Database Administrator

The person who has central control over the system.

- Schema definition
- Schema and physical organization
- Granting of authorization for data access
- Routine maintenance

Entity Relationship Data Model

Introduction:

Building a DBS is complex process

1. Requirement analysis
2. Design

Data model

3. Implementation

Data Model: A set of concepts to describe the *structure* of a database, and certain *constraints* that the database should obey.

Structure of database includes

Data types

Data relationships

Data constraints

Benefits of Data Modelling

1. Focusing on essentials:

develope simple and abstract view of process
ignore details destracting essential features
e.g. In library the fact that books have several editions
may be ignored

2. Ease of communication and understanding:

communication among all parties involved to understand and
document what is being modelling
e.g. In library database system communication between
database
designer, librarians and users(students, professors)

Benefits of Data Modelling

3. Product or process improvement:

Communication among various stakeholders to improve process

4. Exploring alternatives:

build prototype can assist in exploring and evaluating alternatives

e.g allowing students to enroll online in university if is not been done earlier

Types of models

1. Descriptive: aim is to describe and understand what is being modelled

   e.g. Investment companies build models of share market
   to understand it and to predict share prices

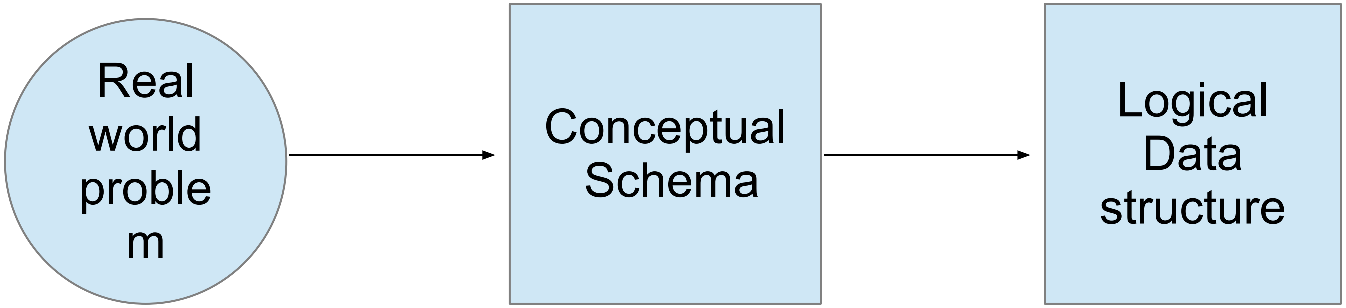
2. Prescriptive: aim is to specify exaclty what is being modelled

   e.g. building prototypes

3. Representative: aim is to simulate what is being modelled

The model need not belong to one class
Prescriptive models are normally used for database modelling.
As they specify what DB system should do although they serve descriptive
Role during communication between DB designer and customer

Phases of database modeling



Real
world
proble
m

Conceptual
Schema

Logical
Data
structure

Data abstraction:
Only what data will be
there

Data abstraction:
Which data model to use

Data Modeling Using the Entity-Relationship Model

Entity-Relationship(ER) Model

The ER model

- a high-level conceptual data model

- not been implemented in any commercial DBMS

- but is a powerful short hand often used in database design for a first rendition of the mini world.

- The ER model was introduced by Peter Chen in 1976, and is now the most widely used conceptual data model.

E – R Model

Example Database Application (COMPANY)

ER Model Concepts

Entities and Attributes

Entity Types, Value Sets, and Key Attributes

Relationships and Relationship Types

Weak Entity Types

Roles and Attributes in Relationship Types

ER Diagrams - Notation

ER Diagram for COMPANY Schema

Alternative Notations – UML class diagrams, others

Example COMPANY Database

Requirements of the Company (oversimplified for illustrative purposes)

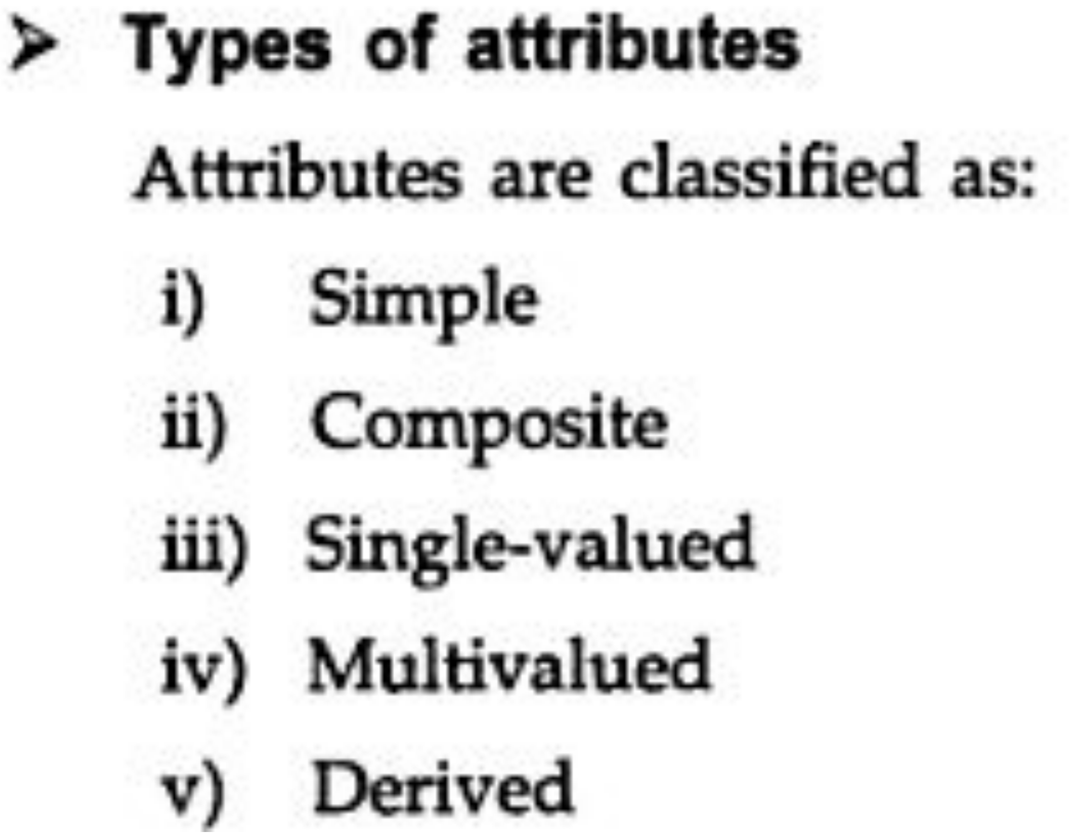
The company is organized into DEPARTMENTs. Each department has a name, number and an employee who *manages* the department. We keep track of the start date of the department manager.

Each department *controls* a number of PROJECTs*.* Each project has a name, number and is located at a single location.

Example COMPANY Database (Cont.)

- We store each EMPLOYEE’s social security number, address, salary, gender, and birthdate. Each employee *works for* one department but may *work on* several projects. We keep track of the number of hours per week that an employee currently works on each project. We also keep track of the *direct supervisor* of each employee.
- Each employee may *have* a number of DEPENDENTs. For each dependent, we keep track of their name, gender, birthdate, and relationship to employee.

E-R model concepts



- **Types of attributes**

Attributes are classified as:

