

Entity-Relationship Modelling

陆伟

College of Software

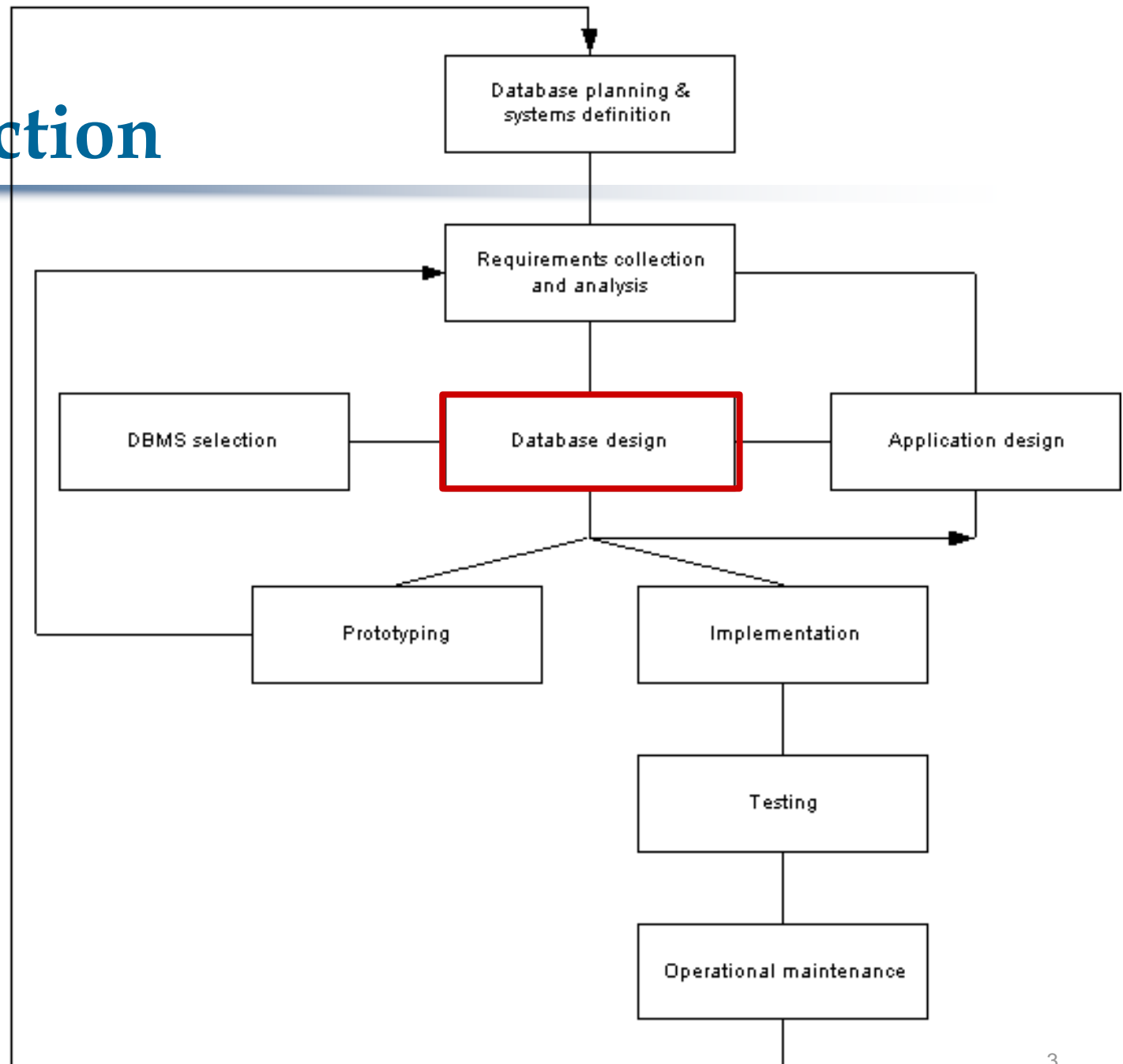
Database Systems-Design and Application

April 16, 2021

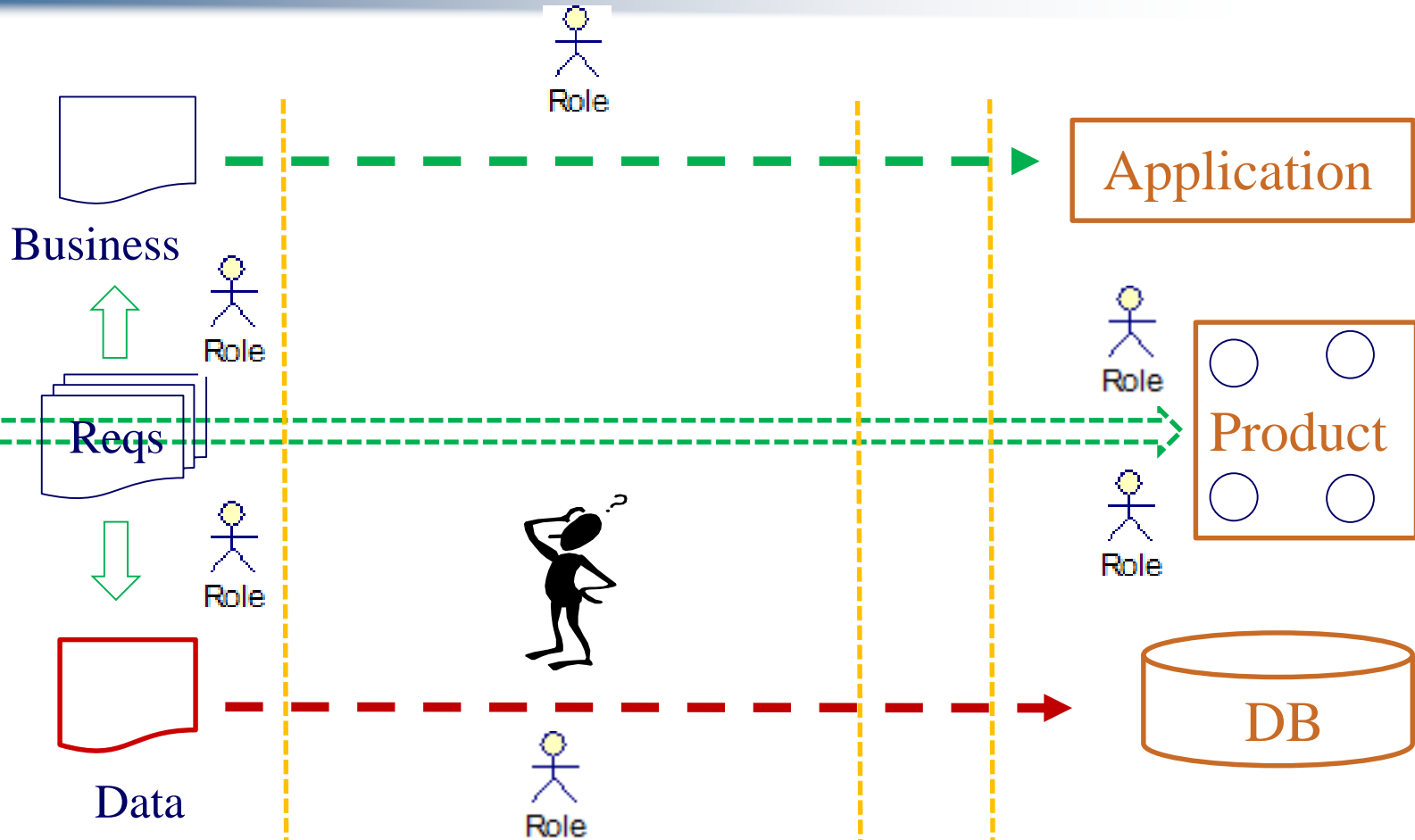
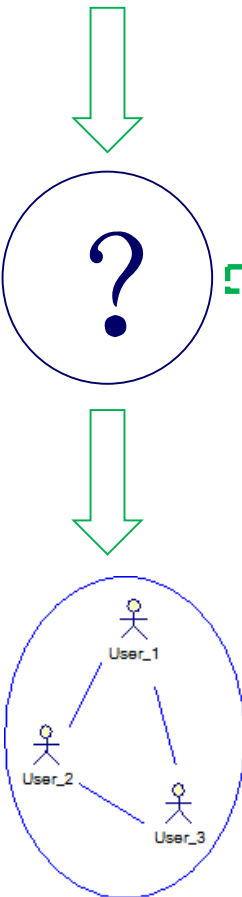
Outline

- Introduction
- Entity and Entity Types
- Relationship and Relationship Types
- Attributes
- Strong and Weak Entity Types
- Structural Constraints
- Problems with ER Models
- Enhanced Entity-Relationship Modelling

