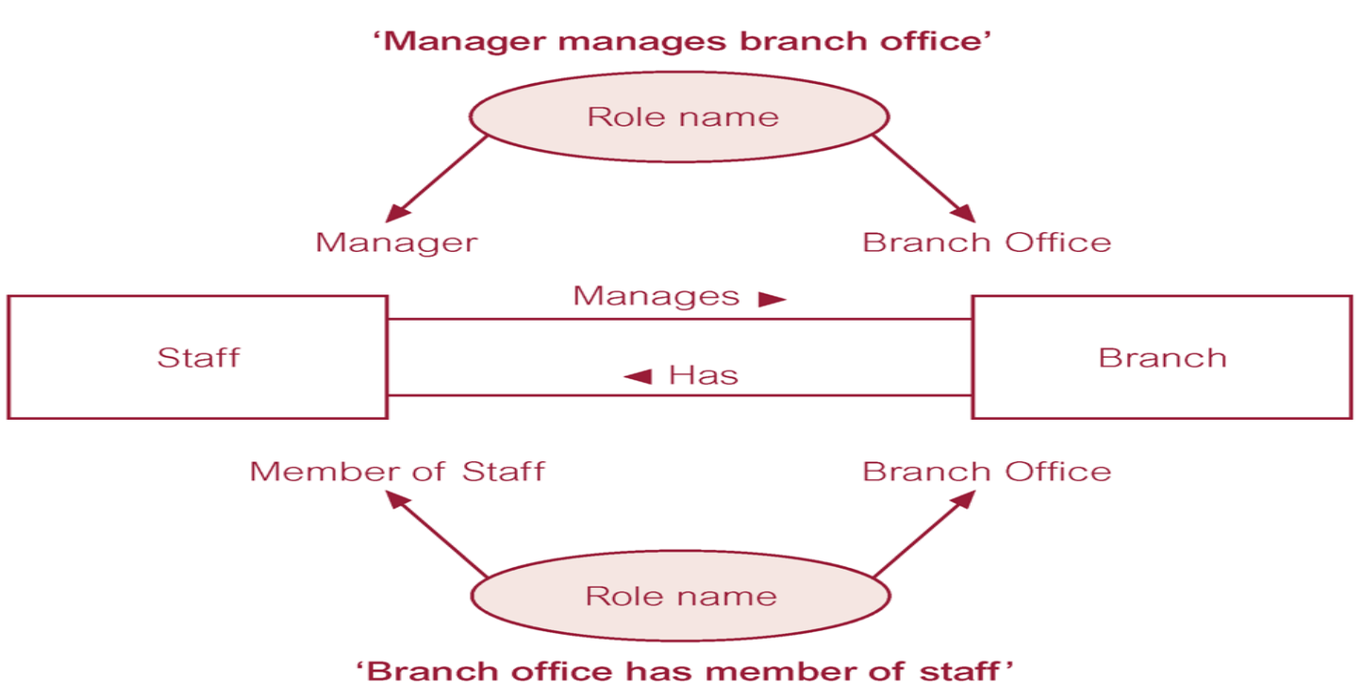
- **Relationship type**: Defines a relationship set among entities of certain entity types.

- Examples:
  - Students *register on* Courses
  - Passengers *reserve* Flights
  - Branch *has* Staff



Relationship Type

- An entity type can have more than one relationship with another entity type. Relationship types may be given role names to indicate purpose that each participating entity type plays in a relationship.



Topics List

- Relationship types and Relationships
- Degree of a Relationship
- Structural Constraints
- Relationship Type Attributes