i) Simple

ii) Composite

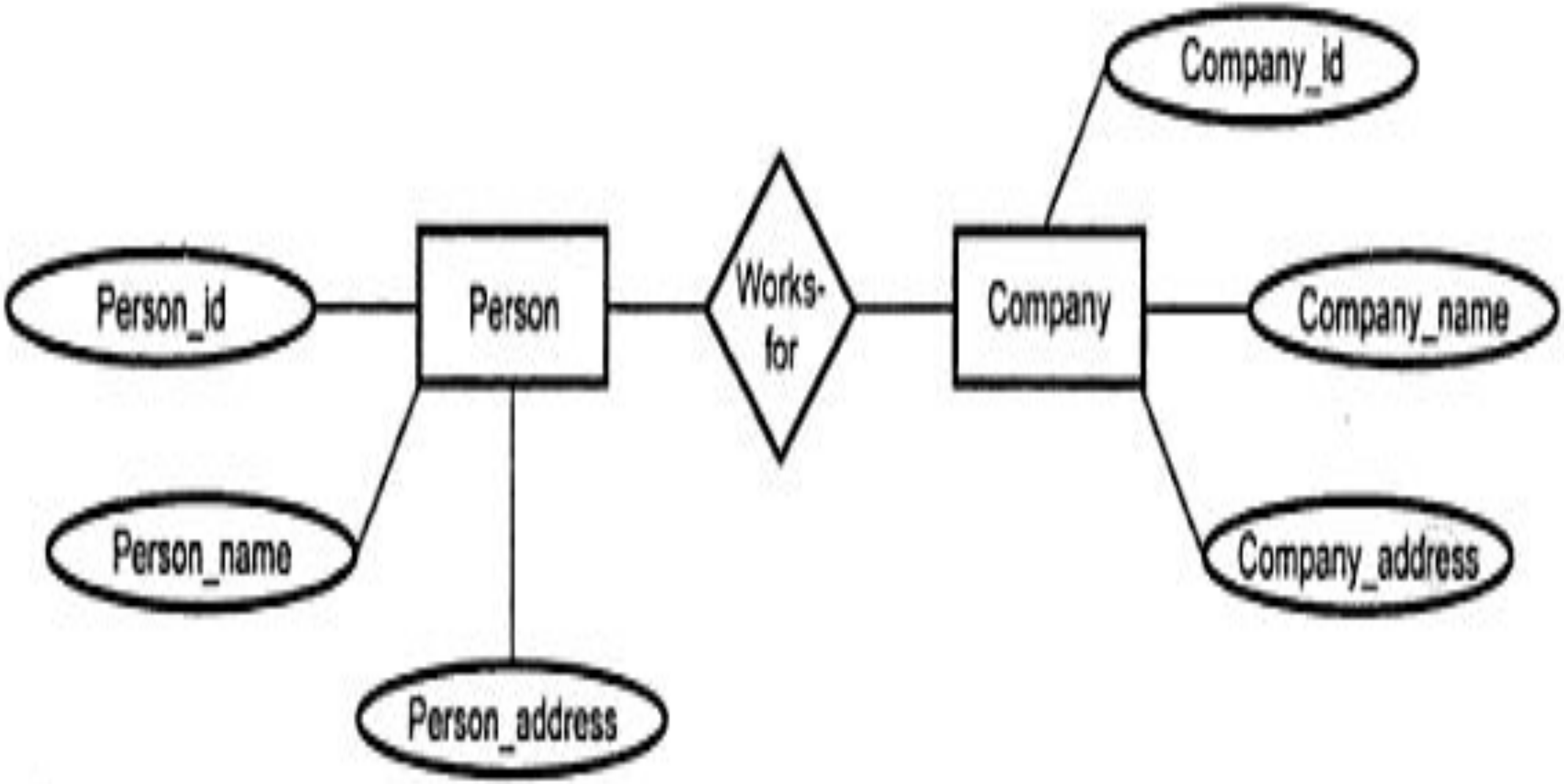
iii) Single-valued

iv) Multivalued

v) Derived

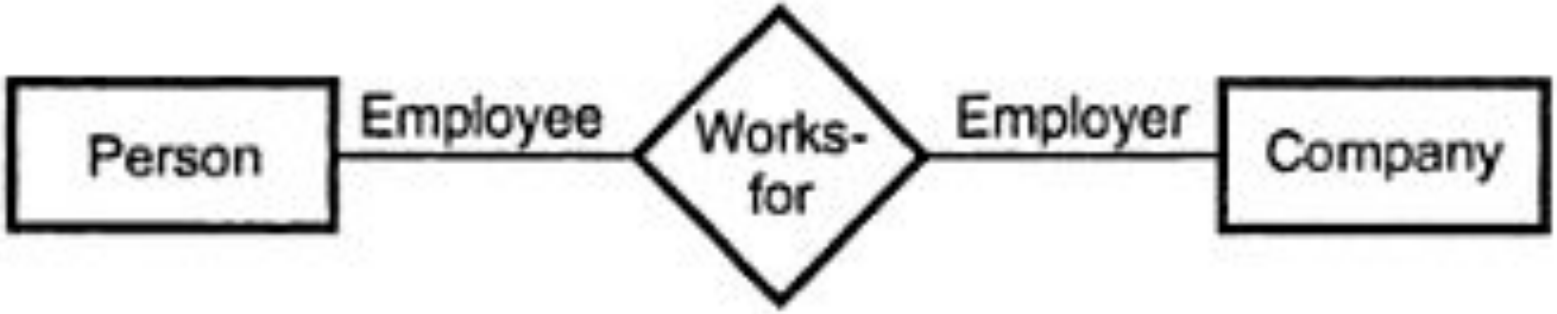
E-R model concepts

Relationship and Relationship set



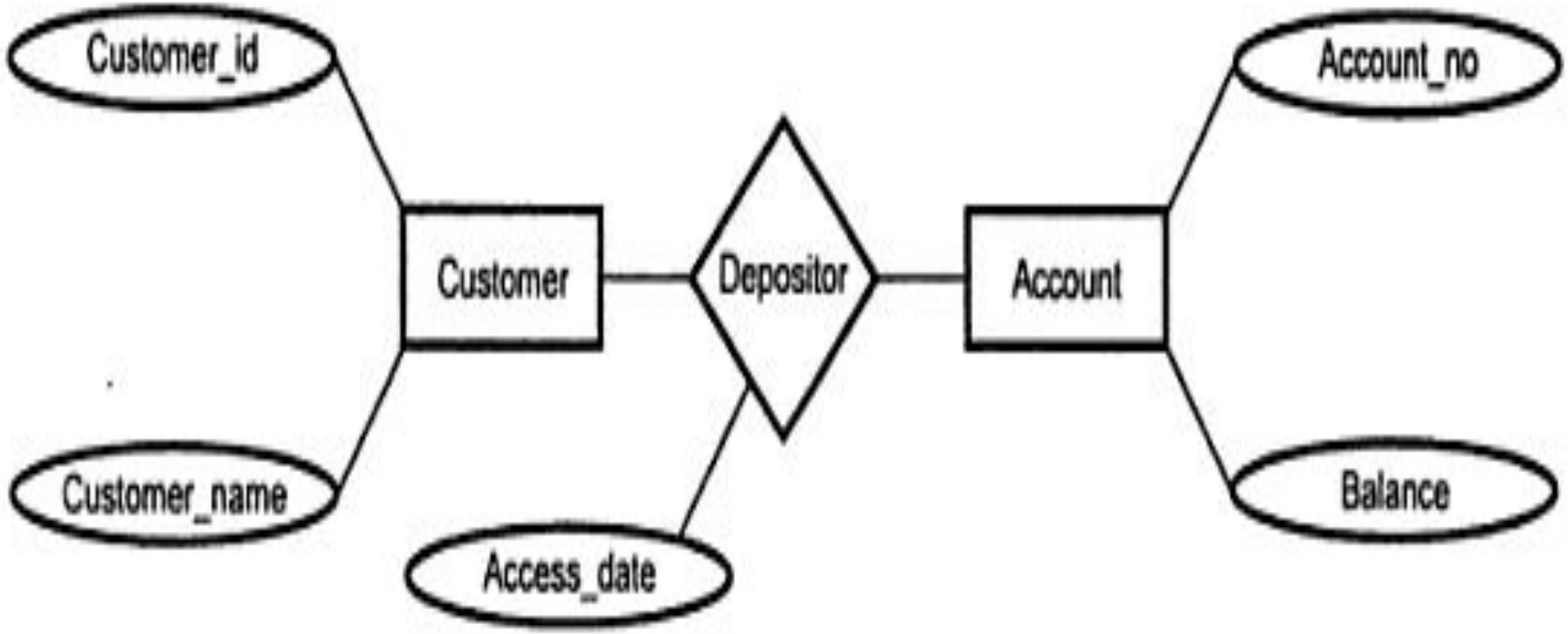
E-R model concepts

Entity Role



E-R model concepts

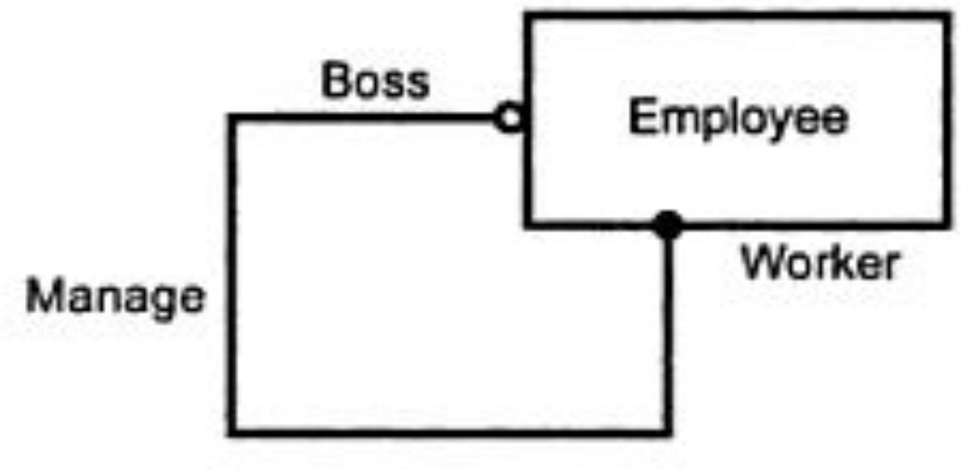
Descriptive attribute



E-R model concepts

Relationship and Relationship set

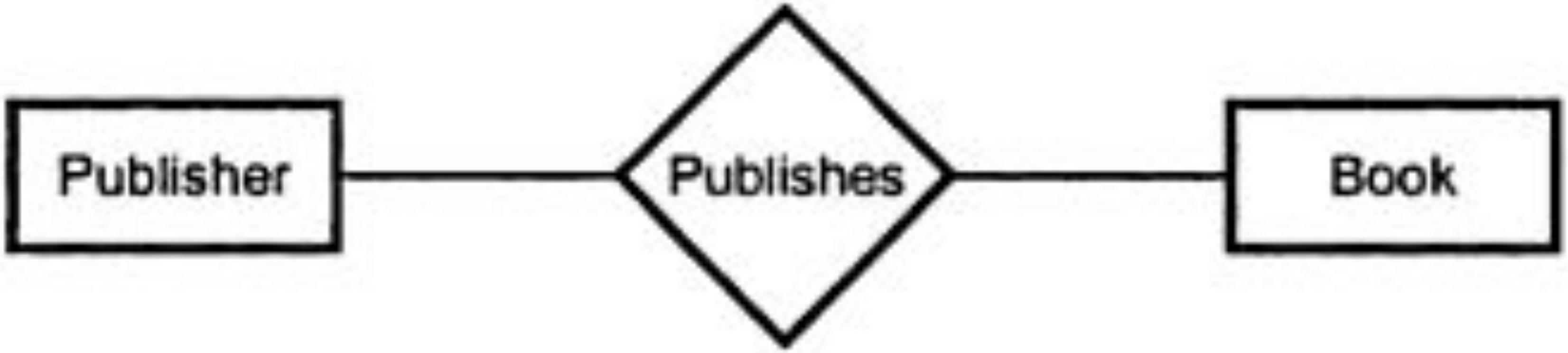
Unary relationship



E-R model concepts

Relationship and Relationship set

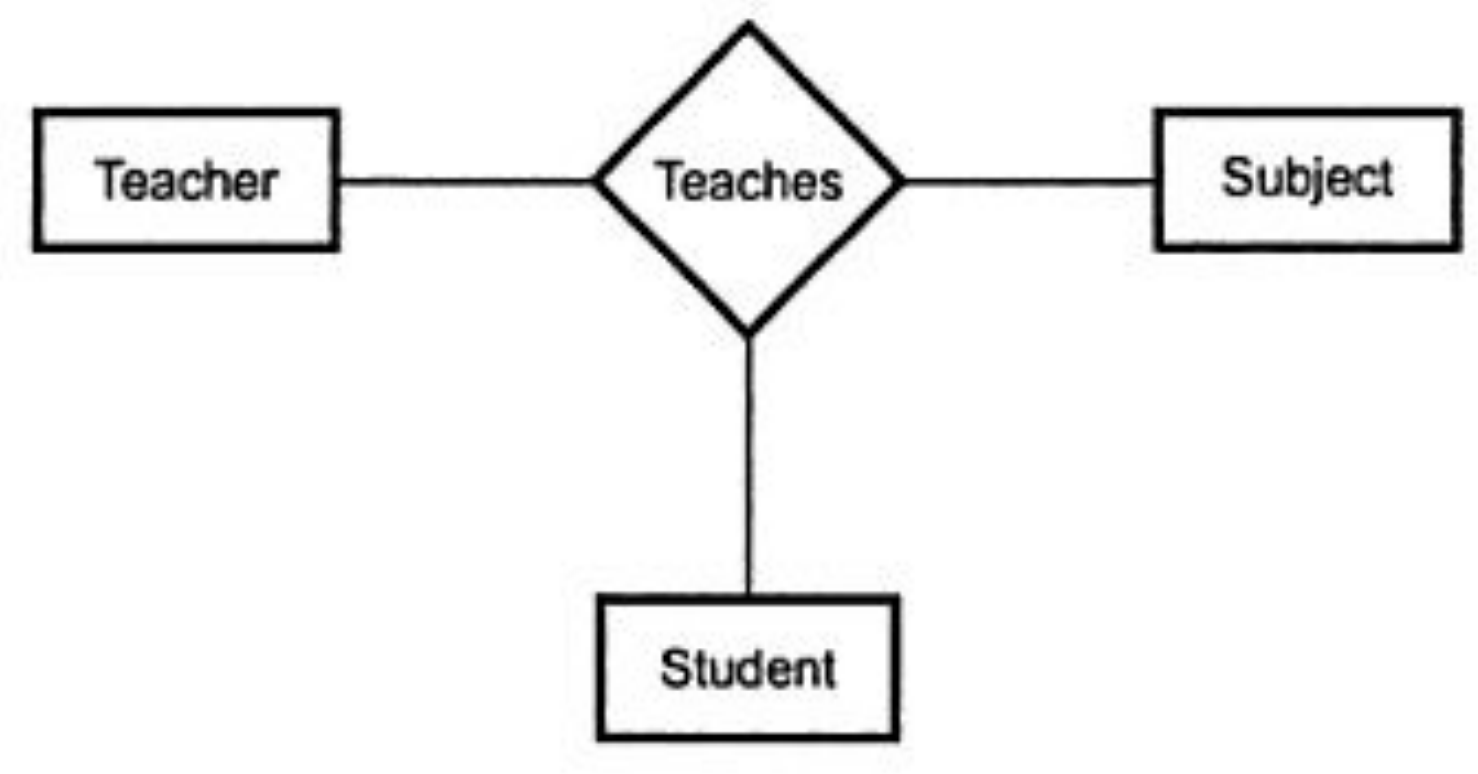
Binary relationship



E-R model concepts

Relationship and Relationship set

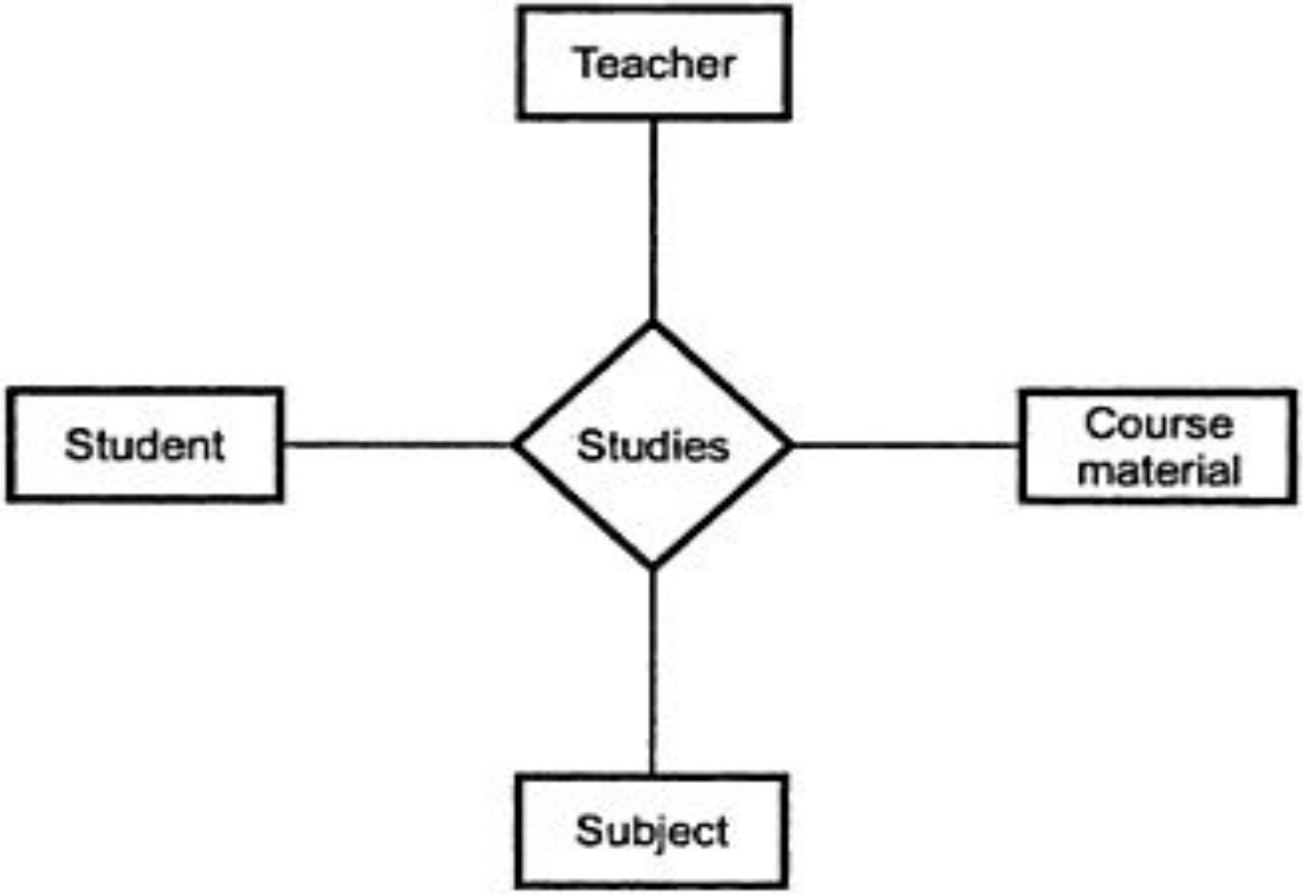
Ternary relationship



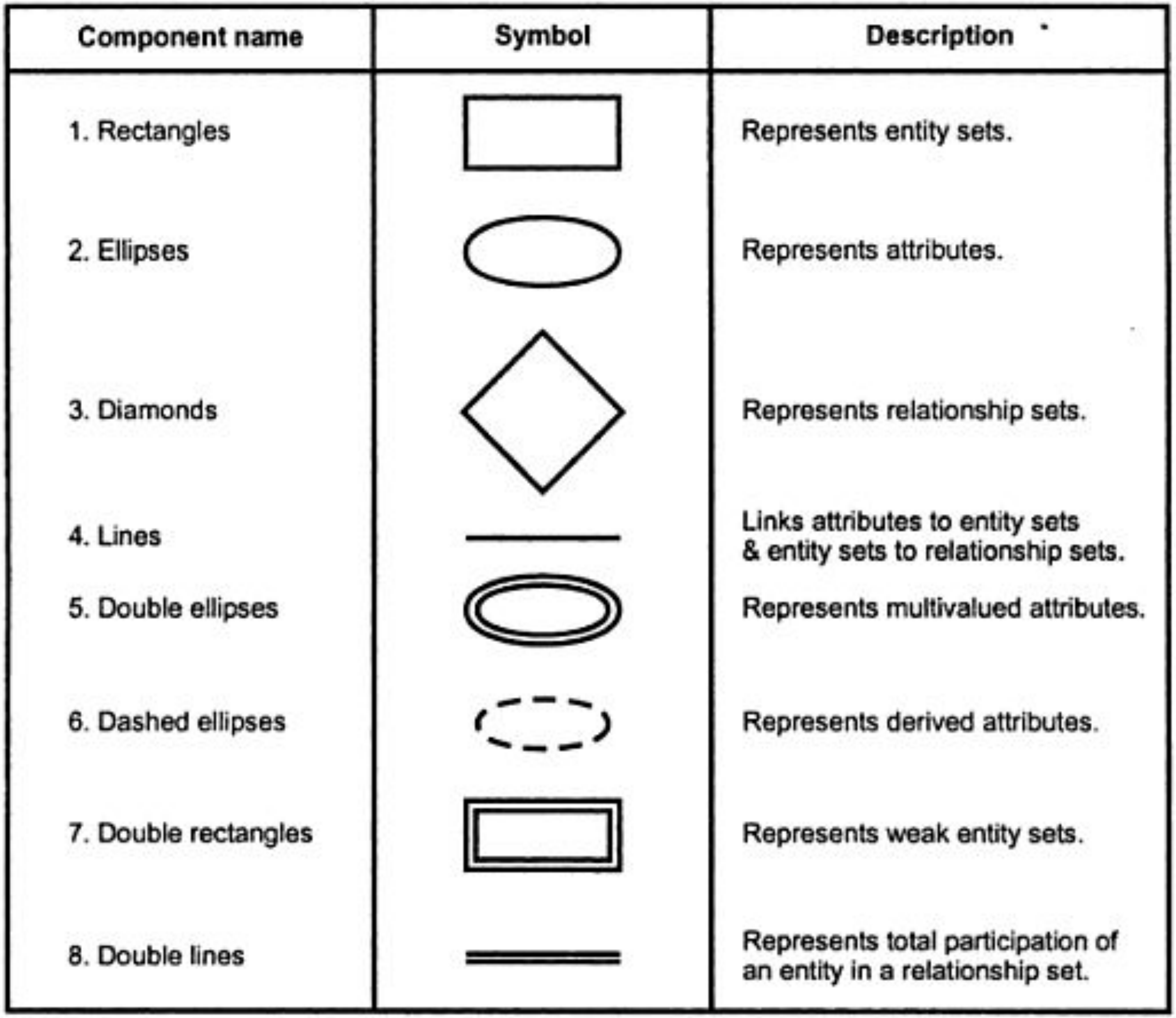
E-R model concepts

Relationship and Relationship set

Quaternary relationship