Introduction



Introduction



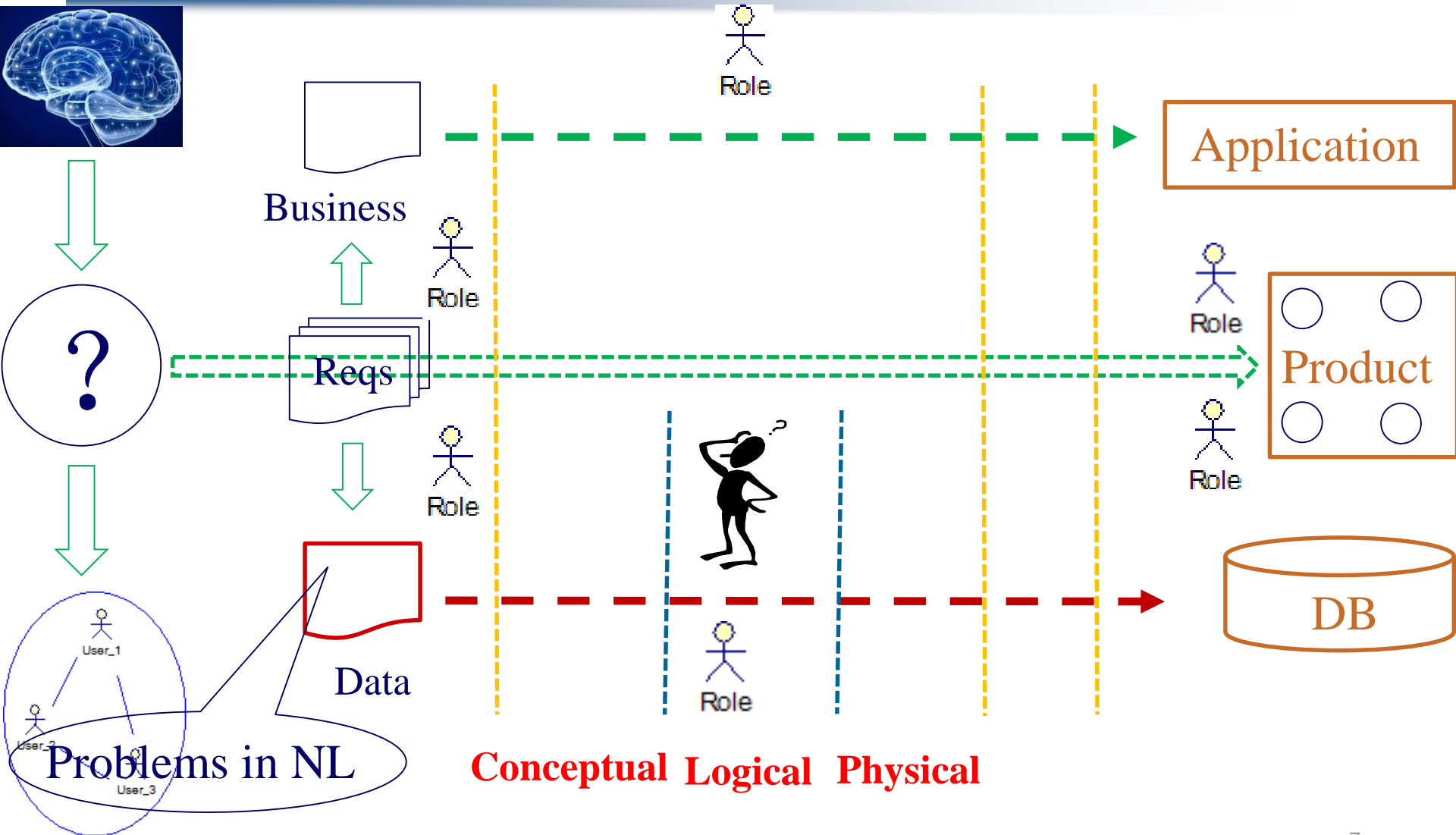
Introduction

- Once the requirements collection and analysis stage of the database application lifecycle is complete and we have **documented the requirements** for the database application, we are ready to begin the database design stage.
- One of the most difficult aspects of database design is **the fact that designers, programmers, and end-users trend to view data and its use in different ways.**

Introduction

- Unless we gain a **common understanding** that reflects how the enterprise operates, the design we produce will fail to meet the users' requirements.
- To ensure that we get a precise understanding of the nature of the data and how it is used by the enterprise, a **model** is needed for communication.

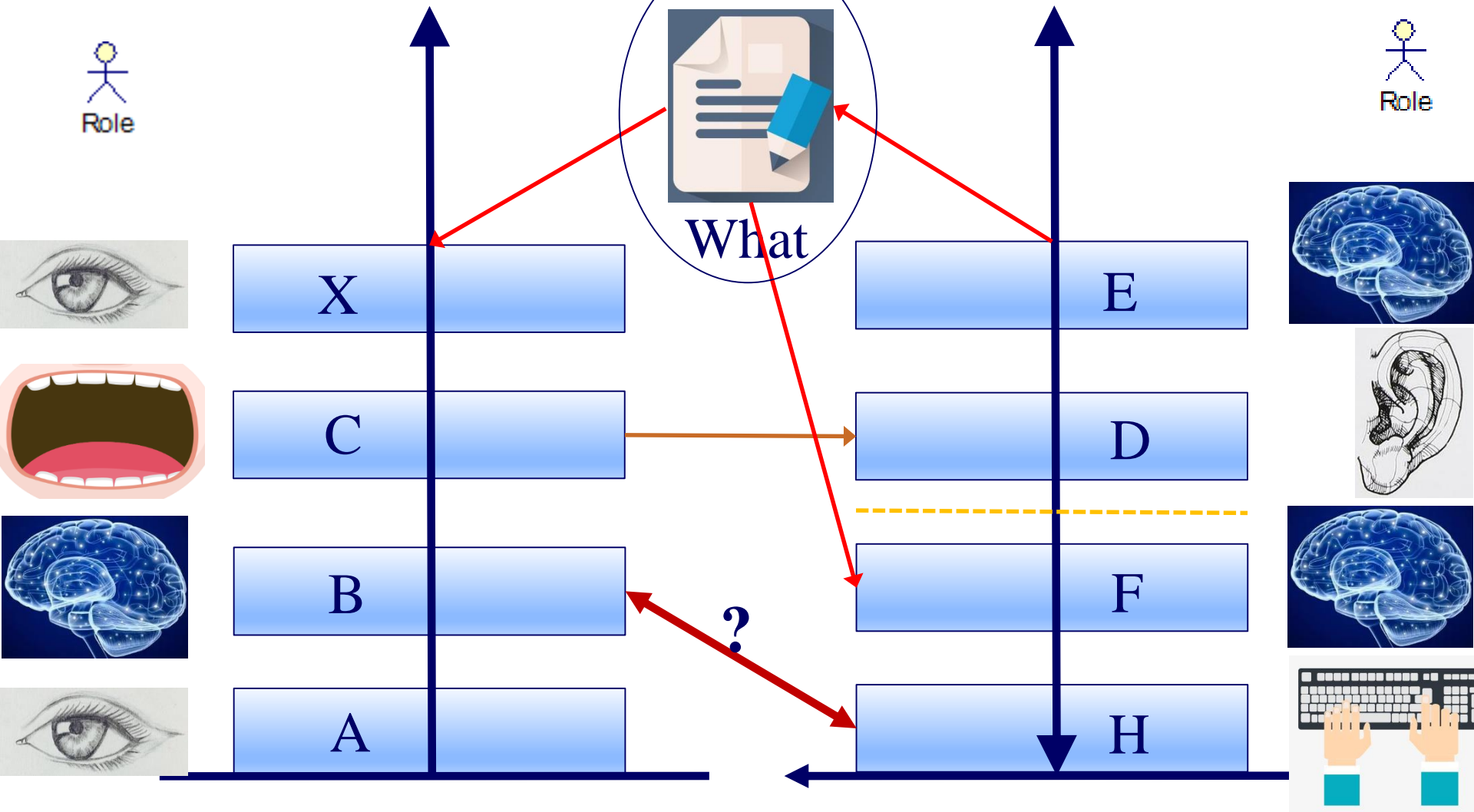
Introduction



Introduction

?

Recess activities



Introduction

- Requirements for the model
 - Non-technical
 - Free of ambiguities (无二义)
- The **Entity-Relationship (ER) model** is one such example.
- ER modeling is a **top-down approach** to database design.
- ER modeling is an important technique for any database designer to master.

Introduction

- The ER model provides a semi-formal notation that allows designers to create a high-level conceptual schema.
- The diagrammatic notations for representing ER model
 - Unified Modeling Language (UML)
 - The Crow's Feet Notation
 - The Chen Notation

Entity and Entity Types

- Entity
 - An *entity* is an object in the real world that is uniquely identifiable and has independent existence.
- Entity Type
 - Entities having the same properties are grouped together to form an *entity type* .
 - Note that the convention is to use singular form for an entity type name.
 - Can be objects with a physical (or real) existence or conceptual existence.

Entity and Entity Types

- Diagrammatic representation of entity types
 - Each entity type is shown as a rectangle labeled with the name of the entity, which is normally a singular noun.
 - The first letter of each word in entity name is upper case.

Student

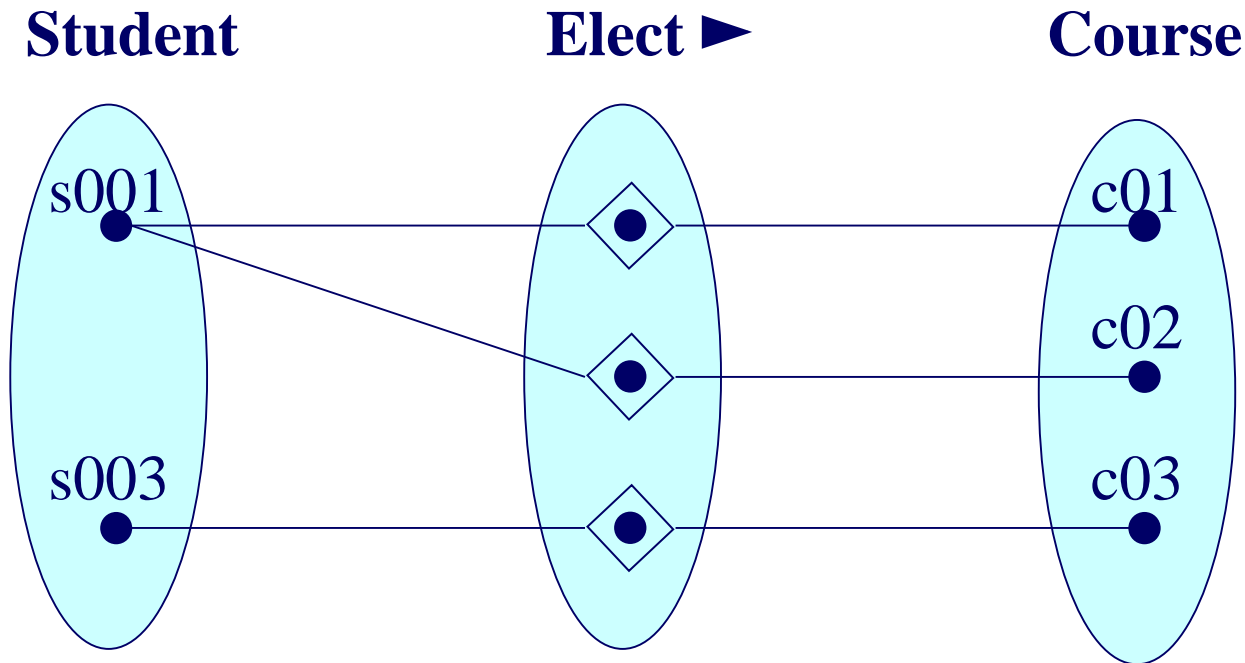
Course

Relationship and Relationship Types

- Relationship
 - A *relationship* is a meaningful association between two or more entities.
- Relationship Type
 - A set of meaningful associations among entity types.
 - A *relationship type* is a grouping of similar relationships.