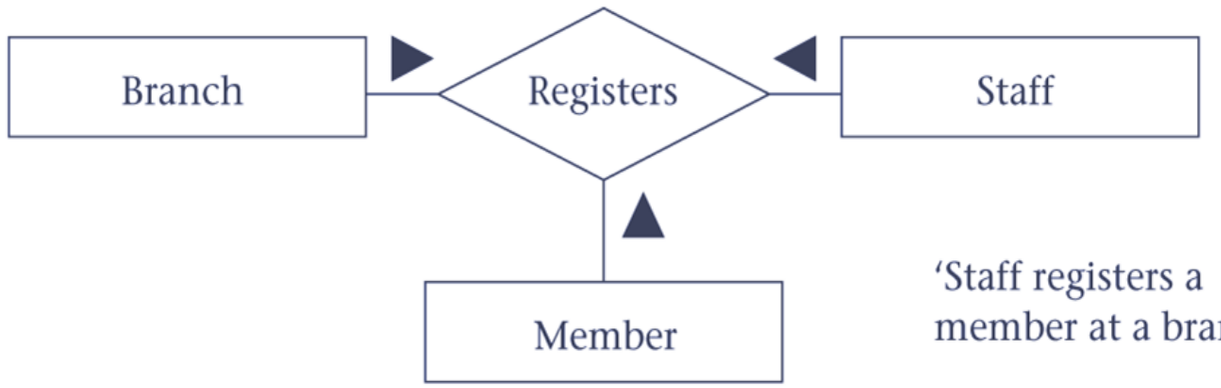
Degree of a Relationship

- ***Degree of a Relationship***: The degree of a relationship is the number of entity sets participating in the relationship. When we speak of a student registered for a course, we are discussing a relationship, *register*, where two entity sets (*Student* and *Course*) are involved; the relationship is of degree *2* because each instance of *register* will always involve one student entity and one course entity.

- Relationships of degree 2 are called *binary relati*onships; relationships of degree 3 are called *ternary relatio*nships. In general, we speak of n-ary relationships where n entities participate in a relationship.

Degree of a Relationship

- The following is an example of a *ternary* relationship.

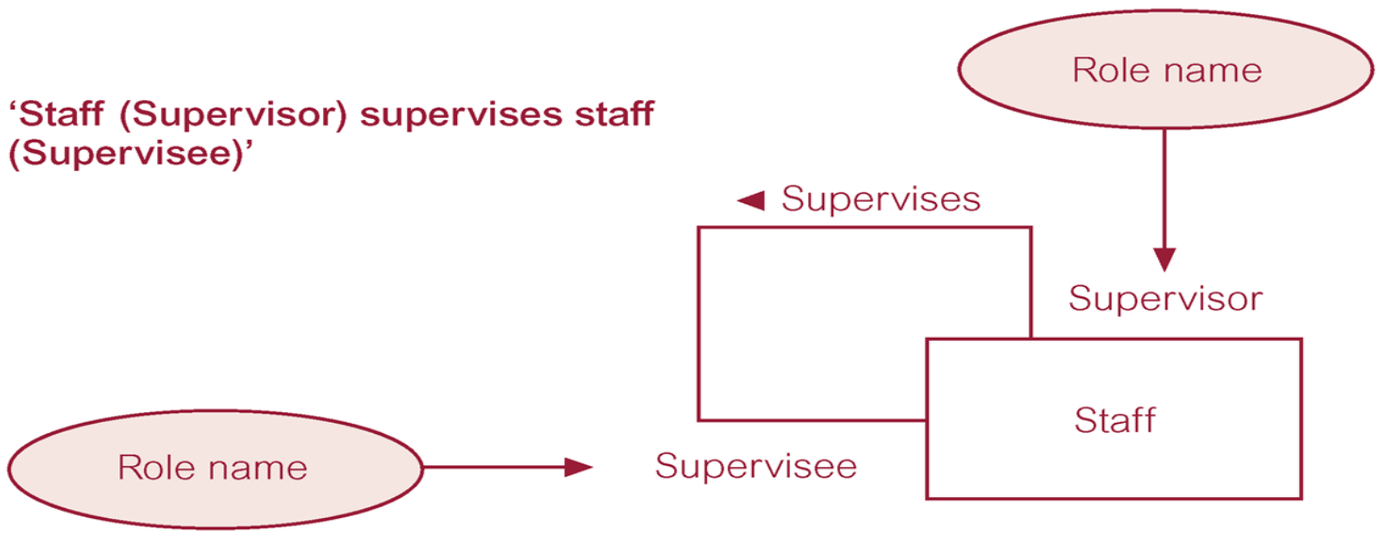


'Staff registers a member at a branch'

Degree of a Relationship

Recursive Relationship

- **Recursive Relationship**: Relationship type where the *same* entity type participates more than once in *different* roles. Sometimes these are called *unary* relationships. A typical business example arises when we have a business rule such as "a staff member supervises other staff members"



Topics List

- Relationship types and Relationships
- Degree of a Relationship
- Structural Constraints
- Relationship Type Attributes

Structural Constraints

- We name a relationship type (usually) using a verb.
- Relationship types are described by giving it a name and by structural constraints (i.e. describing it’s cardinality and participation).