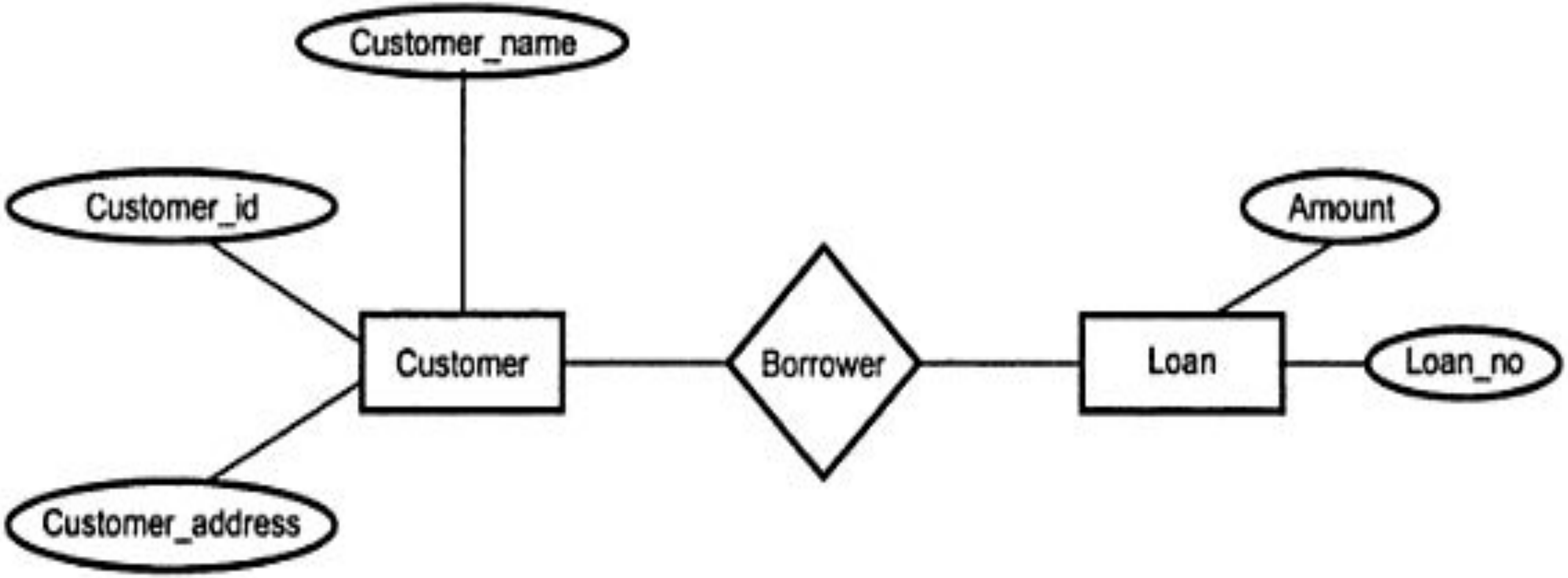
E-R model concepts



Component name	Symbol	Description
1. Rectangles	▭	Represents entity sets.
2. Ellipses	⬭	Represents attributes.
3. Diamonds	◇	Represents relationship sets.
4. Lines	—	Links attributes to entity sets & entity sets to relationship sets.
5. Double ellipses	◎	Represents multivalued attributes.
6. Dashed ellipses	⬯	Represents derived attributes.
7. Double rectangles	⧈	Represents weak entity sets.
8. Double lines	═	Represents total participation of an entity in a relationship set.

E-R model concepts

ER diagram of bank



Mapping cardinality Constraints

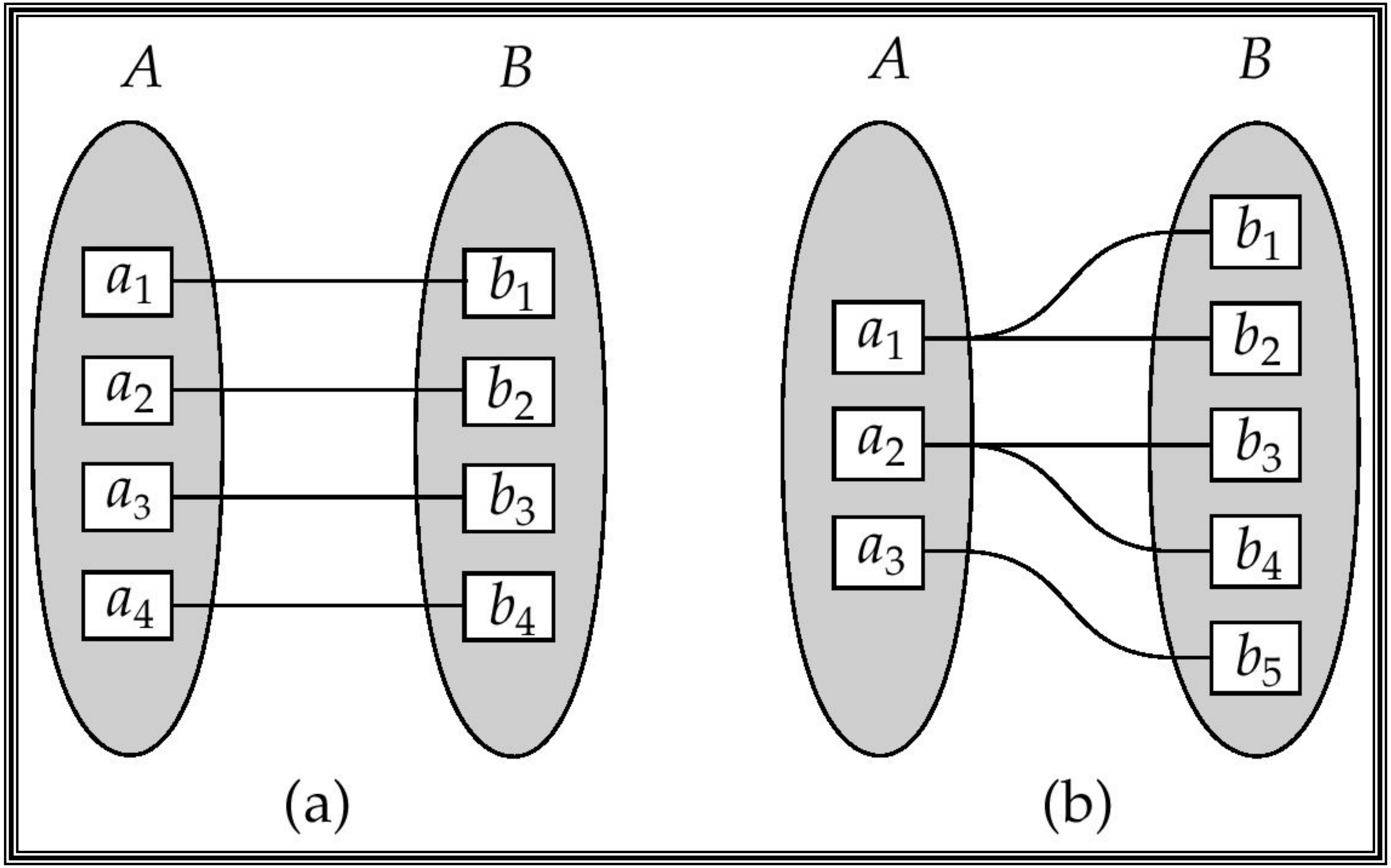
- Express the number of entities to which another entity can be associated via a relationship set.

1-1

An entity in A is associated with at most one entity in B,and an entity in B is associated at most one entity in A