Relationship and Relationship Types

- About semantic net



Relationship and Relationship Types

- Diagrammatic representation of relationship types

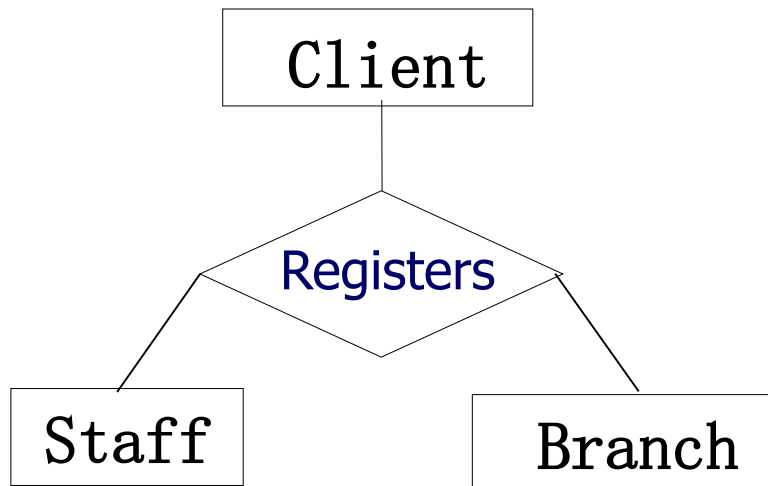


Relationship and Relationship Types

- Degree of Relationship Type
 - The number of participating entity types in a relationship.
- The majority of the relationships in a database are binary.

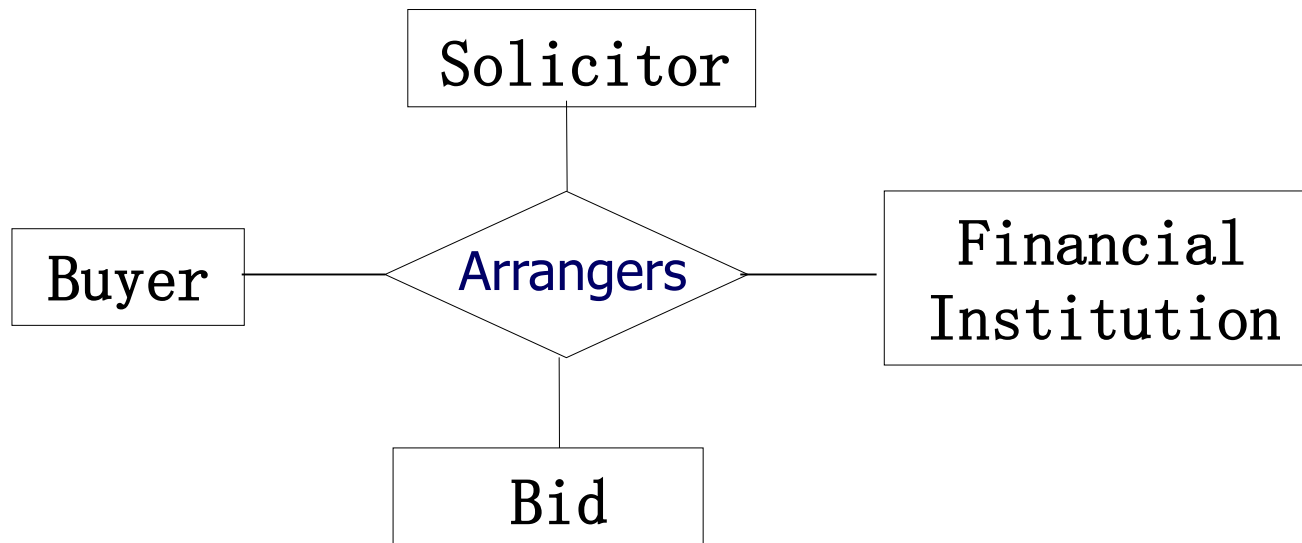
Relationship and Relationship Types

- Diagrammatic representation of *N-ary* relationship types



Relationship and Relationship Types

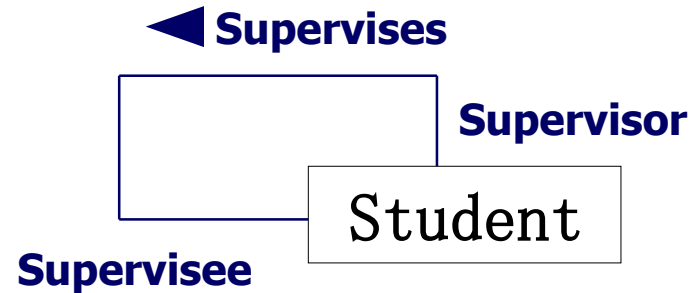
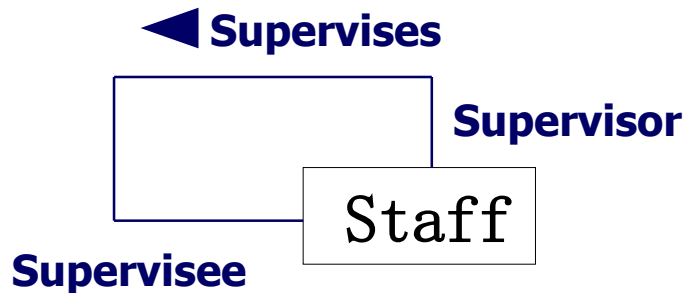
- Diagrammatic representation of *N-ary* relationship types



A buyer(买主), advised by a solicitor(法律顾问) and supported by a financial institution(金融机构), places a bid(投标).

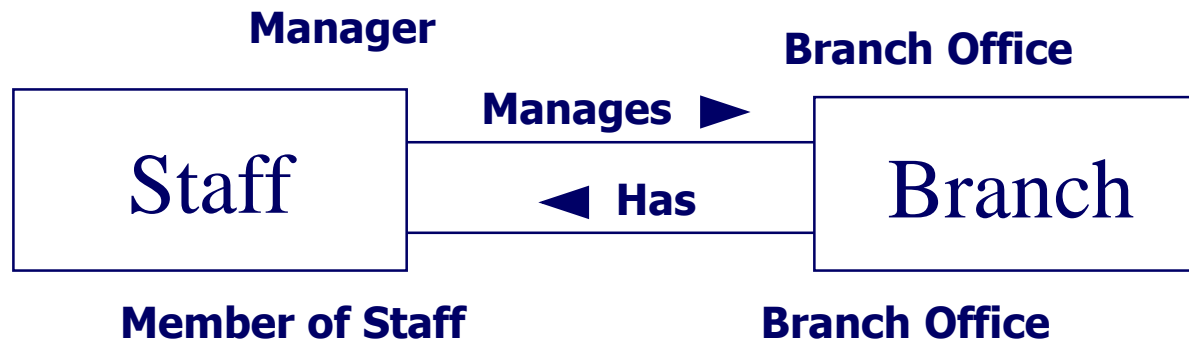
Relationship and Relationship Types

- Recursive Relationship
 - A relationship type where the same entity type participates more than once in different roles.



Relationship and Relationship Types

- Two entities may be associated through more than one relationship. Role may be used.



Attributes

- Attribute
 - A property of an entity or a relationship type.
- Attribute domain
 - The set of allowable values for one or more attributes.
- Simple (atomic) Attribute
 - Attribute composed of a single component with an independent existence.
- Composite Attribute
 - Attribute composed of multiple components, each with an independent existence.

Attributes

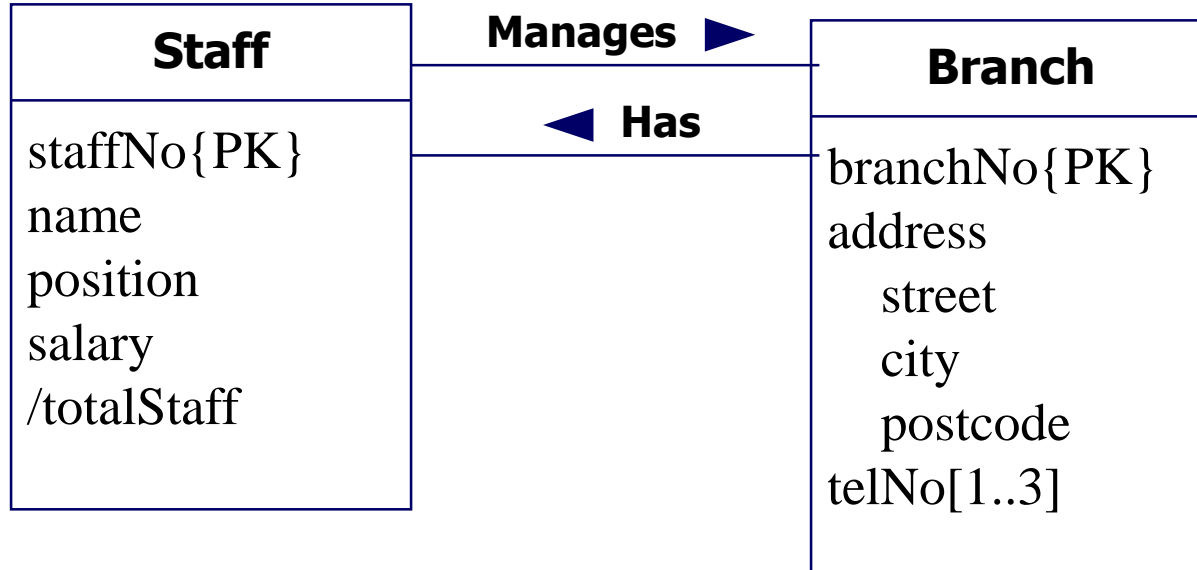
- Single-valued Attribute
 - Attribute that holds a single value for each occurrence of an entity type.
- Multi-valued Attribute
 - Attribute that holds multiple values for each occurrence of an entity type.
- Derived Attribute
 - Attribute that represents a value that is derivable from value of a related attribute, or set of attributes, not necessarily in the same entity type.