Structural Constraints

Cardinality

- ***Cardinality***: Cardinality is a constraint on a relationship specifying the number of entity instances that a specific entity may be related to via the relationship. Consider the relationship "works in". When we ask *How many employees can work in a single department?* or *How many departments can an employee work in?* we are asking questions regarding the cardinality of the relationship.

Structural Constraints

Cardinality

- The three classifications are:
  - one-to-one (1:1) ,
  - one-to-many (1:*) , and
  - many-to-many (*:*) .

Structural Constraints
Cardinality

one-to-one (1:1) Cardinality

- One-to-one relationships have 1 specified for both cardinalities, and do not seem to arise very often. To illustrate a one-to-one, we require very specific business rules.

Structural Constraints
Cardinality

one-to-one (1:1) Cardinality

- Suppose we have Staff and Vehicles. Assume that we are only concerned with the current driver of a vehicle, and that we are only concerned with the current vehicle that a driver is operating. Then, we have a one-to-one relationship between Staff and Vehicle.

Structural Constraints
Cardinality

one-to-many (1:*) Cardinality

- One-to-many relationships has 1 specified at one side of the relationship type and many (*) at the other side.

- For example, A Branch entity occurrence has many Staff members; and one Staff entity occurrence works in one Branch.

Structural Constraints
Cardinality

many-to-many (*:*) Cardinality

- Many-to-many relationships have many (*) specified for both cardinalities.

- For example, An Actor entity occurrence stars in many Videos; and a Video entity occurrence has many Actors in it.

Structural Constraints

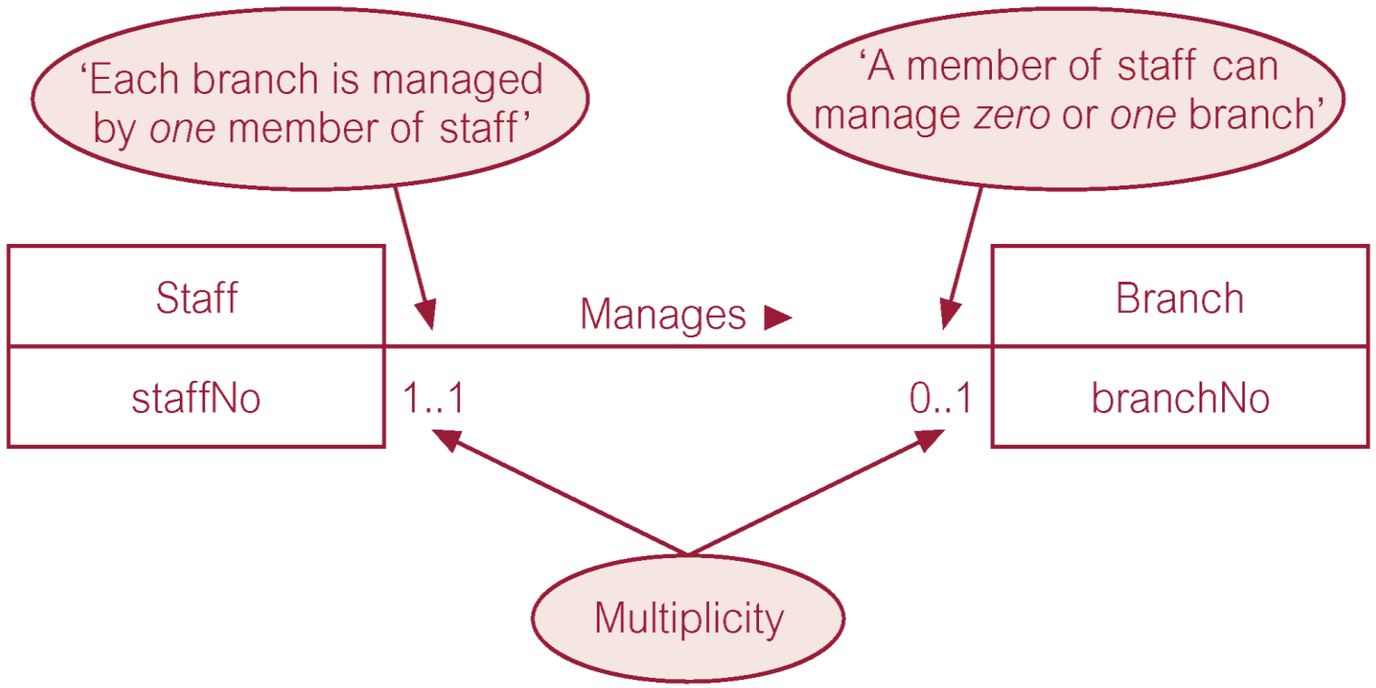
Participation

- **Participation**: Determines whether all or only some entity occurrences participate in a relationship.