- 1-M
- M-1
- M-M

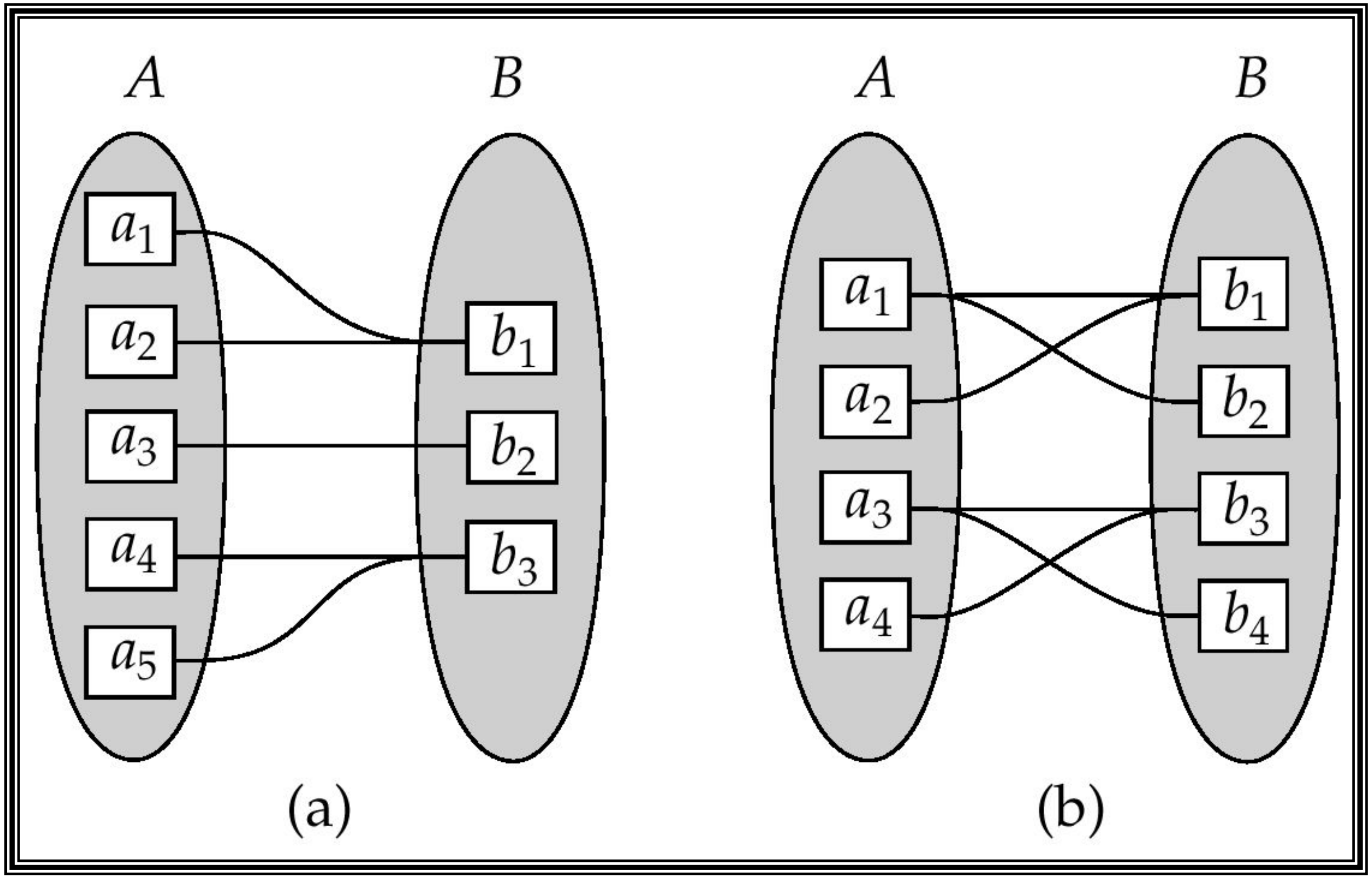
Mapping cardinality Constraints



One to one

One to many

Mapping cardinality Constraints



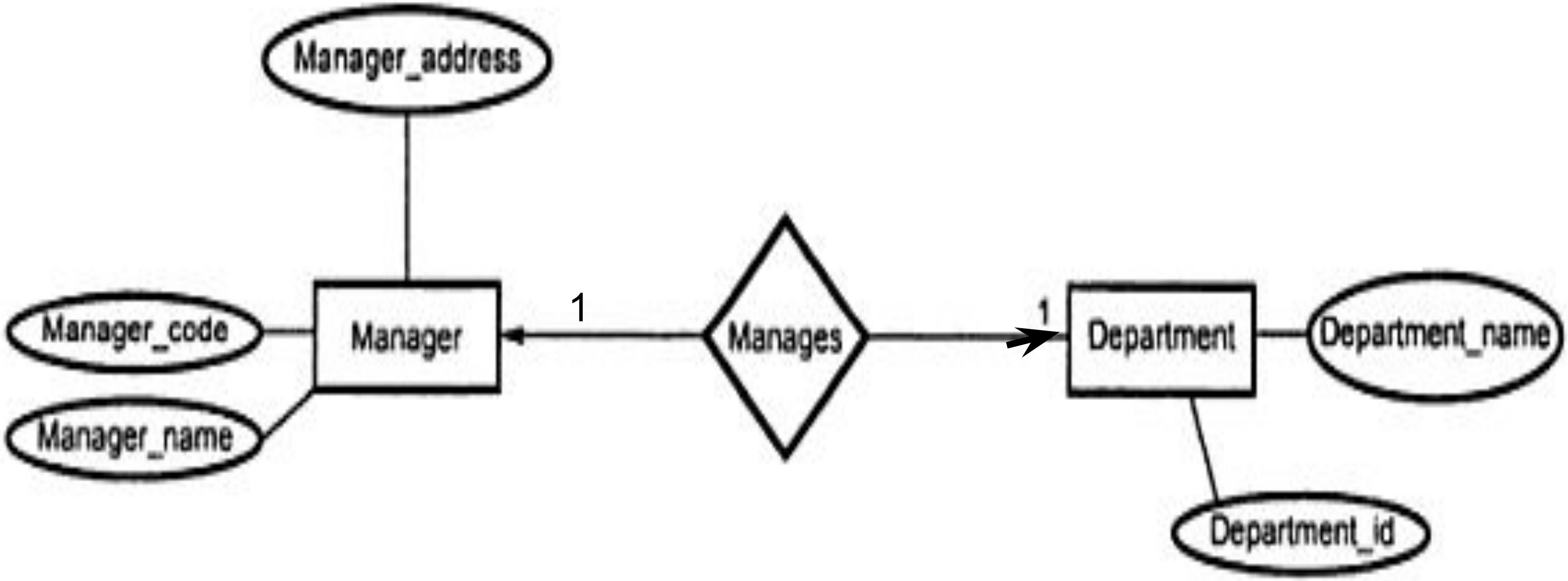
Many to one

Many to many

E-R model concepts

Constraints

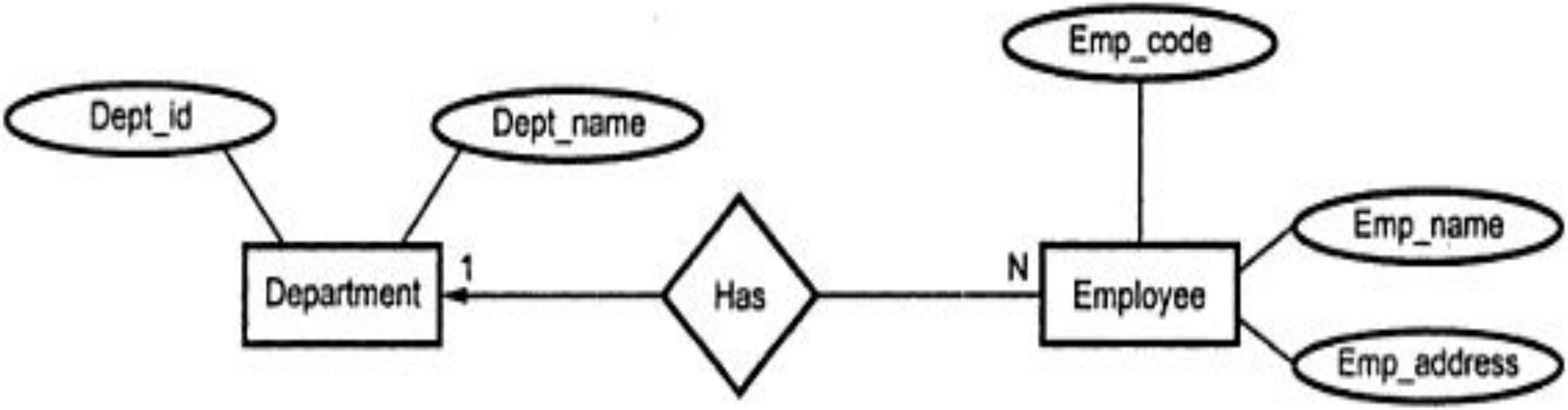
Mapping cardinality Constraints 1 to 1



E-R model concepts

Constraints

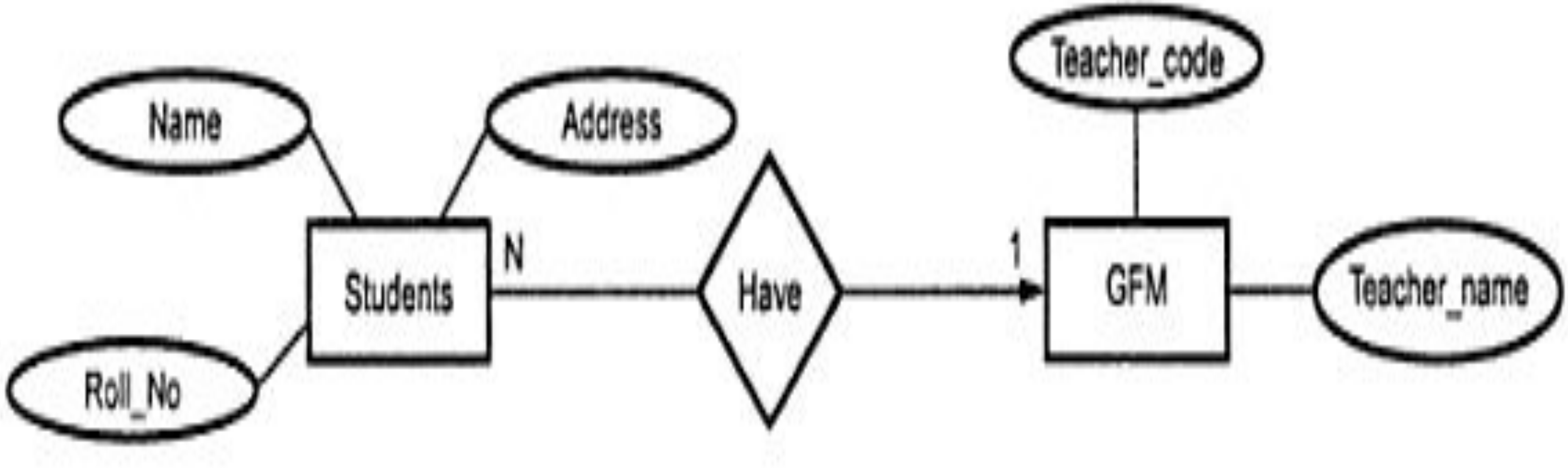
Mapping cardinality Constraints 1 to M



E-R model concepts

Constraints

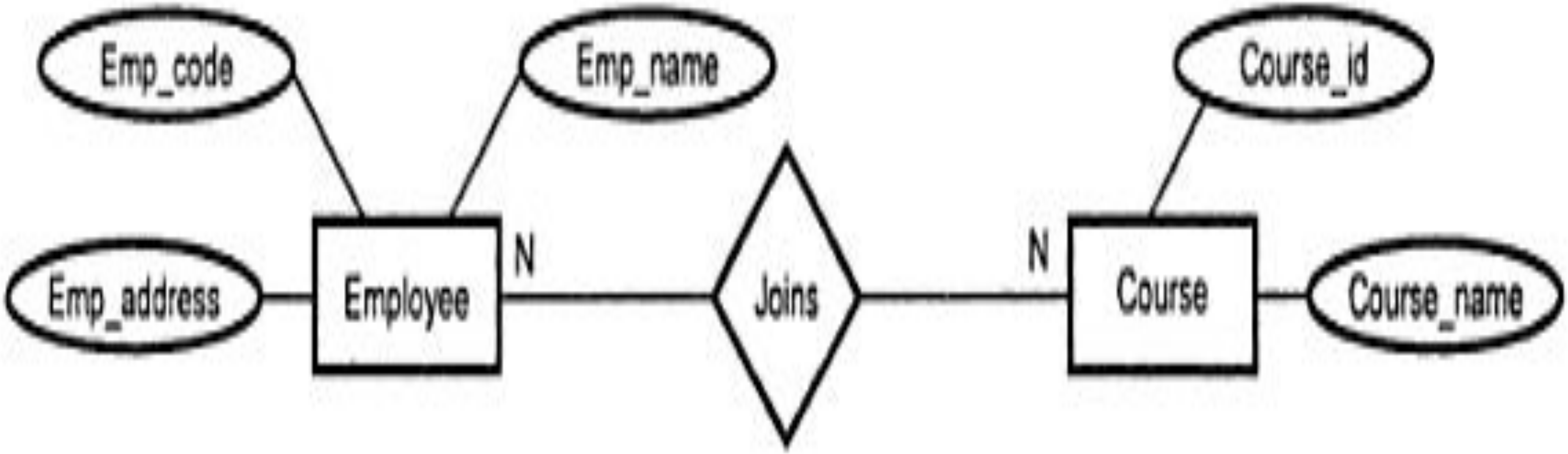
Mapping cardinality Constraints M to 1



E-R model concepts

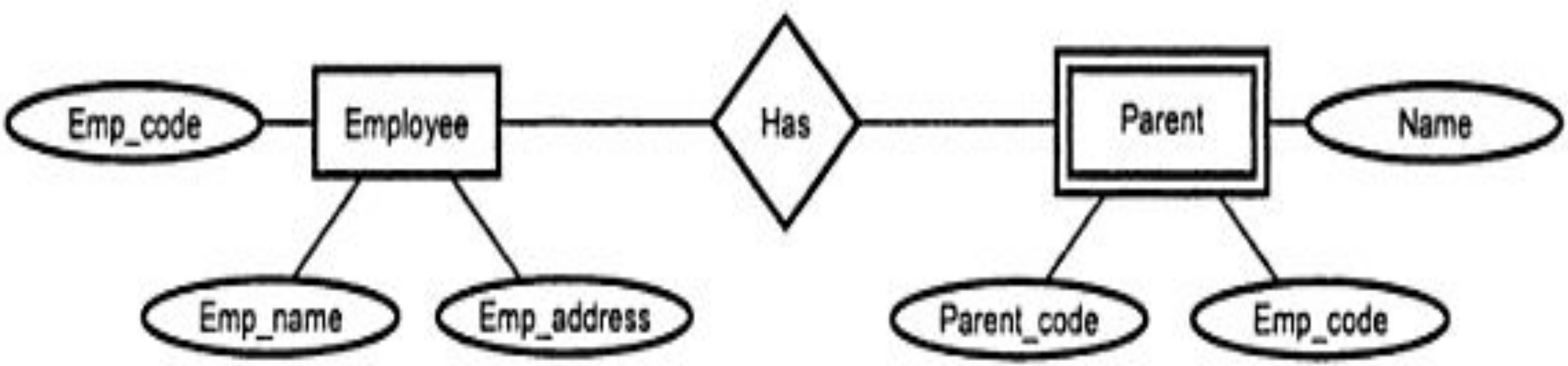
Constraints

Mapping cardinality Constraints M to M



E-R model concepts

Strong and weak entity set



E-R model concepts

An entity is an object in the miniworld.

For example the EMPLOYEE John Smith, the Research DEPARTMENT, the ProductX PROJECT

An attribute of an entity can have a value from a value set (domain)

For example an EMPLOYEE entity may have a Name, SSN, Address, gender, BirthDate

Each attribute has a *value set* (or data type) associated with it – e.g. integer, string, subrange, enumerated type, …

Each entity belongs to some one entity type s.t. entities in one entity type have the same attributes (so each entity type is a set of similar entities).

E-R model concepts (con’t)

A key attribute of an entity type is one whose value uniquely identifies an entity of that type.

For example, SSN of EMPLOYEE.

A combination of attributes may form a composite key.

For example, VehicleTagNumber is a key of the CAR entity type with components (Number, State).

If there is no applicable value for an attribute that attribute is set to a null value.

An entity type may have more than one key.

For example, the CAR entity type may have two keys:

VehicleIdentificationNumber (popularly called VIN) and

VehicleTagNumber (Number, State), also known as license_plate number.

Entity Type / Entity Set

Entity Type (Intension): EMPLOYEE

Attributes: Name, Age, Salary

Entity Set (Extension):

e_1 = (John Smith, 55, 80000)

e_2 = (Joe Doe, 40, 20000)

e_3 = (Jane Doe, 27, 30000)

.

.

.

ENTITY SET corresponding to the ENTITY TYPE CAR

CAR
Registration(RegistrationNumber, State), VehicleID, Make, Model, Year, (Color)

car$_{1}$
((ABC 123, TEXAS), TK629, Ford Mustang, convertible, 1999, (red, black))
car$_{2}$
((ABC 123, NEW YORK), WP9872, Nissan 300ZX, 2-door, 2002, (blue))
car$_{3}$
((VSY 720, TEXAS), TD729, Buick LeSabre, 4-door, 2003, (white, blue))

.

.

.

Attributes

Attributes can be

composite / simple (atomic)

single-valued / multivalued

stored / derived

key / nonkey.

Attribute Examples

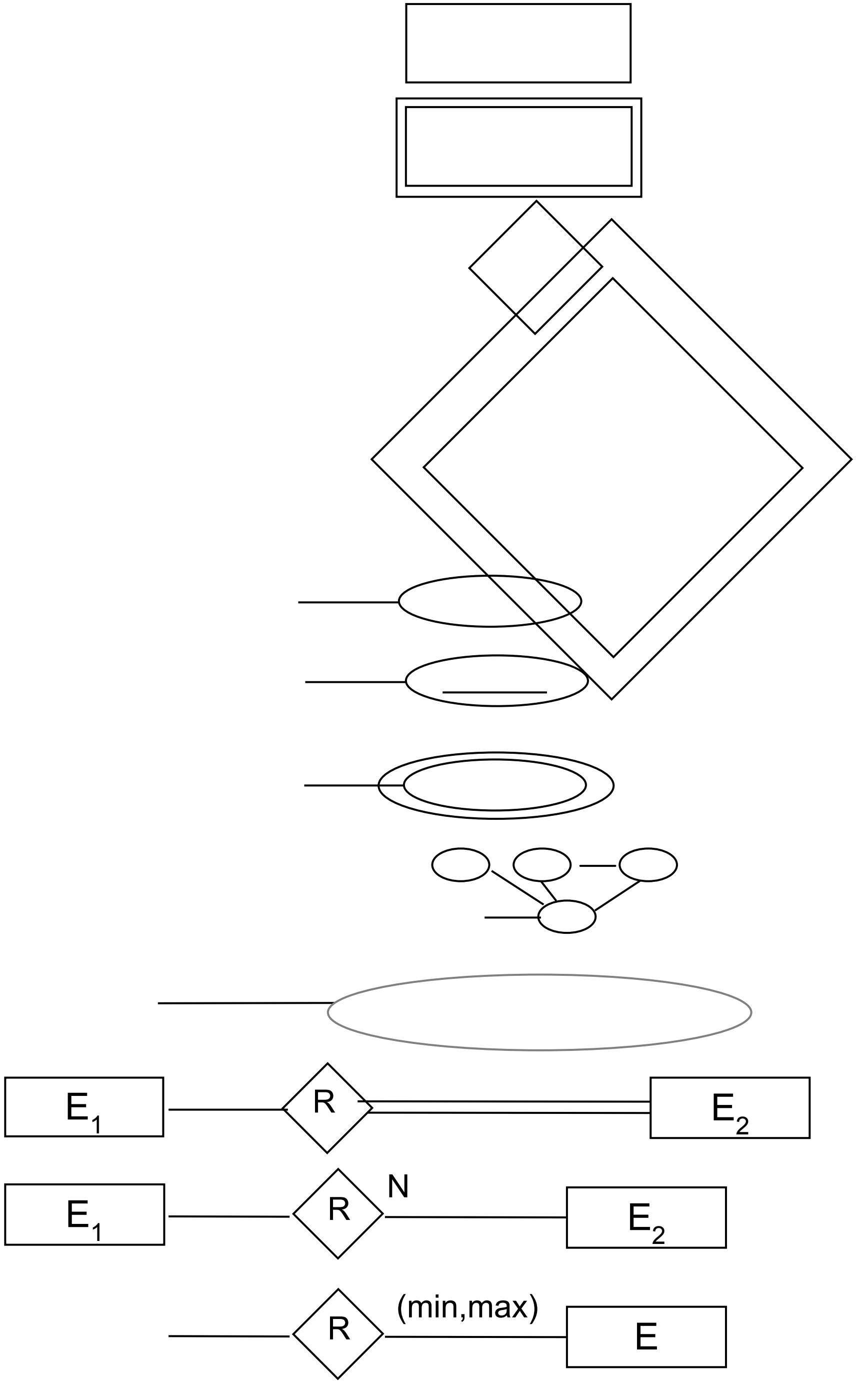
Name = John Doe
Birthdate = May 10, 1989
Age = 42
Degree = null
SSN = 123456789

Name = Jane Doe
Birthdate = May 10, 1989
Age = 42
Degree = B.S., M.S.
SSN = 987654321

Name = John Doe
Birthdate = May 10
Birthyear = 1989
Age = 42
Degree = null
SSN = 123456789