Attributes

- Candidate Key
 - Minimal set of attributes that uniquely identifies each occurrence of an entity type.
- Primary Key
 - Candidate key selected to uniquely identify each occurrence of an entity type.
- Composite Key
 - A candidate key that consists of two or more attributes.

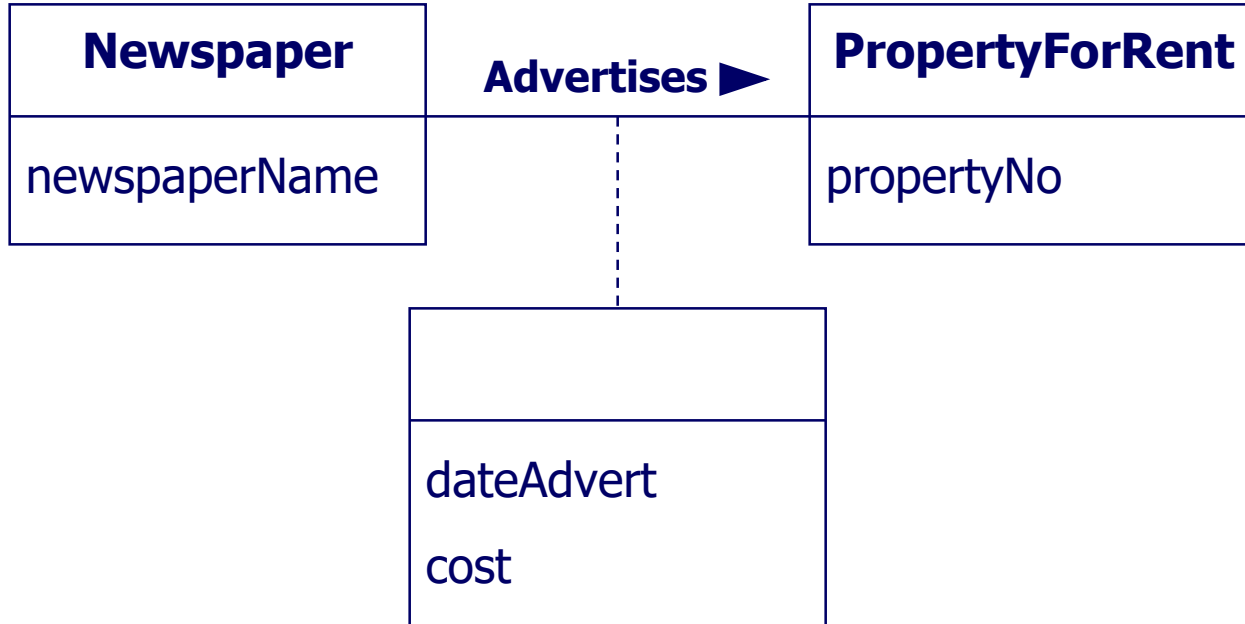
Attributes

- Diagrammatic representation of attributes



Attributes

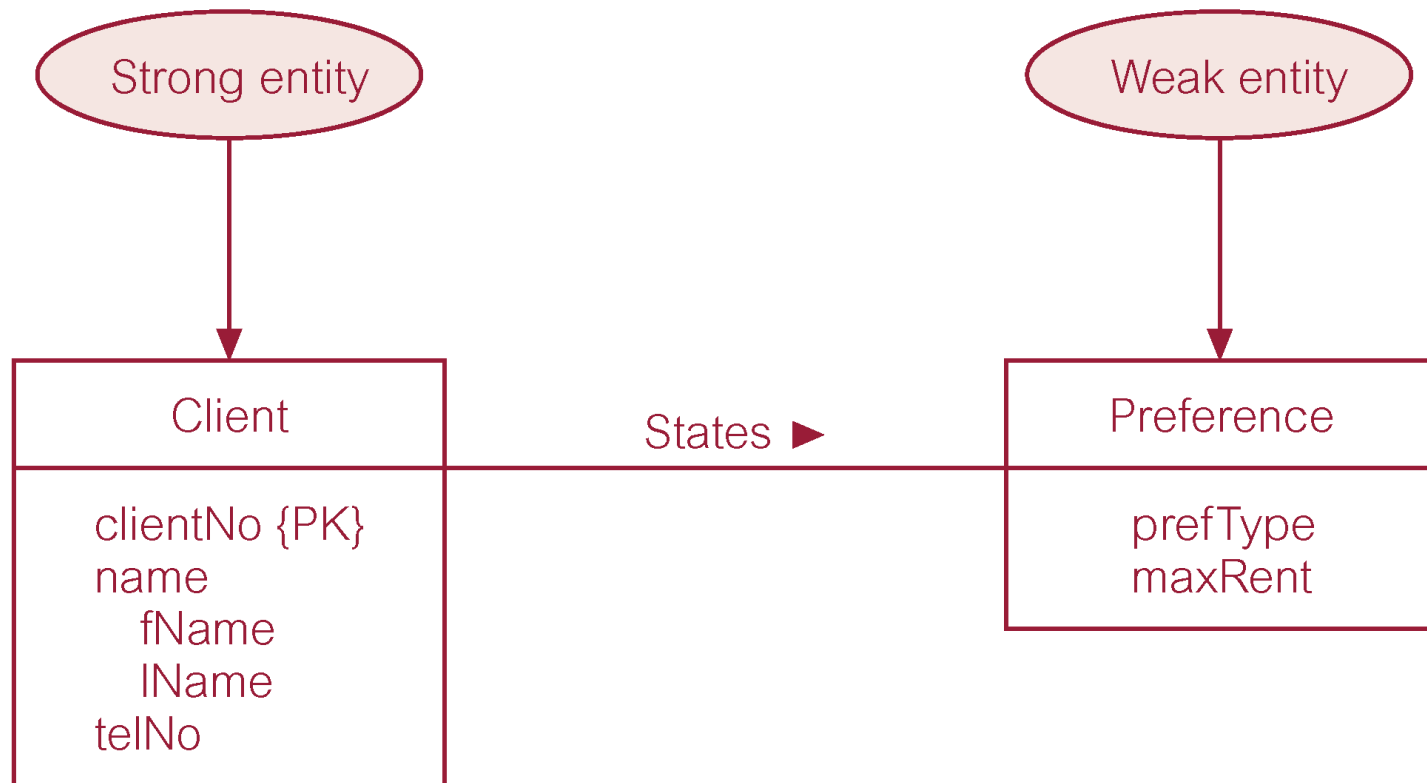
- Attributes on Relationships



Strong and Weak Entity Types

- Strong Entity Type
 - Entity type that is not existence-dependent on some other entity type.
- Weak Entity Type
 - Entity type that is existence-dependent on some other entity type.

Strong and Weak Entity Types



Structural Constraints

- There should be constraints placed on entity types that participate in a relationship.



- Some questions can't be answer by the model
- The constraints on relationships are used to express restrictions on relationships that exist in the real world.

Structural Constraints

- Main type of constraint on relationships is called **multiplicity** (多样性).
- Multiplicity
 - The number (or range) of possible occurrences of an entity type that may relate to a single occurrence of an associated entity type through a particular relationship.
- Represents policies (called business rules) established by user or company.

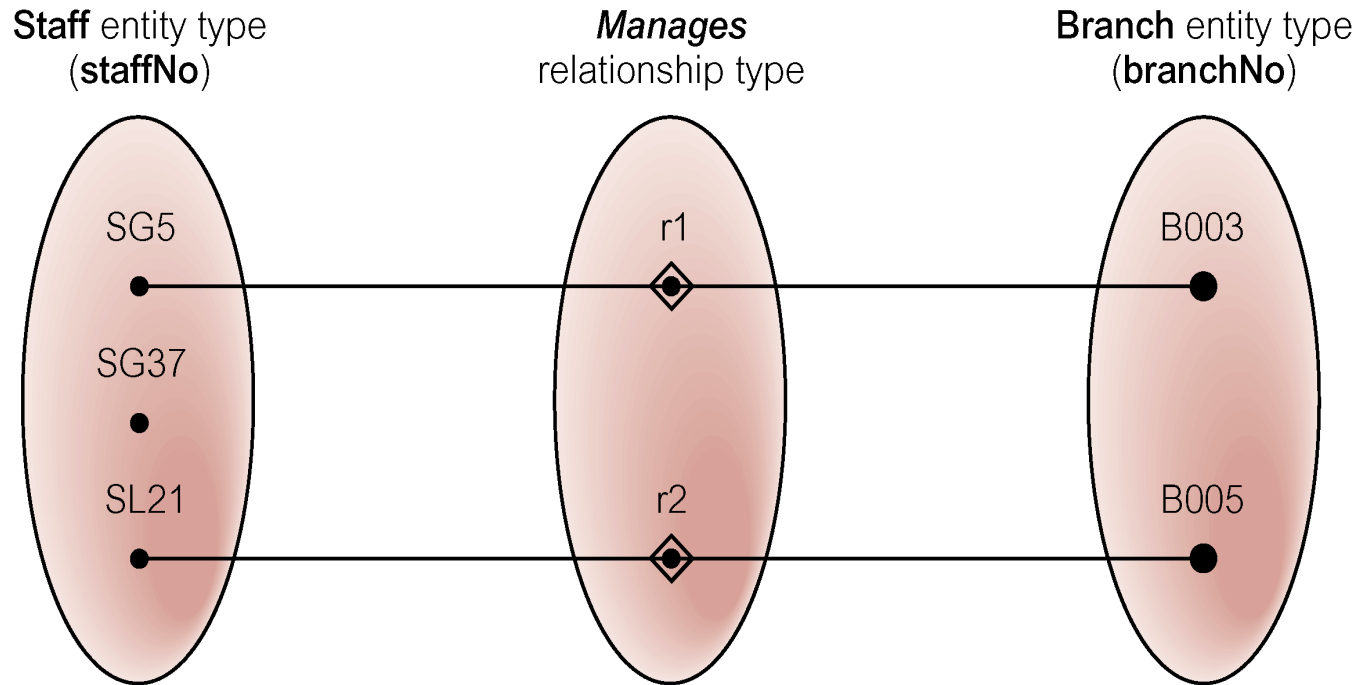
Structural Constraints

- In ER model, we can capture business rules using two types of multiplicity constraints, namely, *cardinality ratio* and *participation*.
- Cardinality ratio
 - Describes maximum number of possible relationship occurrences for an entity participating in a given relationship type.
- Participation
 - Determines whether all or only some entity occurrences participate in a relationship.