- To represent (record) the cardinality and participation constraints, we will use 2 numbers (min, max) where *min represents the minimum participation of an entity occurrence in the relationship (participation). Max represents the maximum participation of an entity occurrence in the relationship (cardinality)*.

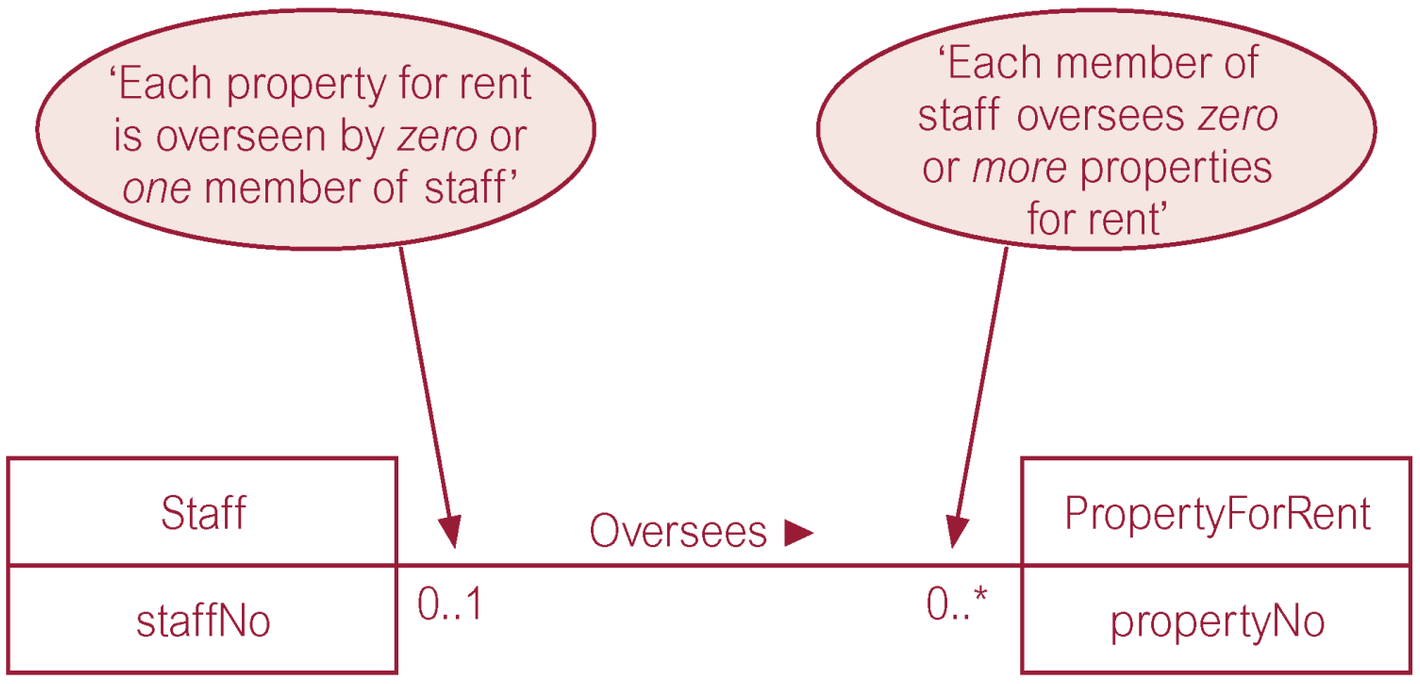
Structural Constraints

Participation



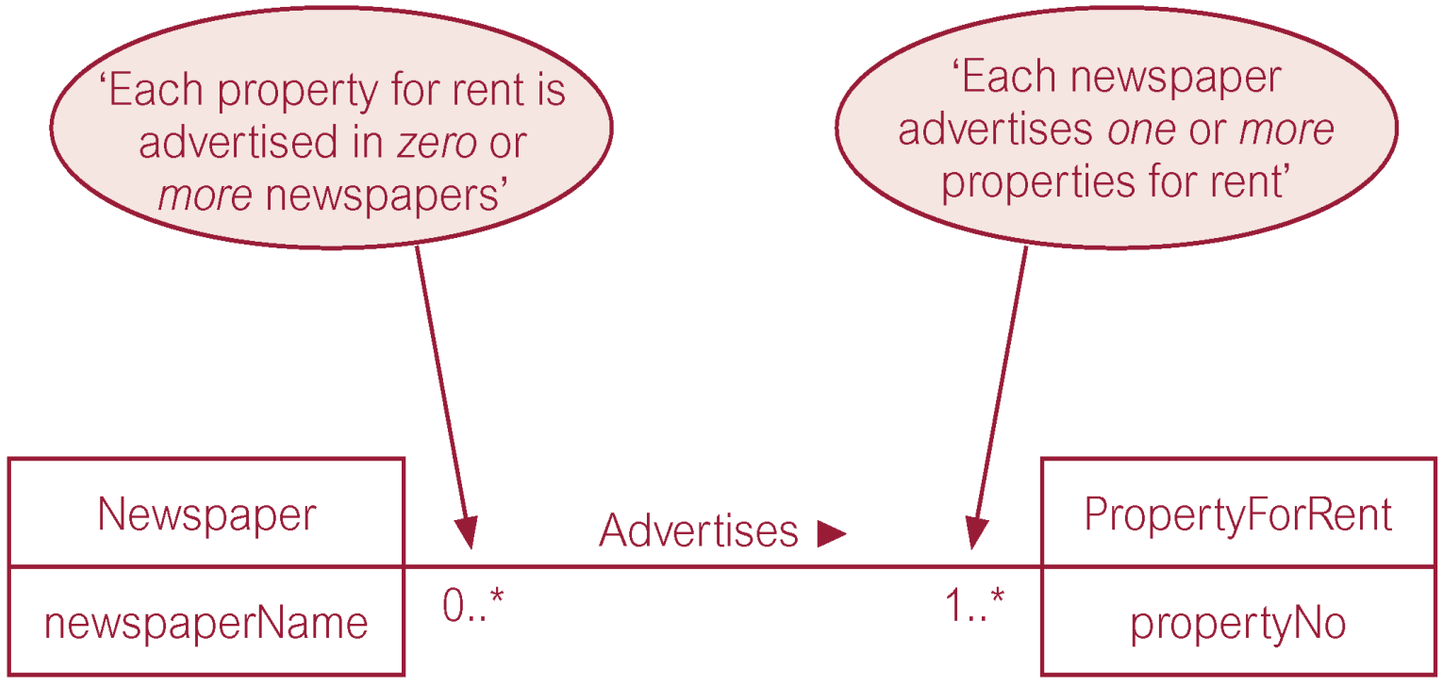
Structural Constraints

Participation



Structural Constraints

Participation



Structural Constraints

- Summary of Structural Constraints

Alternative ways to represent multiplicity constraints	Meaning
0..1	Zero or one entity occurrence
1..1 (or just 1)	Exactly one entity occurrence
0..* (or just *)	Zero or many entity occurrences
1..*	One or many entity occurrences
5..10	Minimum of 5 up to a maximum of 10 entity occurrences
0, 3, 6–8	Zero or three or six, seven, or eight entity occurrences

Structural Constraints

- **Exercise**
  - Draw Entity and Relationship types (with structural constraints) for the following:
    - A course offers at least 6 modules and a module may or may not be offered on a few courses. A lecturer is assigned 3 (and only 3) modules and a module may or may not be only assigned to 1 lecturer.

Note: Only assign one attribute to each entity type.

Structural Constraints

Topics List

- Relationship types and Relationships
- Degree of a Relationship
- Structural Constraints
- Relationship Type Attributes

Relationship Type Attributes

- Attributes can also be assigned to relationship types. They represent a data item value that needs to be recorded when an entity occurrence on each side of the relationship is associated with one another.

- For Example, suppose we have entity types Student and Module that are associated via a *studies* relationship type. An attribute that helps to describe a Student studying a Module is *grade*. This will record the grade that each student obtains on each module that they are associated with.

Relationship Type Attributes

- **Exercise**
  - If we have entity types Employee and Project that are associated via a *worksOn* relationship. What attribute(s) are required capture information about each instance of an Employee working on one Project?