NOTATION FOR ER SCHEMAS

Symbol



Meaning

ENTITY TYPE

WEAK ENTITY TYPE

RELATIONSHIP TYPE

IDENTIFYING RELATIONSHIP TYPE

ATTRIBUTE

KEY ATTRIBUTE

MULTIVALUED ATTRIBUTE

COMPOSITE ATTRIBUTE

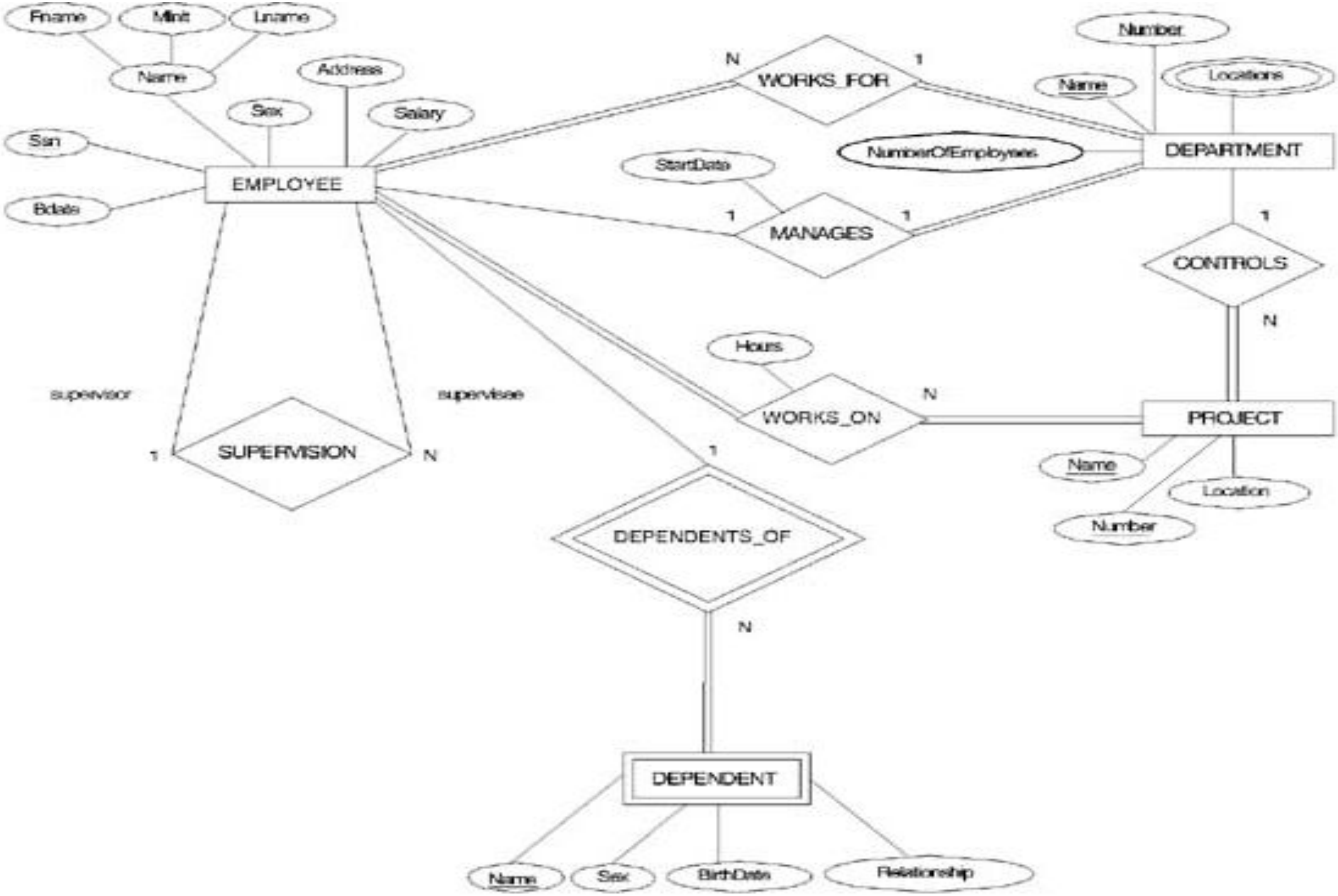
DERIVED ATTRIBUTE

TOTAL PARTICIPATION OF E_2 IN R

CARDINALITY RATIO 1:N FOR E_1:E_2 IN R

STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R

ER DIAGRAM – Entity Types are:
EMPLOYEE, DEPARTMENT, PROJECT, DEPENDENT



Relationships and Relationship Types (1)

A relationship relates two or more distinct entities with a specific meaning.

For example, EMPLOYEE John Smith works on the ProductX PROJECT

EMPLOYEE Franklin Wong manages the Research

DEPARTMENT.

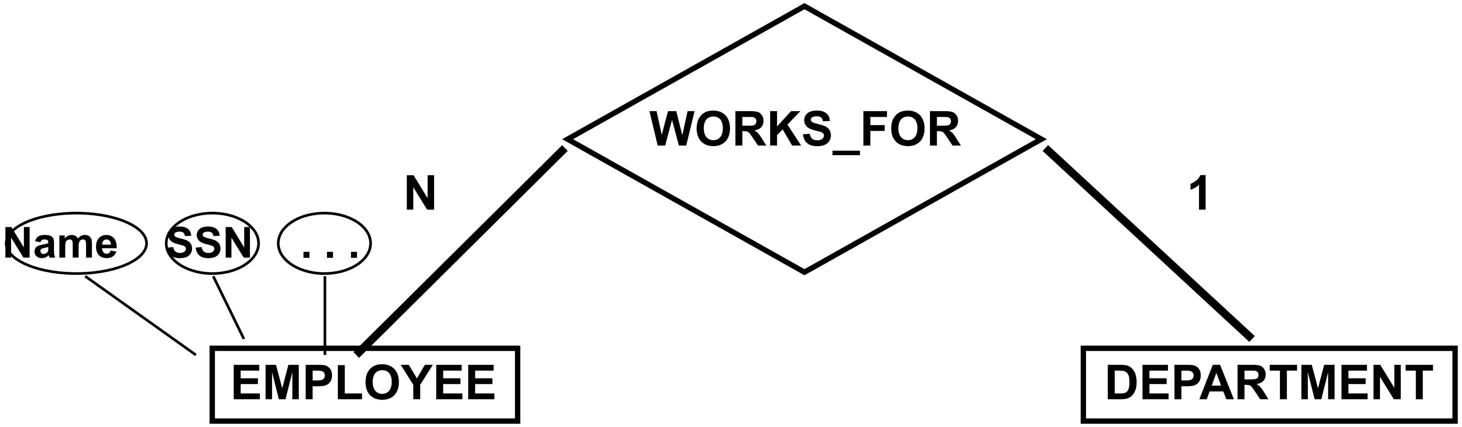
The degree of a relationship type is the number of participating entity types.

Both MANAGES and WORKS_ON are binary relationships.

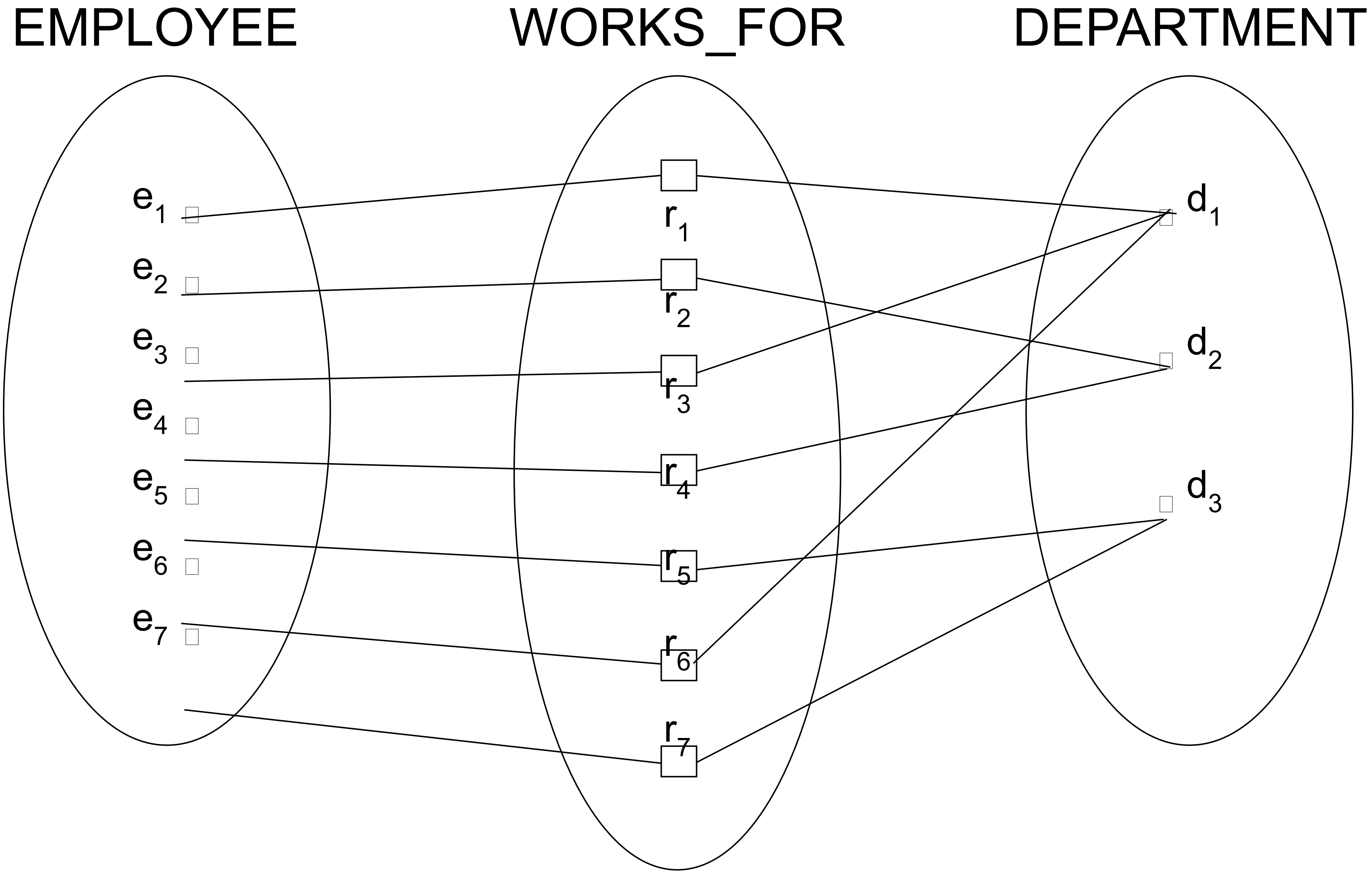
Relationships and Relationship Types

Relationships of the same type are grouped or typed into a relationship type.

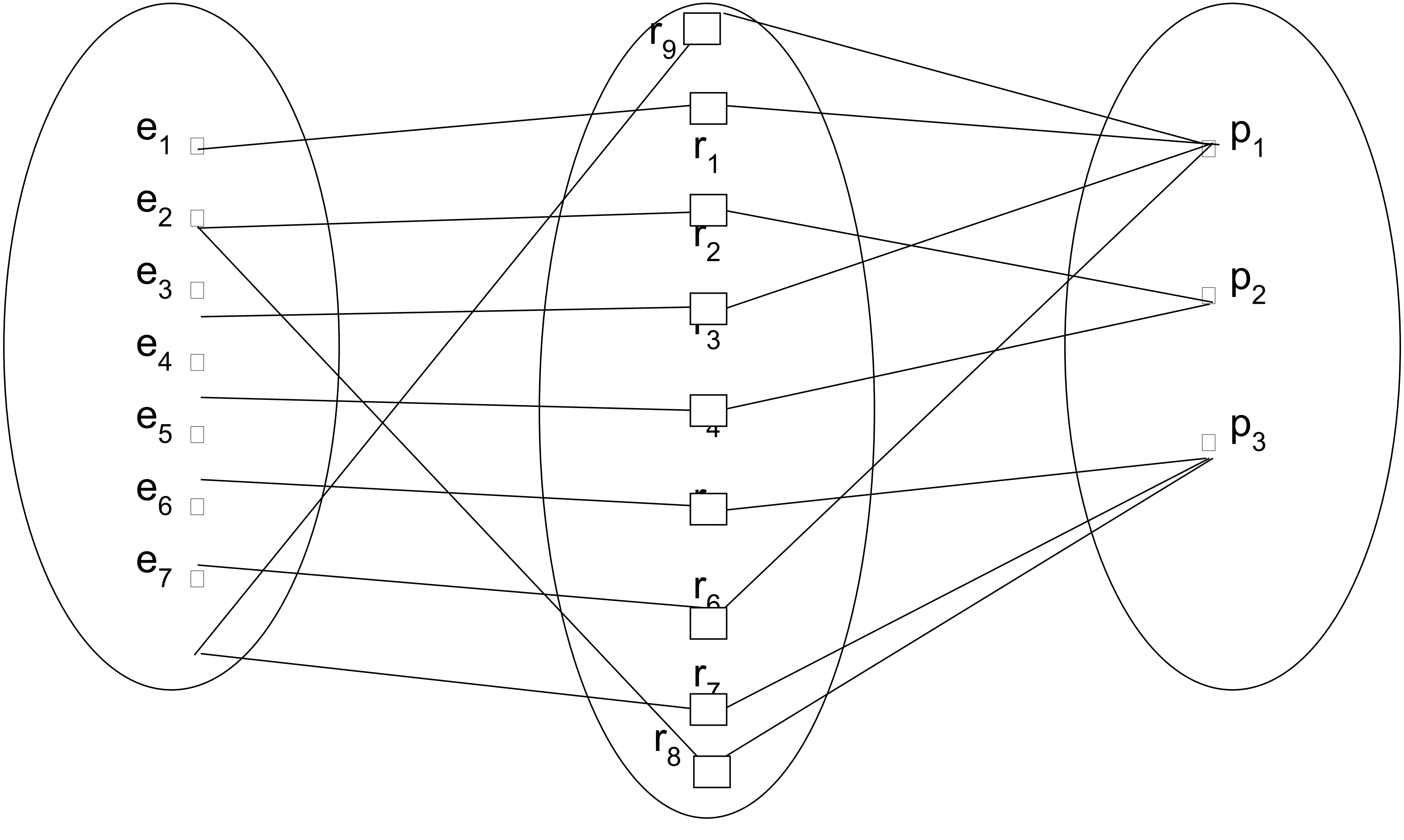
For example, the WORKS_FOR relationship type in which EMPLOYEEs and DEPARTMENTs participate



Example relationship instances of the WORKS_FOR relationship between EMPLOYEE and DEPARTMENT



Example relationship instances of the WORKS_ON relationship between EMPLOYEE and PROJECT