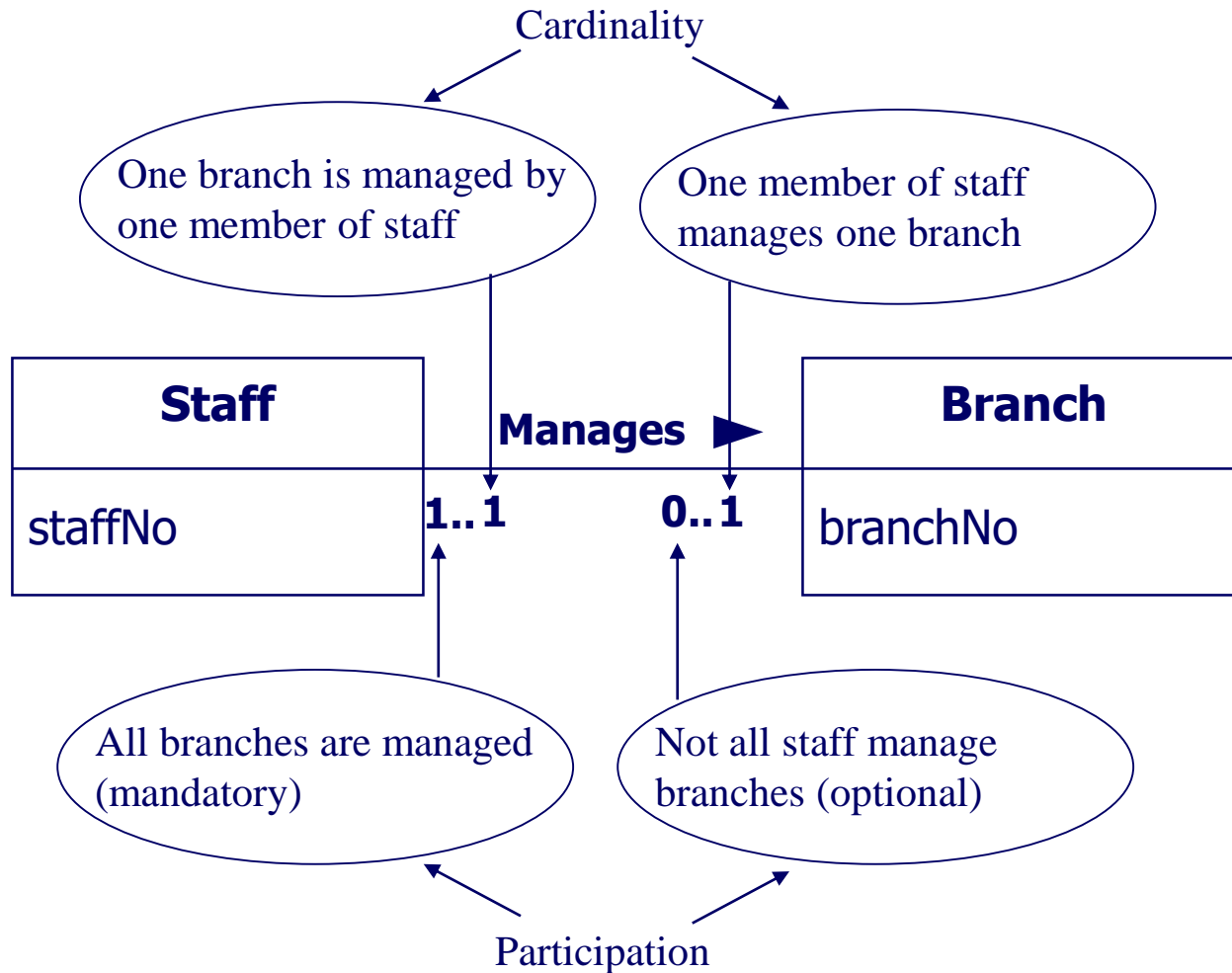
Structural Constraints

- For cardinality constraint, binary relationships are generally referred to as being:
 - one-to-one (1:1)
 - one-to-many (1:*)
 - many-to-many (*:*)
- There are two types of participation constraints
 - Mandatory participation
 - Optional participation