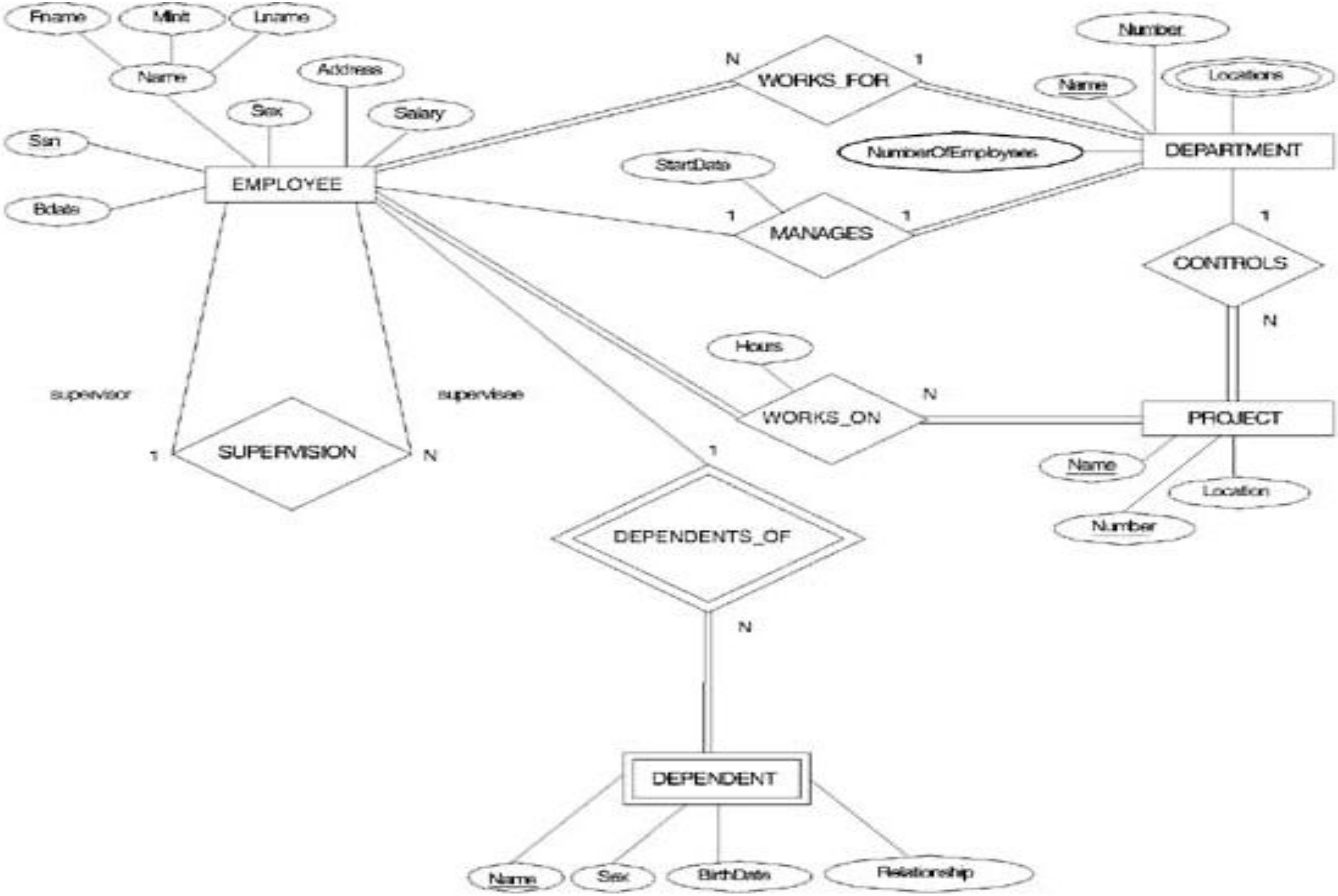
Relationships and Relationship Types (2)

More than one relationship type can exist with the same participating entity types.

For example, MANAGES and WORKS_FOR are distinct relationships between EMPLOYEE and DEPARTMENT,

but with different meanings and different relationship instances.

ER DIAGRAM – Relationship Types are: WORKS_FOR, MANAGES, WORKS_ON, CONTROLS, SUPERVISION, DEPENDENTS_OF



Weak Entity Types

An entity that does not have a key attribute

A weak entity must participate in an identifying relationship type with an owner or identifying entity type

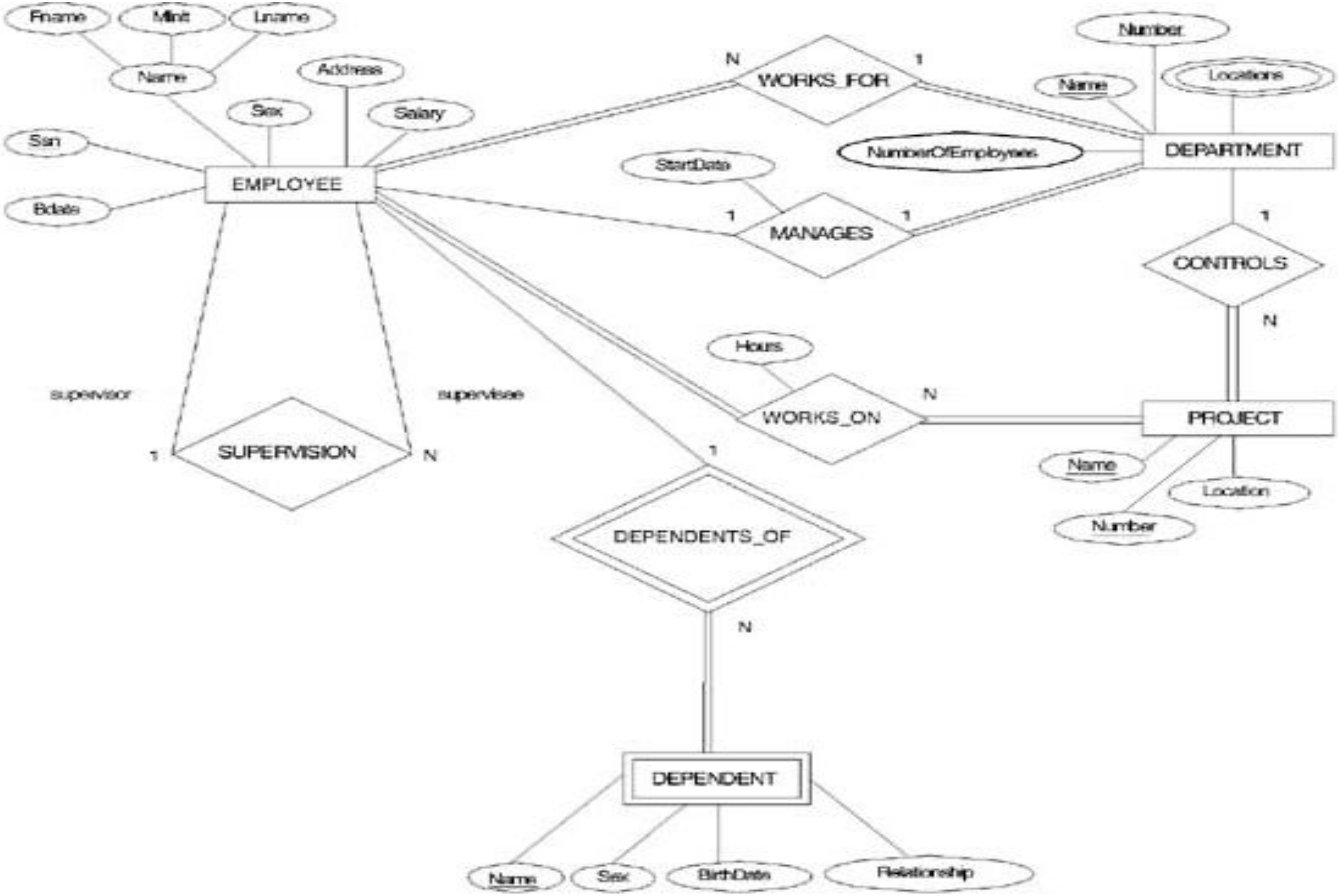
Entities are identified by the combination of:

- A partial key of the weak entity type
- The particular entity they are related to in the identifying entity type

Example:

Suppose that a DEPENDENT entity is identified by the dependent's first name and birthdate, *and* the specific EMPLOYEE that the dependent is related to. DEPENDENT is a weak entity type with EMPLOYEE as its identifying entity type via the identifying relationship type DEPENDENT_OF

Weak Entity Type is: DEPENDENT
Identifying Relationship is: DEPENDENTS_OF



Structural Constraints on Relationships

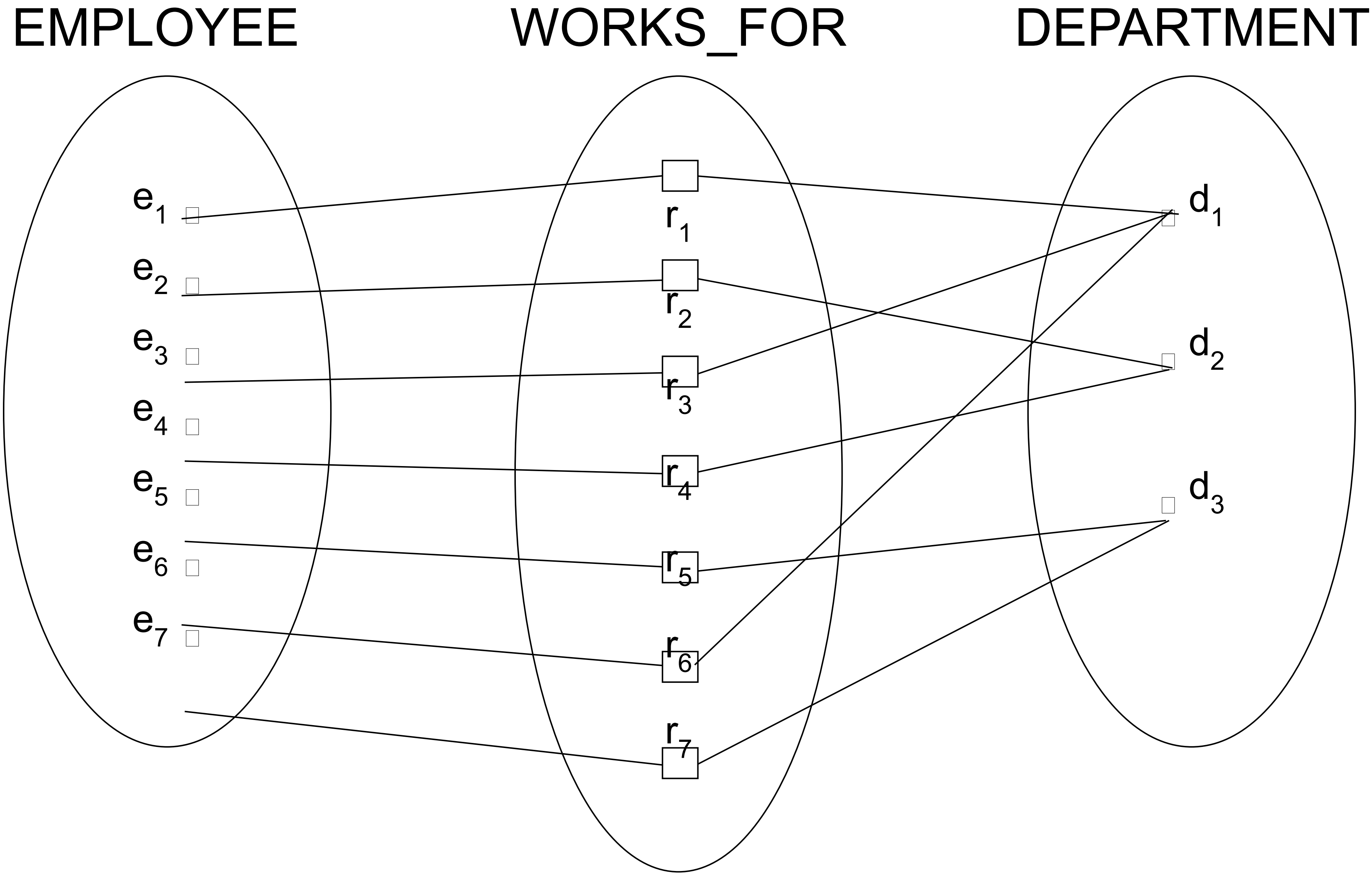
Cardinality constraints:

- One-to-one (1:1)
- One-to-many (1:N)
- Many-to-one (N:1)
- Many-to-many

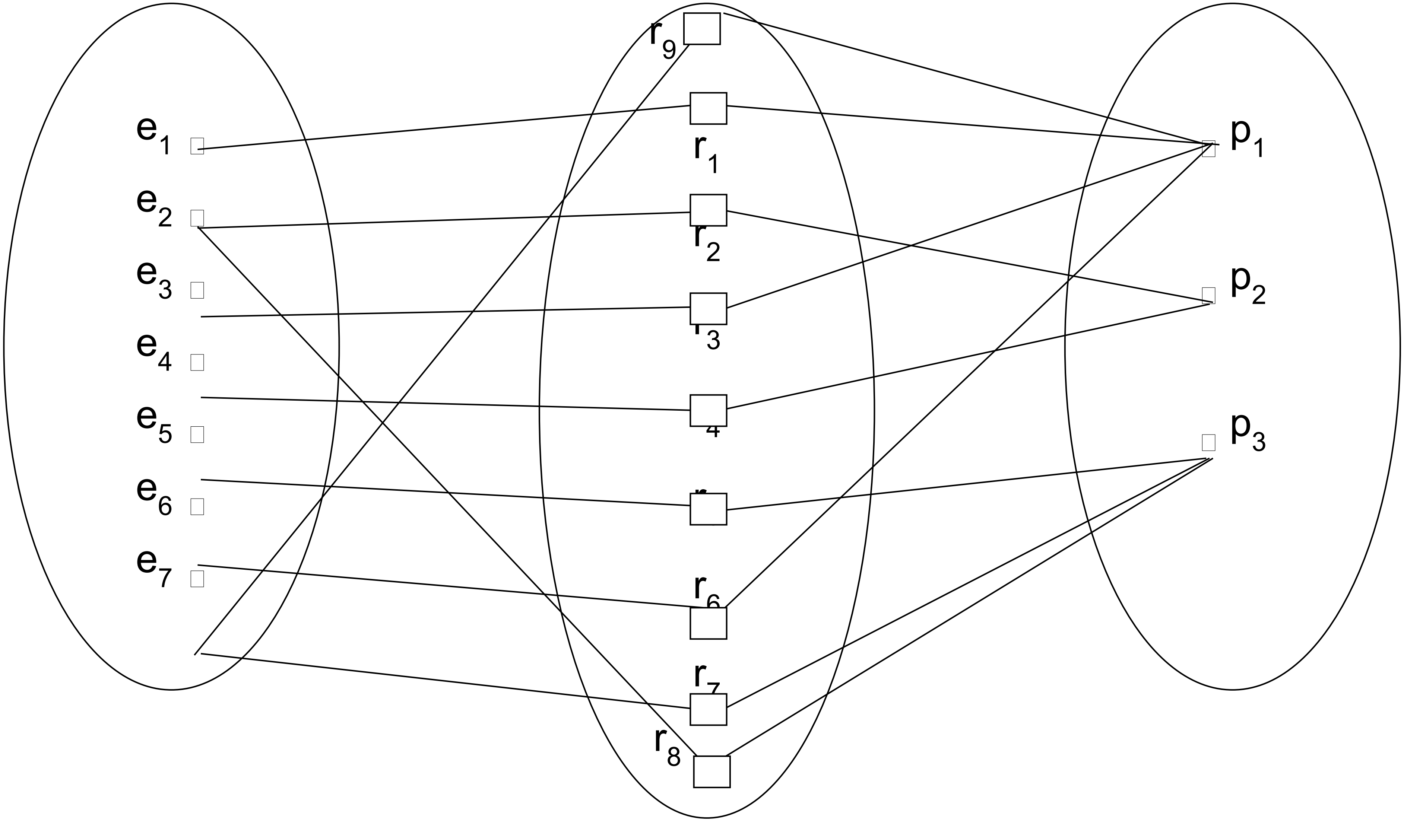
participation constraint:

- Total participation
- Partial participation

Many-to-one (N:1) RELATIONSHIP



Many-to-many (M:N) RELATIONSHIP



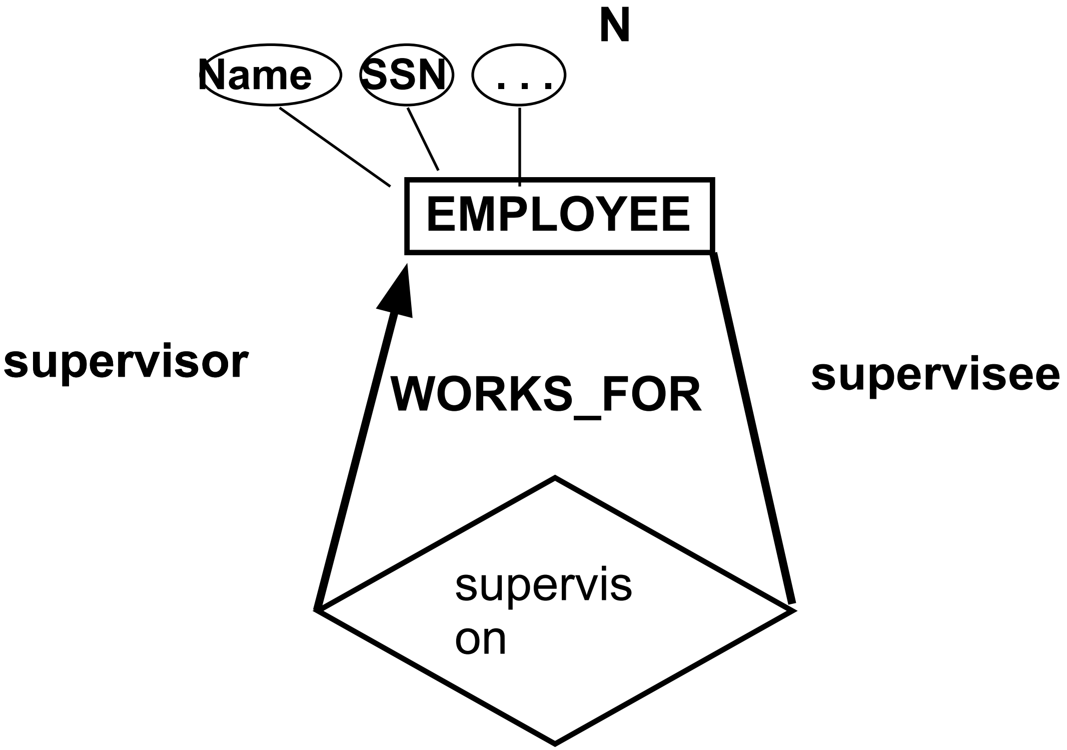
Relationships and Relationship Types (3)

We can also have a **recursive** relationship type.

Both participations are same entity type in different roles.

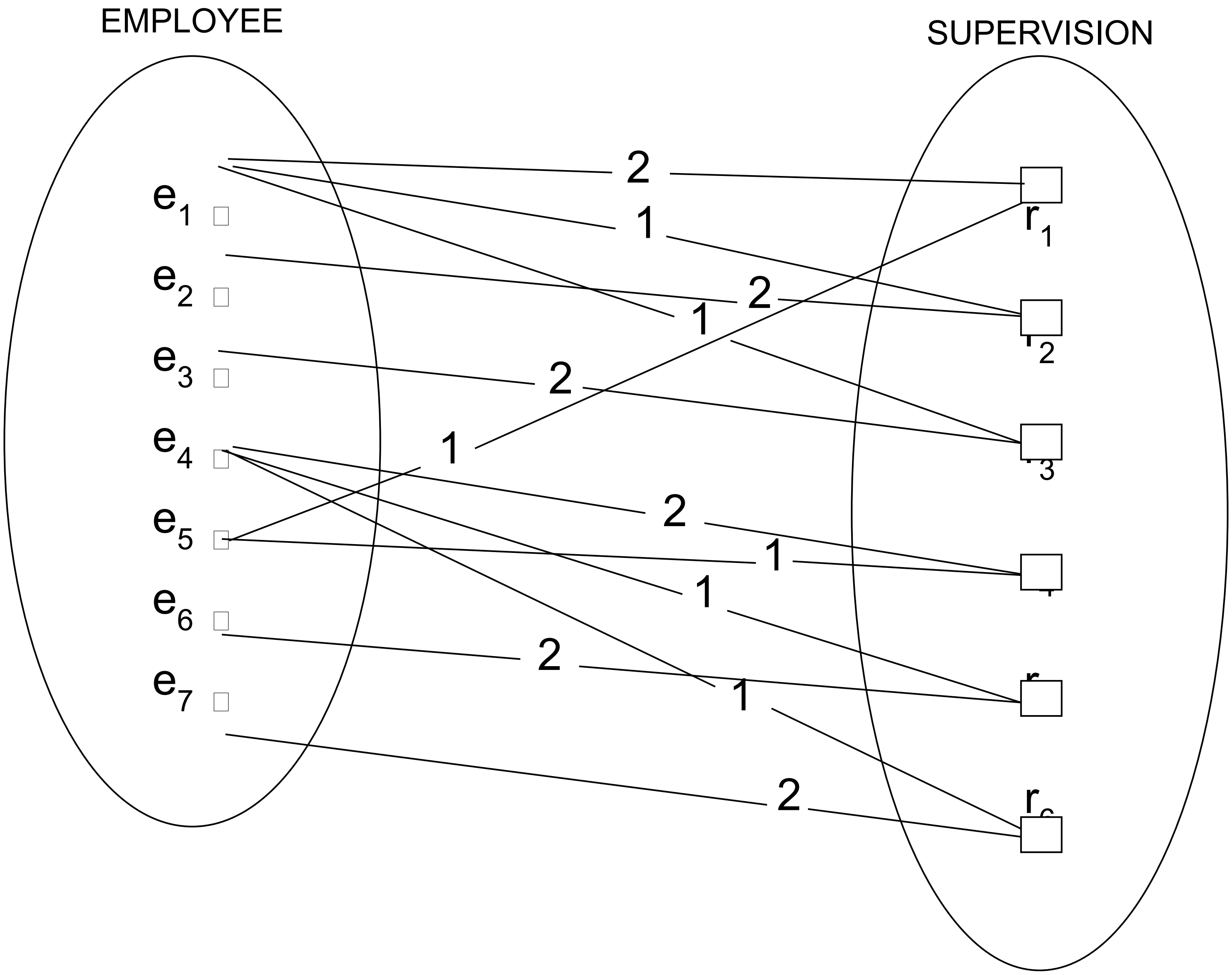
For example, SUPERVISION relationships between EMPLOYEE (in role of supervisor or boss) and (another) EMPLOYEE (in role of subordinate or worker).

In ER diagram, need to display role names to distinguish participations.



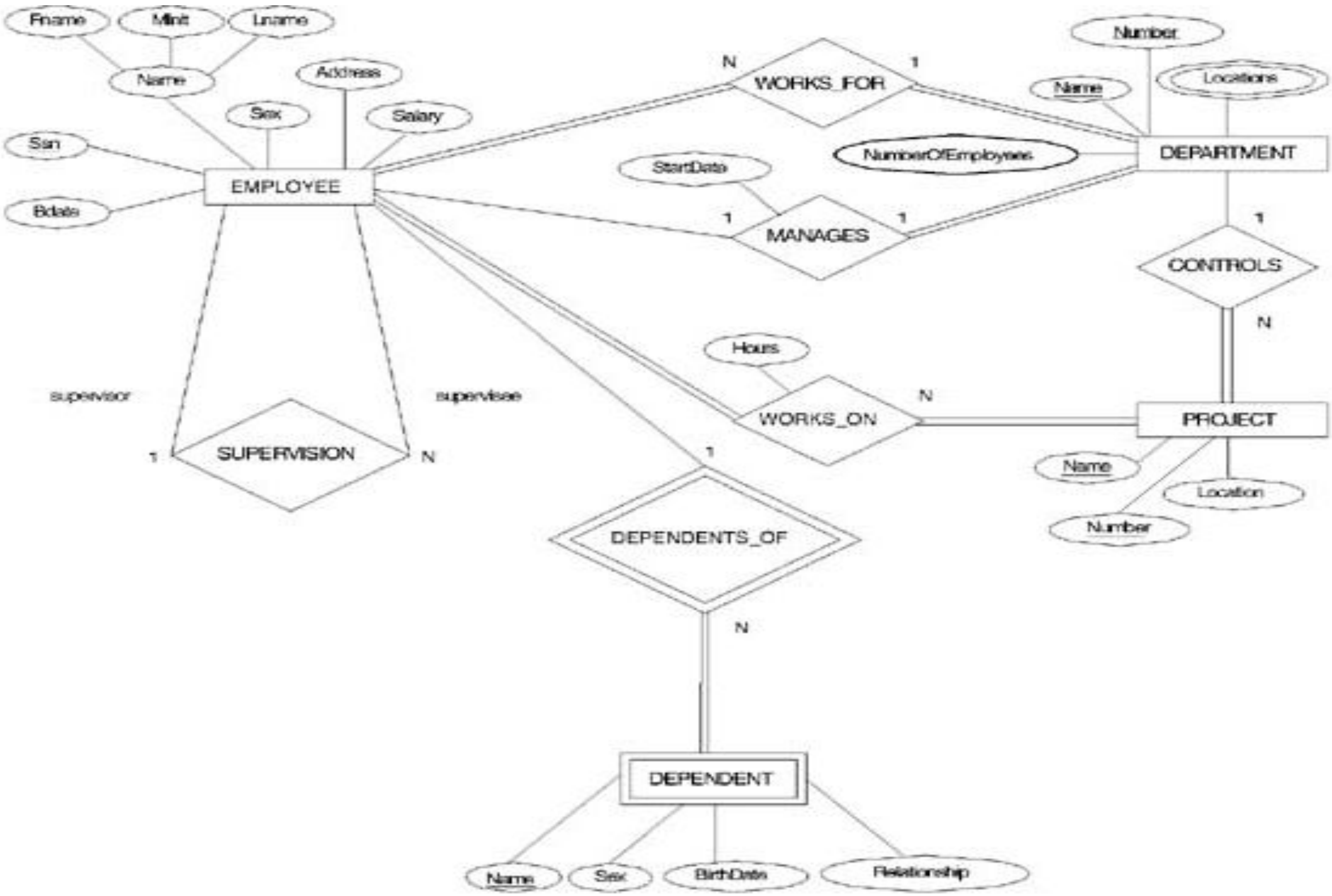
A RECURSIVE RELATIONSHIP SUPERVISION

In following figure, first role participation labeled with 1 and second role participation labeled with 2.



© The Benjamin/Cummings Publishing Company, Inc. 1994, Elmasri/Navathe, Fundamentals of Database Systems, Second Edition

Recursive Relationship Type is: SUPERVISION (participation role names are shown)

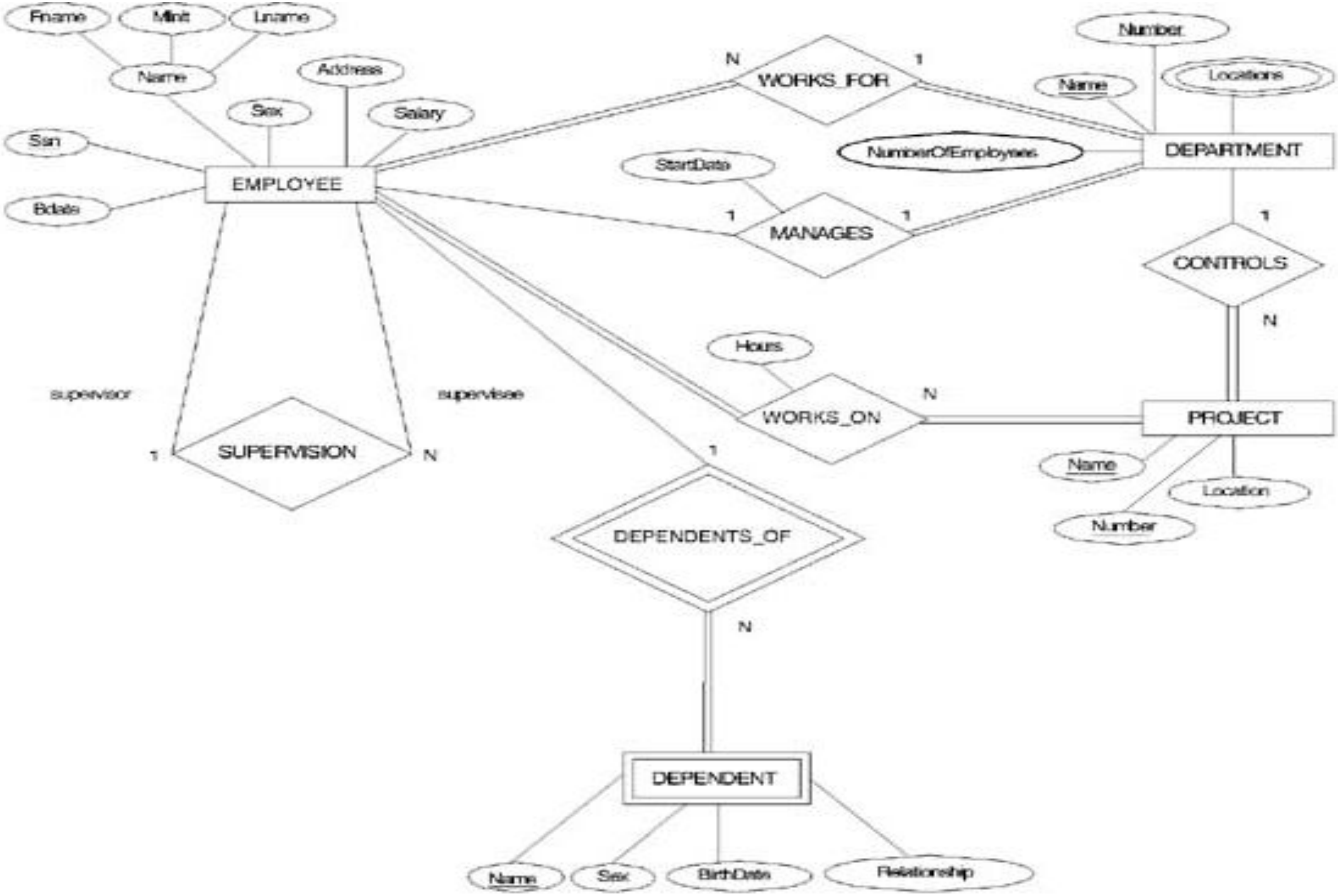


Attributes of Relationship types

A relationship type can have attributes;

for example, HoursPerWeek of WORKS_ON; its value for each relationship instance describes the number of hours per week that an EMPLOYEE works on a PROJECT.