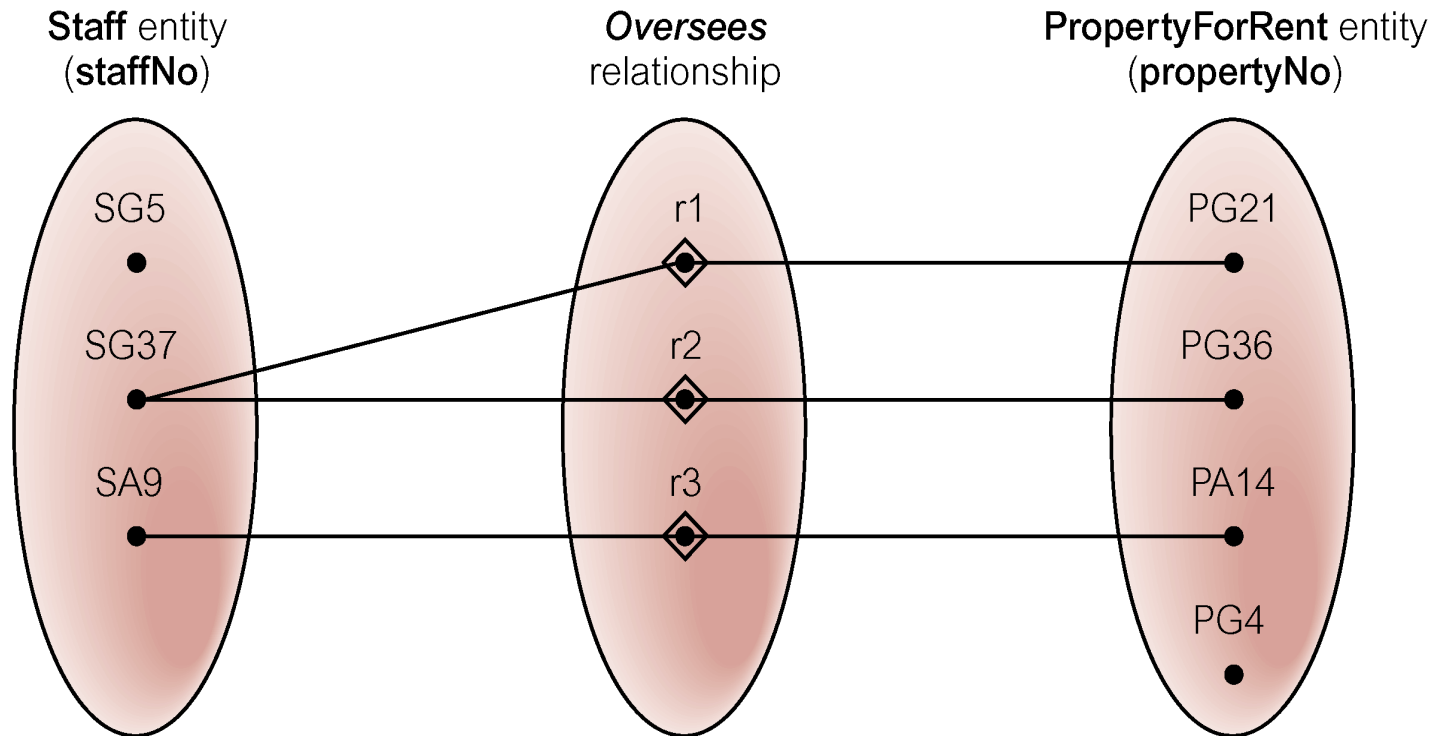
Structural Constraints



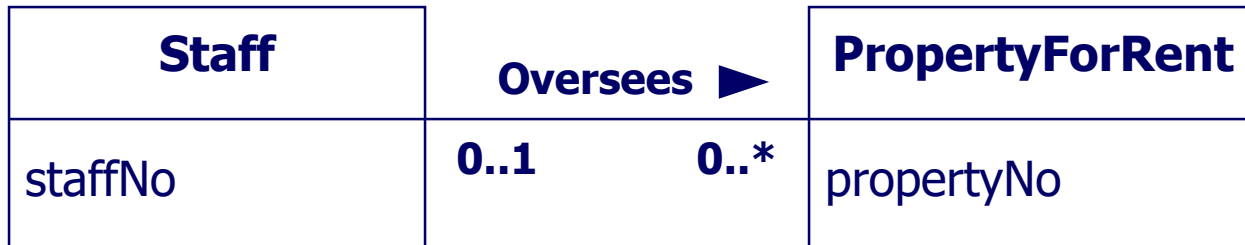
Structural Constraints



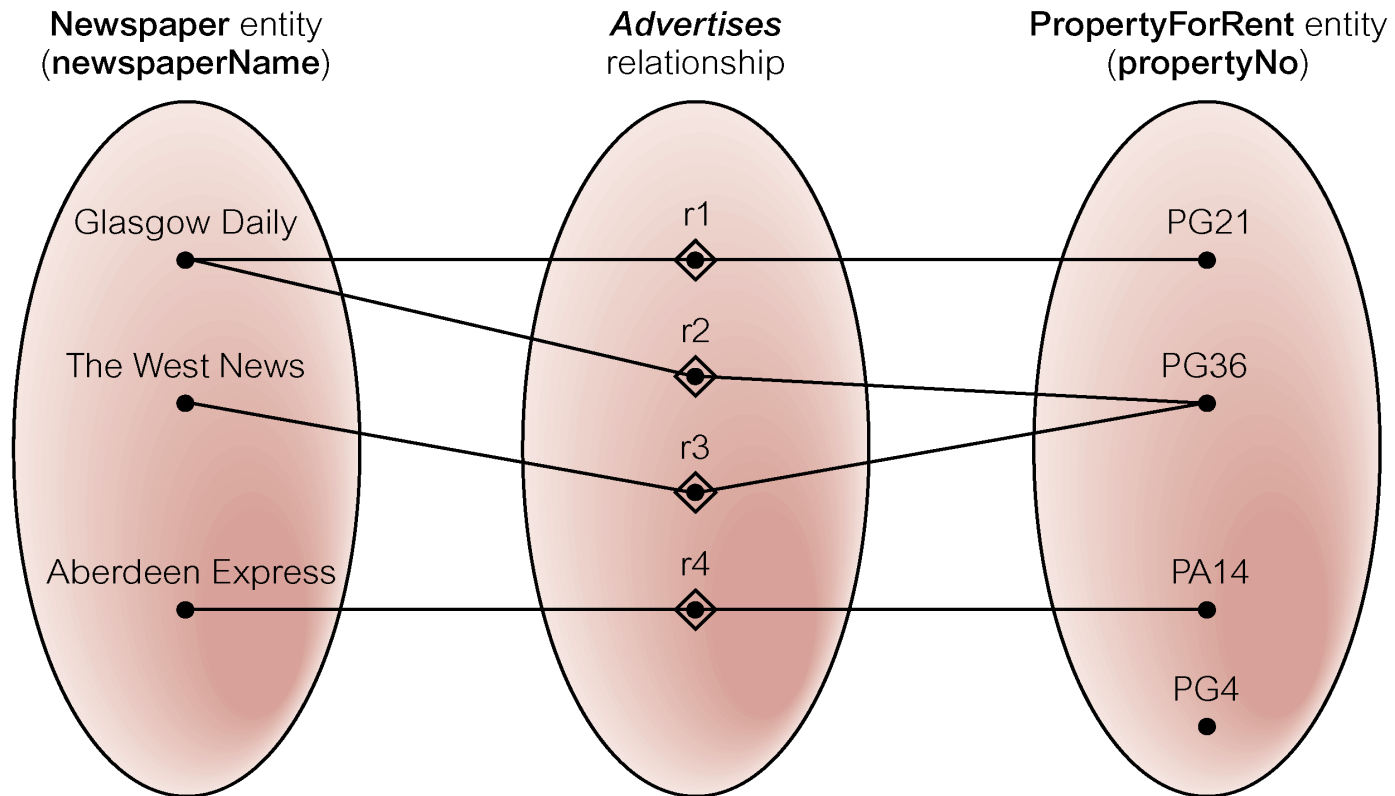
Structural Constraints



Structural Constraints



Structural Constraints



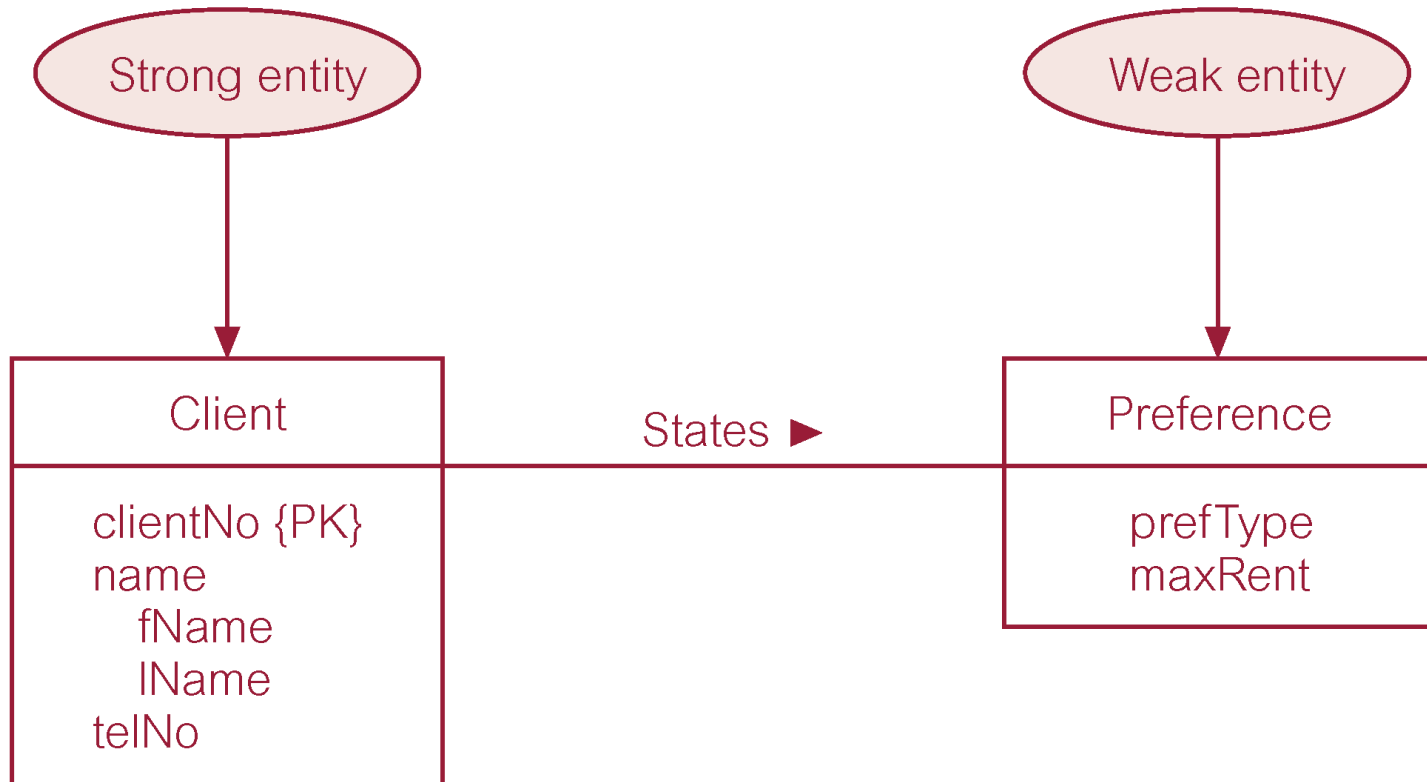
Structural Constraints



Structural Constraints

- Discuss about the multiplicity constraints between strong entity type and weak entity type.
 - A weak entity type must participate in a one-to-many relationship type with the identifying owner.
 - Weak entities have a total participation in the identifying owner relationship, and the identifying owner has a partial participation in the identifying owner relationship.
 - Weak entities have a total participation in the identifying owner relationship, and the identifying owner has a partial participation in the identifying owner relationship.

Structural Constraints



Problems with ER Models

- Problems may arise when designing a conceptual data model called **connection traps**.
- Often due to a misinterpretation of the meaning of certain relationships.
- Two main types of connection traps are called **fan traps** and **chasm traps**.

Problems with ER Models

- Fan Trap
 - Where a model represents a relationship between entity types, but pathway between certain entity occurrences is ambiguous.
- Chasm Trap
 - Where a model suggests the existence of a relationship between entity types, but pathway does not exist between certain entity occurrences.

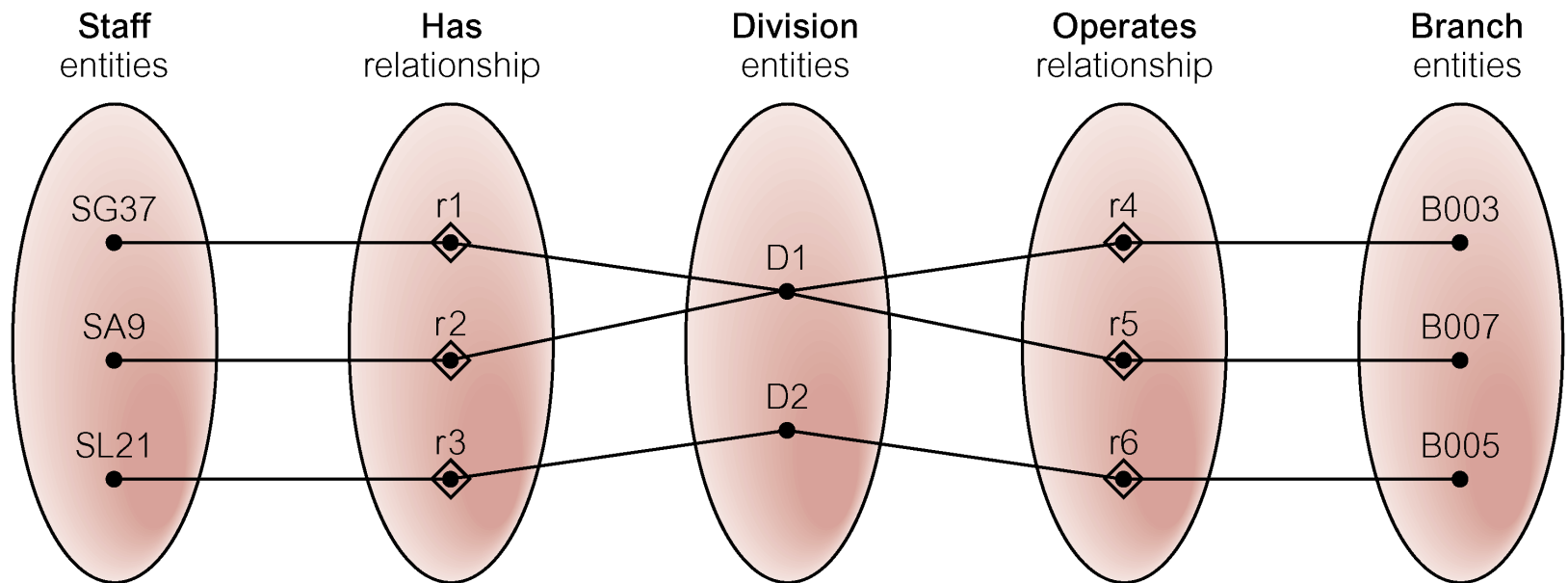
Problems with ER Models

- An Example of a Fan Trap



Problems with ER Models

■ An Example of a Fan Trap



At which branch office does staff number SG37 work?