Attribute of a Relationship Type is: Hours of WORKS_ON



Structural Constraints – one way to express semantics of relationships

<u>**Structural constraints on relationships:**</u>

- **Cardinality ratio** (of a binary relationship): 1:1, 1:N, N:1, or M:N

  SHOWN BY PLACING APPROPRIATE NUMBER ON THE LINK.

- **Participation constraint** (on each participating entity type): total (called *existence dependency*) or partial.

  SHOWN BY DOUBLE LINING THE LINK

NOTE: These are easy to specify <u>for Binary Relationship Types</u>.

Alternative (min, max) notation for relationship structural constraints:

- Specified on *each participation* of an entity type E in a relationship type R
- Specifies that each entity e in E participates in *at least* min and *at most* max relationship instances in R
- Default(no constraint): min=0, max=n
- Must have min≤max, min≥0, max ≥1
- Derived from the knowledge of mini-world constraints

Examples:

- A department has *exactly one* manager and an employee can manage *at most one* department.

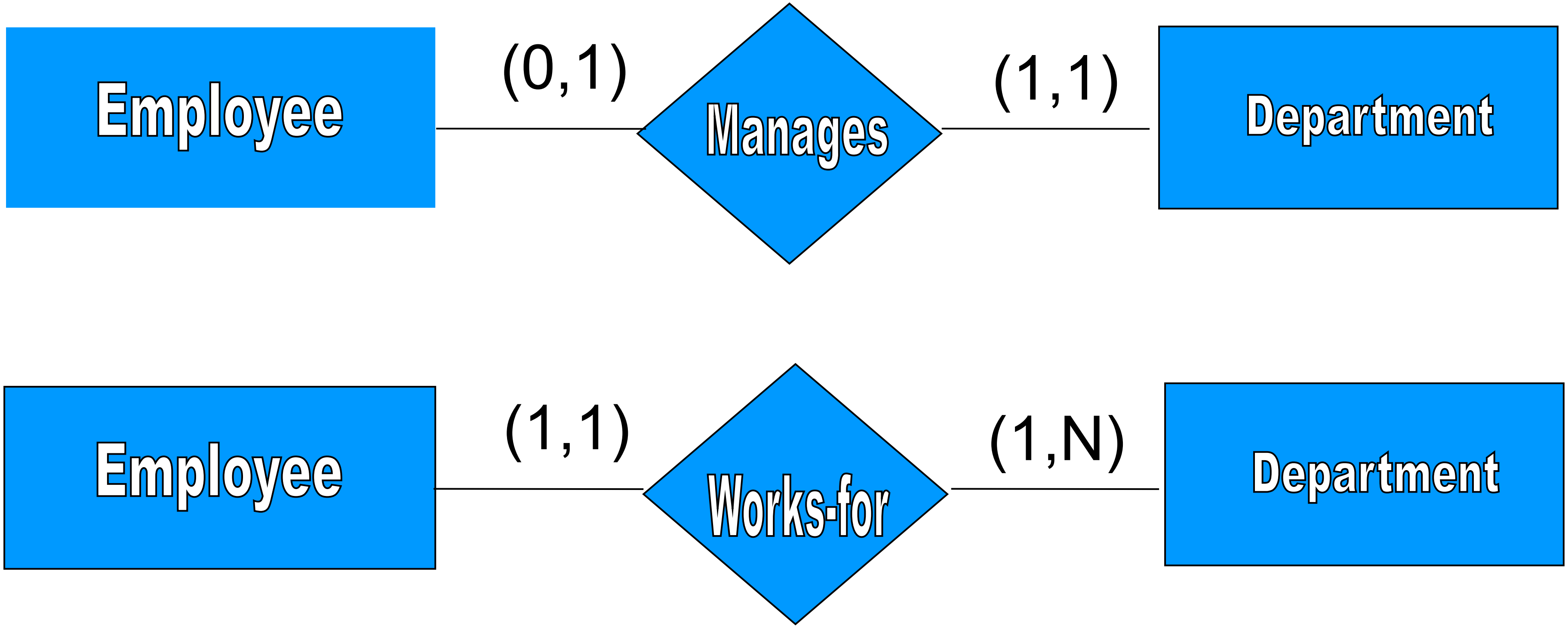
  Specify (0,1) for participation of EMPLOYEE in MANAGES

  Specify (1,1) for participation of DEPARTMENT in MANAGES
- An employee can work for *exactly one* department but a department can have *any number of employees*.

  Specify (1,1) for participation of EMPLOYEE in WORKS_FOR

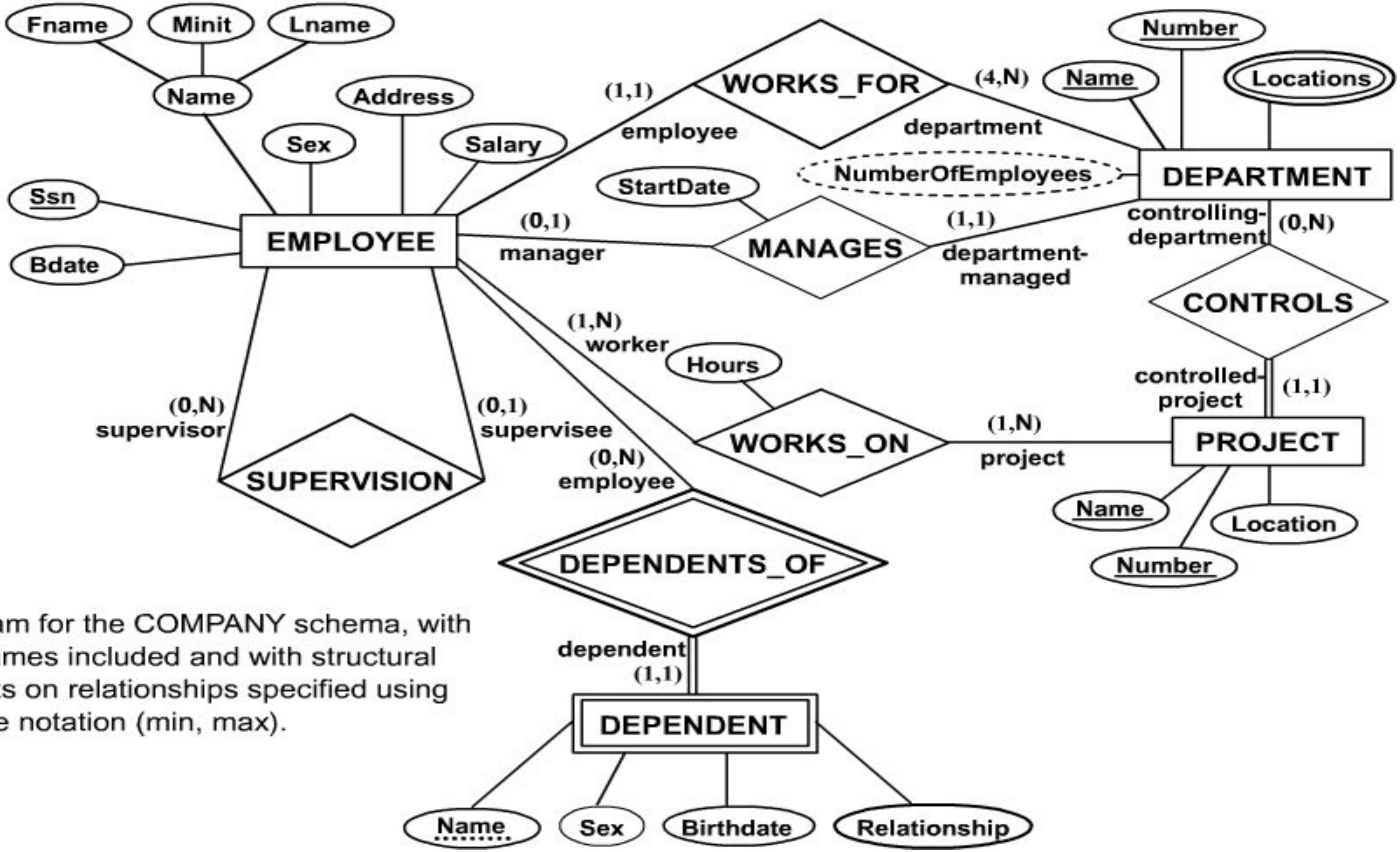
  Specify (1,n) for participation of DEPARTMENT in WORKS_FOR

The (min,max) notation relationship constraints



COMPANY ER Schema Diagram using (min, max) notation

Alternative ER Notations



ER diagram for the COMPANY schema, with all role names included and with structural constraints on relationships specified using alternative notation (min, max).

Relationships of Higher Degree

- Relationship types of degree 2 are called **binary**
- Relationship types of degree 3 are called **ternary** and of degree n are called **n-ary**
- In general, an n-ary relationship *is not* equivalent to n binary relationships
- Higher-order relationships discussed further in Chapter 4

Data Modeling Tools

A number of popular tools that cover conceptual modeling and mapping into relational schema design. Examples: ERWin, S- Designer (Enterprise Application Suite), ER- Studio, etc.

POSITIVES: serves as documentation of application requirements, easy user interface - mostly graphics editor support

Problems with Current Modeling Tools

DIAGRAMMING

Poor conceptual meaningful notation.

To avoid the problem of layout algorithms and aesthetics of diagrams, they prefer boxes and lines and do nothing more than represent (primary-foreign key) relationships among resulting tables.(a few exceptions)

METHODOLGY

lack of built-in methodology support.

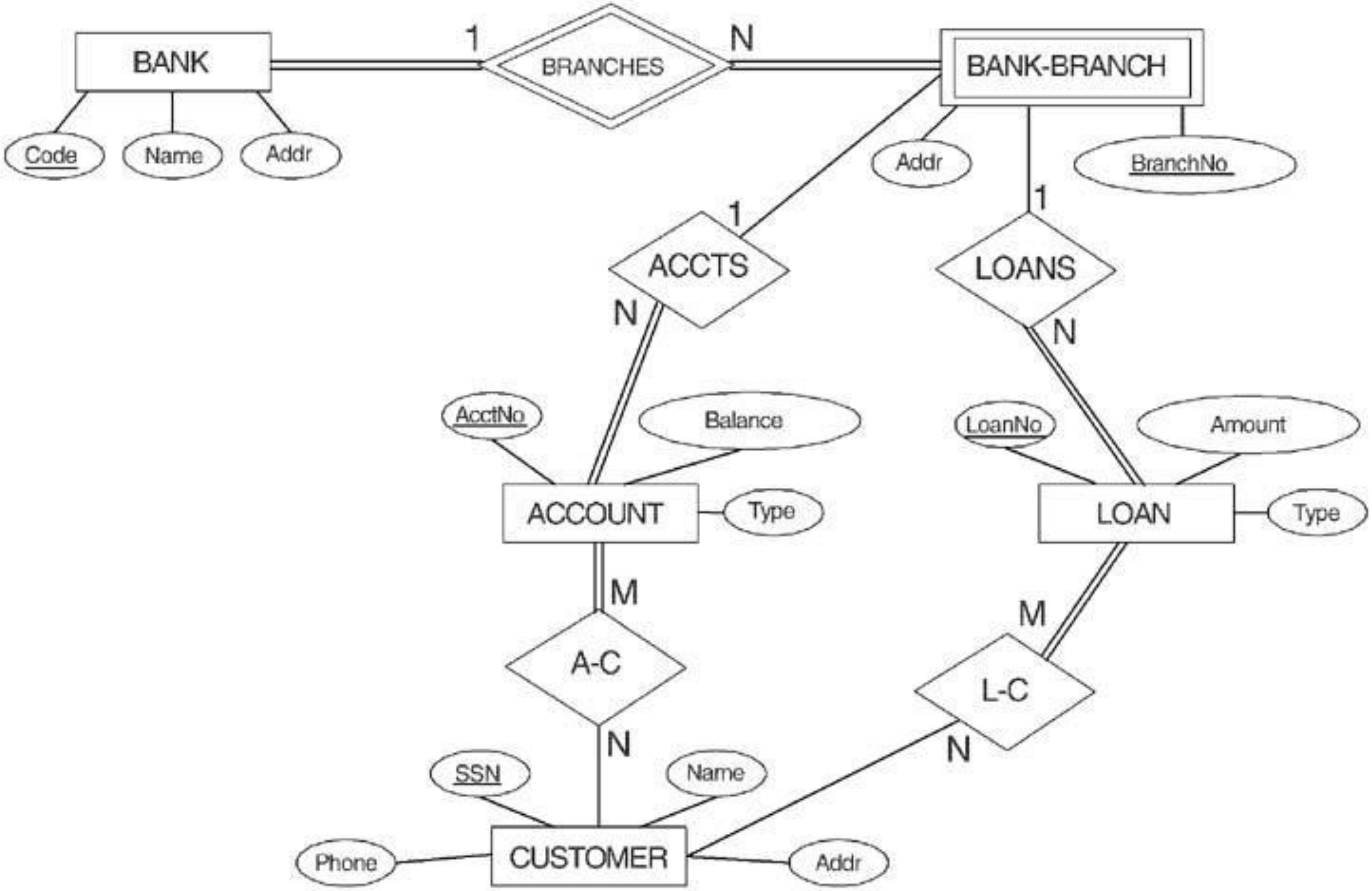
poor tradeoff analysis or user-driven design preferences.

poor design verification and suggestions for improvement.

Some of the Currently Available Automated Database Design Tools

COMPANY	TOOL	FUNCTIONALITY
Embarcadero Technologies	ER Studio	Database Modeling in ER and IDEF1X
	DB Artisan	Database administration and space and security management
Oracle	Developer 2000 and Designer 2000	Database modeling, application development
Popkin Software	System Architect 2001	Data modeling, object modeling, process modeling, structured analysis/design
Platinum Technology	Platinum Enterprice Modeling Suite: Erwin, BPWin, Paradigm Plus	Data, process, and business component modeling
Persistence Inc.	Pwertier	Mapping from O-O to relational model
Rational	Rational Rose	Modeling in UML and application generation in C++ and JAVA
Rogue Ware	RW Metro	Mapping from O-O to relational model
Resolution Ltd.	Xcase	Conceptual modeling up to code maintenance
Sybase	Enterprise Application Suite	Data modeling, business logic modeling

ER DIAGRAM FOR A BANK DATABASE



© The Benjamin/Cummings Publishing Company, Inc. 1994, Elmasri/Navathe, Fundamentals of Database Systems, Second Edition

PROBLEM with ER notation

THE ENTITY RELATIONSHIP MODEL IN ITS ORIGINAL FORM DID NOT SUPPORT THE SPECIALIZATION/ GENERALIZATION ABSTRACTIONS

Extended Entity-Relationship (EER) Model

Incorporates Set-subset relationships

Incorporates Specialization/Generalization Hierarchies

NEXT CHAPTER ILLUSTRATES HOW THE ER MODEL CAN BE EXTENDED WITH

- Set-subset relationships and Specialization/Generalization Hierarchies and how to display them in EER diagrams