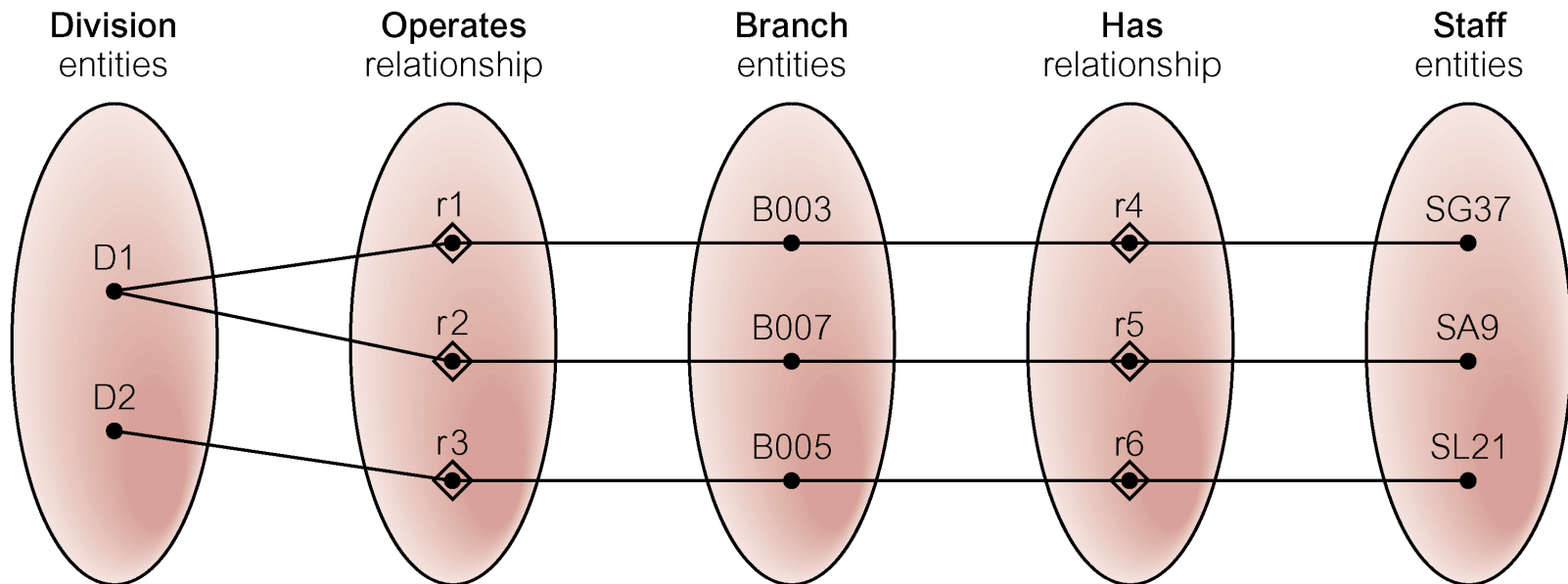
Problems with ER Models

- Restructuring ER model to remove Fan Trap



Problems with ER Models

- Restructuring ER model to remove Fan Trap



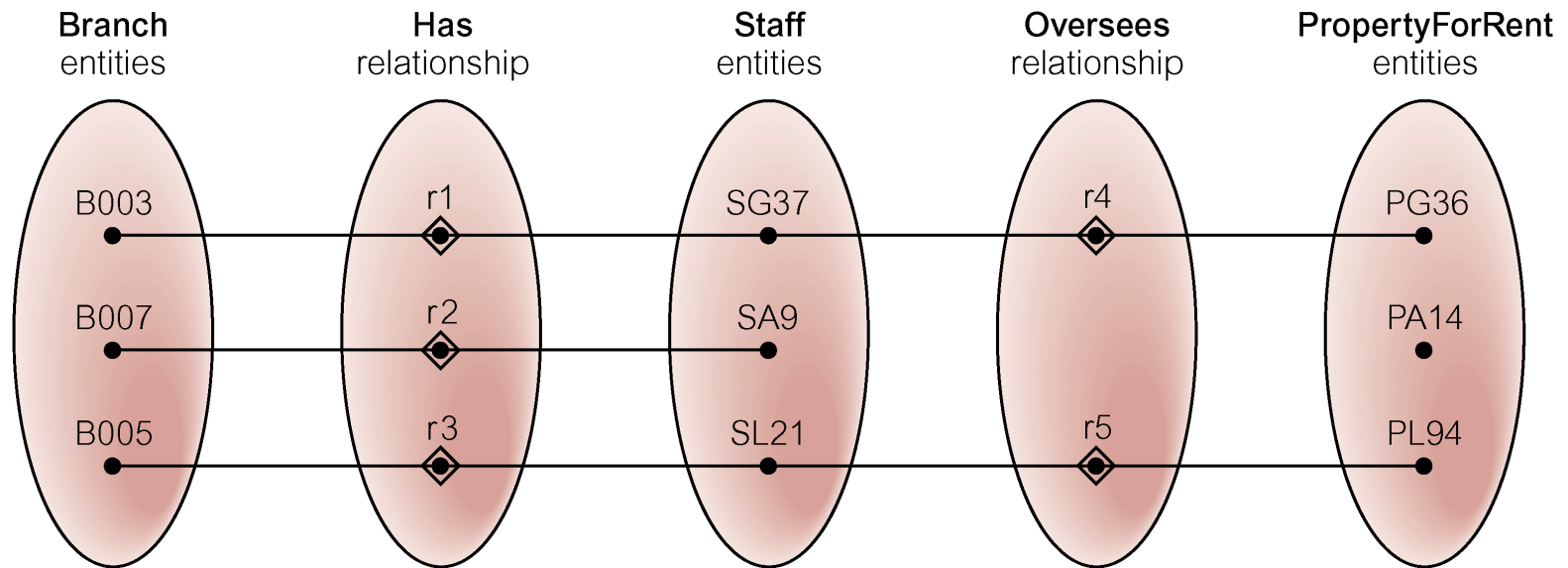
Problems with ER Models

- An Example of a Chasm Trap



Problems with ER Models

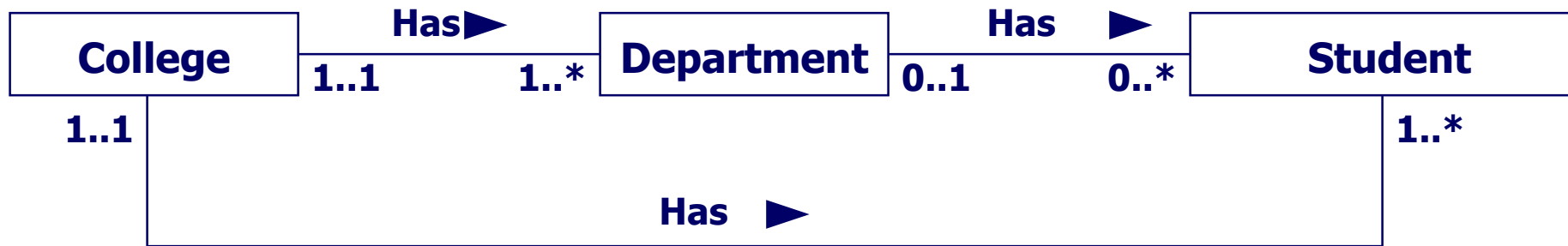
■ An Example of a Chasm Trap



At which branch office is property PA14 available?

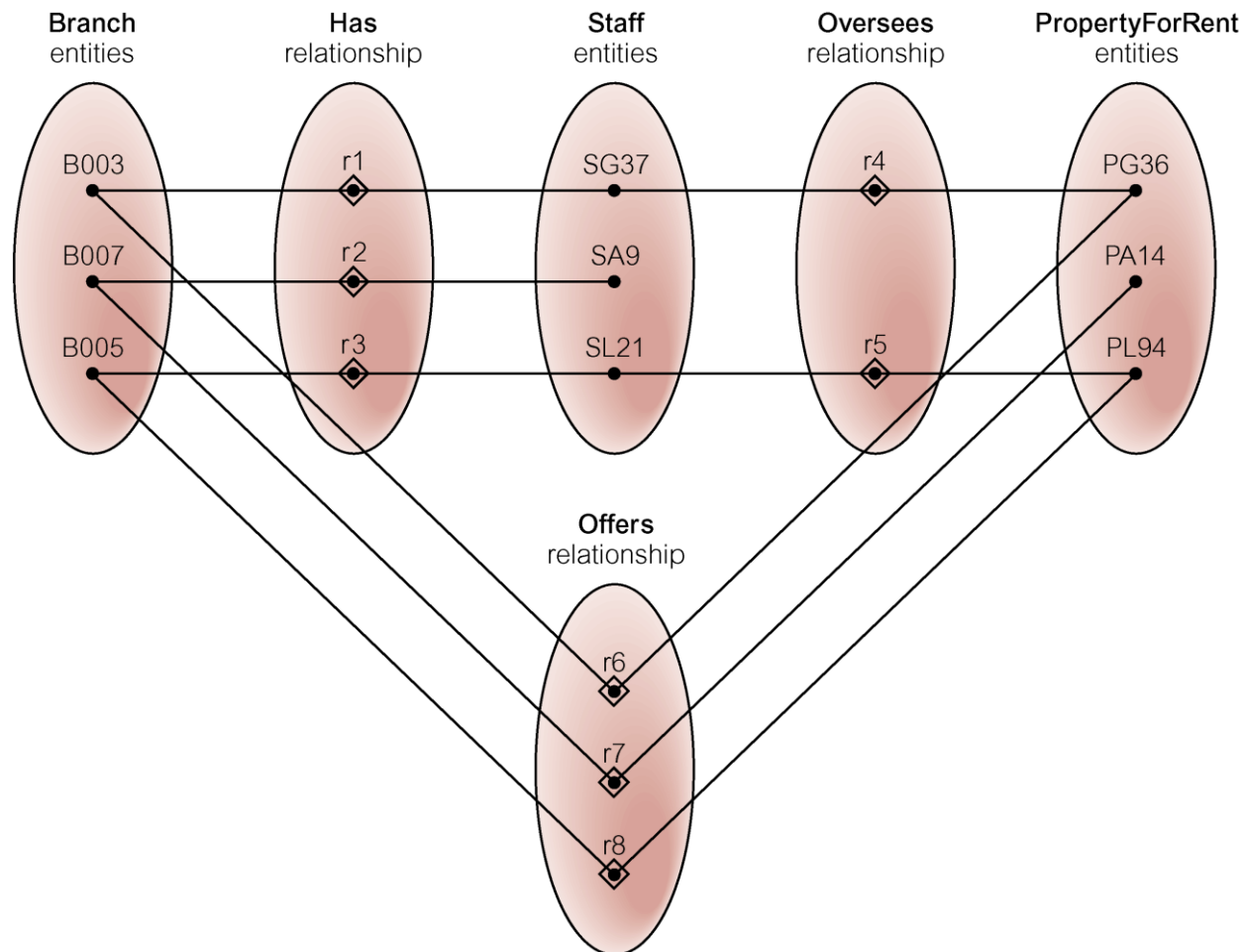
Problems with ER Models

- Restructuring ER model to remove Chasm Trap



Problems with ER Models

- Restructuring ER model to remove Chasm Trap



Problems with ER Models

- Discuss
 - When a fan trap may exist?
 - When a chasm trap may occur?
- A fan trap may exist where two or more 1:* relationships fan out from the same entity.
- A chasm trap may occur where there are one or more relationships with a minimum multiplicity of zero (that is optional participation) forming part of the pathway between related entities.

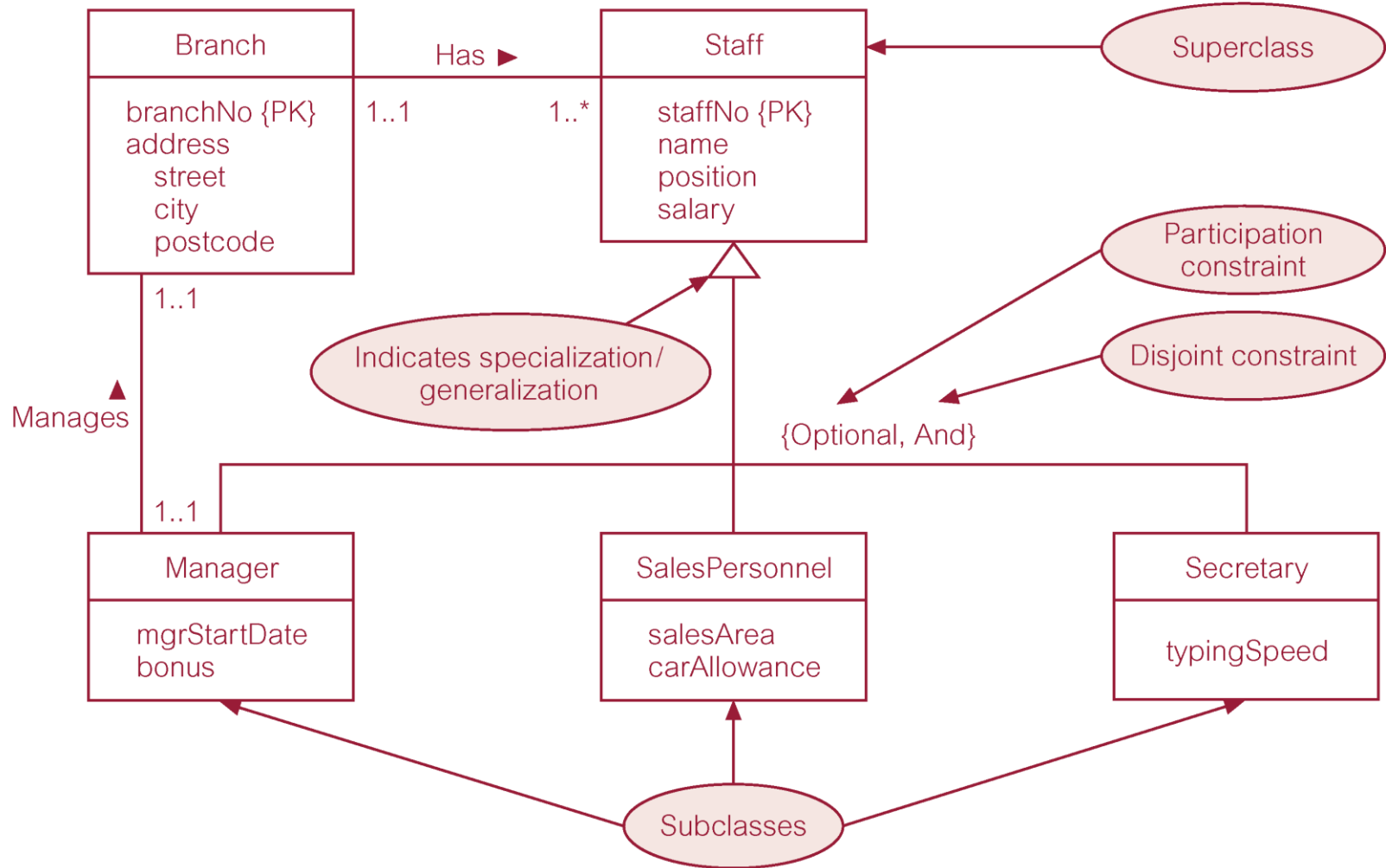
Enhanced Entity-Relationship Modelling

- Since 1980s there has been an increase in emergence of new database applications with more demanding requirements.
- Basic concepts of ER modeling are not sufficient to represent requirements of newer, more complex applications.
- Response is development of additional 'semantic' modeling concepts.

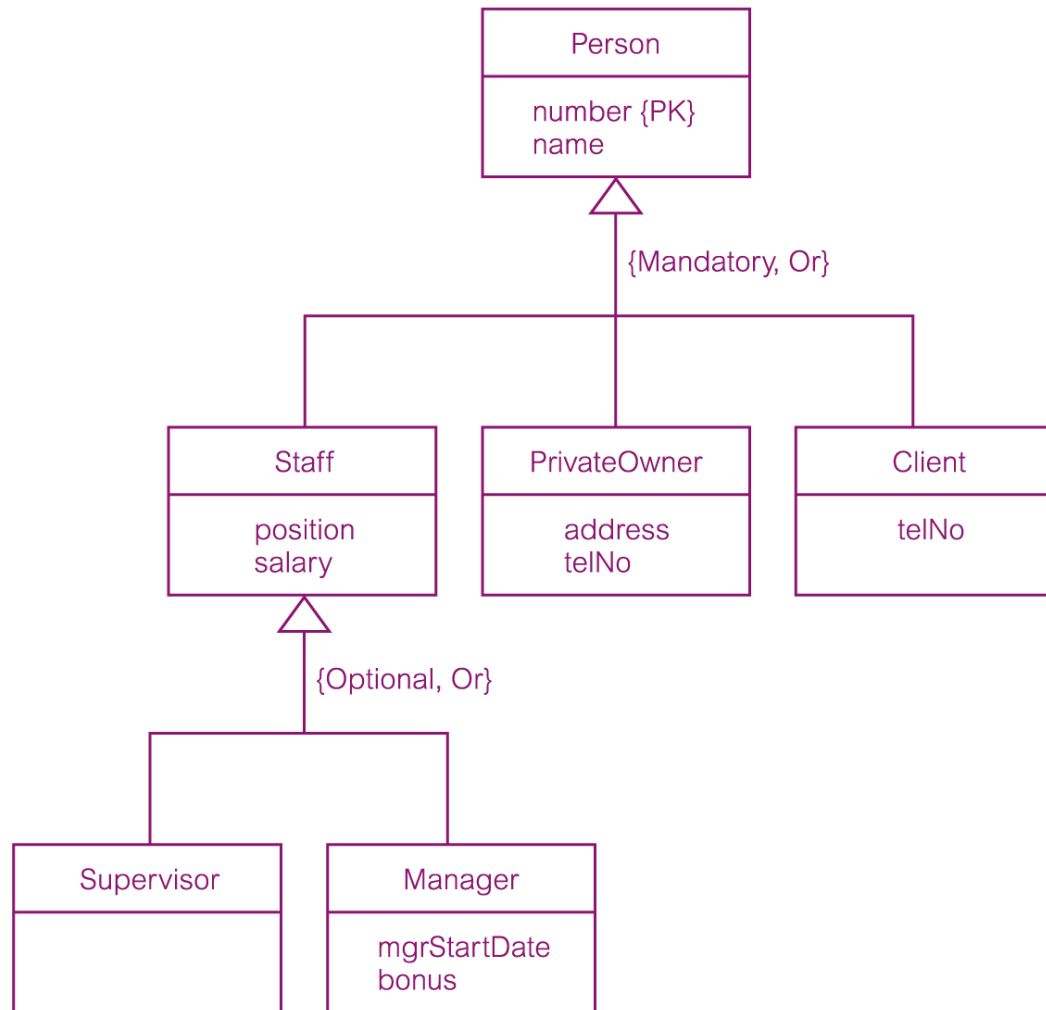
Enhanced Entity-Relationship Modelling

- Semantic concepts are incorporated into the original ER model and called the Enhanced Entity-Relationship (EER) model.
- Examples of additional concept of EER model is called specialization / generalization.

Enhanced Entity-Relationship Modelling



Enhanced Entity-Relationship Modelling



Summary



- In this chapter you should have learned:
 - How to use Entity-Relationship (ER) modeling in a database design
 - The basic concepts associated with the Entity-Relationship (ER) model
 - A diagrammatic technique for displaying an ER model using the Unified Modeling Language (UML) or the Crow's Feet Notation.
 - How to identify and resolve some problems with ER models
 - How to build an ER model from a requirements specification

