

Chapter 3

The Relational Data Model and Relational Database Constraints

Department of Health Informatics

Sep 2022

Chapter Objectives

By the end of this chapter, you will be able to:

- Identify relational database model concepts
- Identify relational database constraints and schemas
- Create conceptual schema design using entity-relation (ER) model
- Create logical schema design using commercial DBMS
- Update Operations, Transactions, and
- Deal with Constraint Violations

Presentation Outline

- Introduction
- Relational Model Concepts
- Relational Model Constraints and Relational Database Schemas
- Update Operations, Transactions, and Dealing with Constraint Violations

Introduction

- In this chapter, we principally see the modeling concepts of the **entity–relationship (ER) model**
- The ER model and its variations are frequently used for the **conceptual schema design** of database applications
- We also describe the basic data-structuring **concepts** and **constraints** of the ER model and discuss their use in the design of conceptual schemas for database applications.
- The diagrammatic notation associated with the ER model, called **ER diagrams** are presented

Relational Model Concepts

- A relational database model represents the database as a collection of relations.
- A relation is a **table** of values.
- Every row in the table represents a collection of related data values.
- These rows in the table denote a real-world **entity** or **relationship**.

Relational Model Concepts....

- **Attribute**: each column in a table.
- **Tables**: relations are saved in the table format. A table has two properties rows and columns.
- **Tuple** a single row of a table, which contains a single record.
- **Relation** schema: represents the name of the relation with its attributes.
- **Degree**: the total number of attributes
- **Cardinality**: total number of rows
- **Column**: the set of values for a specific attribute.
- **Attribute domain** – the set of values allowed in an attribute

Steps in designing a relational database models:

Specify the data and functional requirements:

- Understand and document the **data requirements** the database users
- Produce a written set of **users' requirements**.
- In parallel, specify the **functional requirements**.
- The **functional requirements** include the user-defined operations (or transactions

Steps in designing a relational database...

Create conceptual schema design:

- Once the requirements have been collected, create a **conceptual schema using high-level data model.**
- The conceptual schema design is a concise description of the data requirements including detailed descriptions of the **entity types, relationships, and constraints.**
- Because these concepts do not include implementation details, they are usually easier to communicate with non-technical.
- The **ER model** could be used to create conceptual schema design

Steps in designing a relational database...

Create logical schema design or data model mapping (implementation):

- Transform the high-level data model (conceptual design) into an **implementational data model** using commercial DBMS, such as relational (SQL) DBMS. The result of this step is creation of a **relational database model schema**.

Steps in designing a relational database...

Specify the physical design (physical schema)

- Specify the internal storage structures, file organizations, indexes, access paths, and physical design parameters for the database files.

The ER model to represent conceptual schema designs

- An ER **model** describes the **conceptual schema** design with the help of a diagram, called **ER diagram**.
- An ER model is a design or blueprint of a database that can later be implemented as a logical schema design
- The main components of ER model are **entity attributes and relationships**.
- See an ER diagram for a hypothetical COMPANY database.

[CHAPTER 3 FIGURE 1 ER Diagram.docx](#)

Elements of ER Models

- Entity and their various types of attributes
- Entity types and entity sets
- Key attributes
- Value Sets (Domains) of Attributes
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- Participation Constraints and Existence Dependencies (Total Participation)

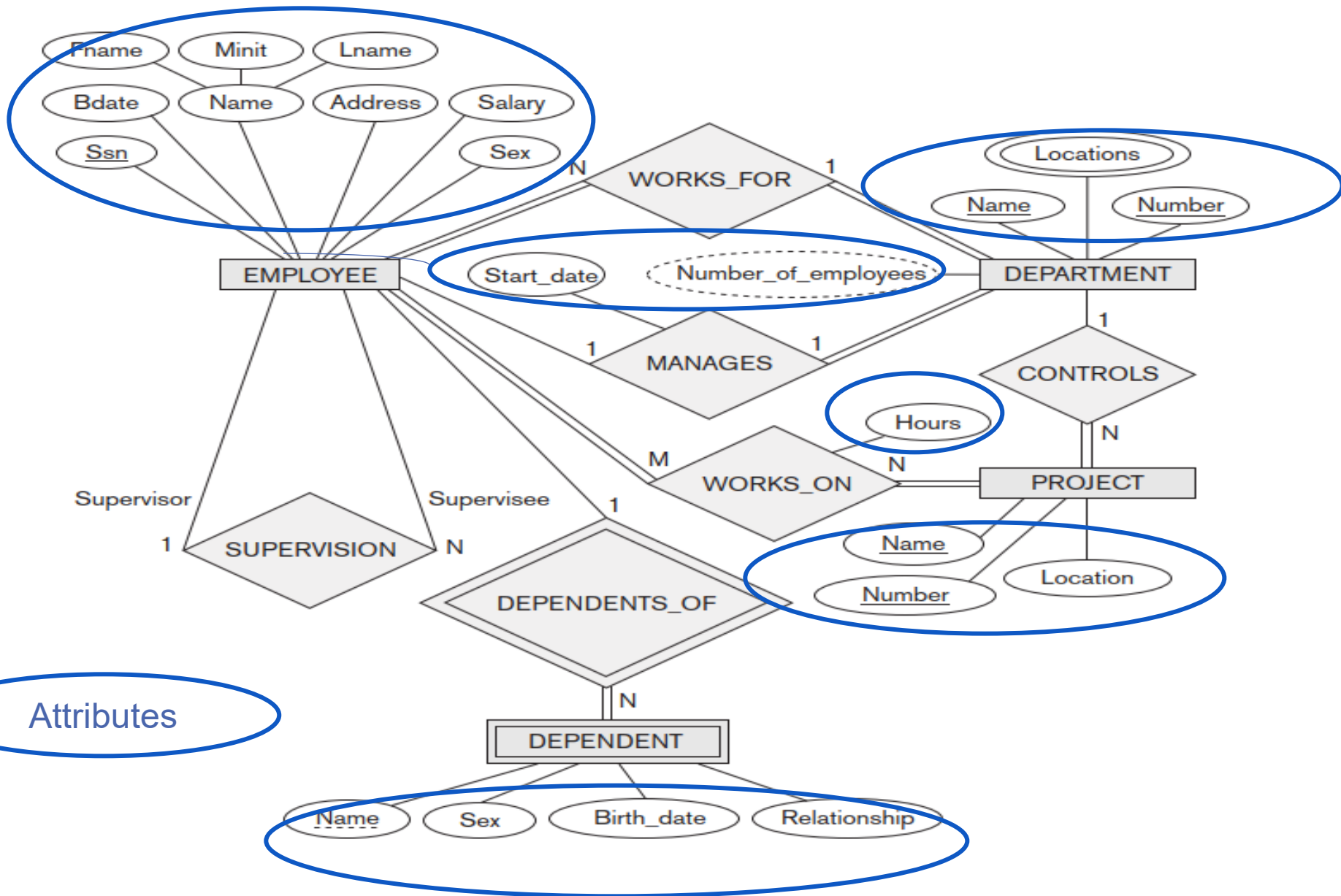
Elements of ER Models

- **Entity and their various types of attributes**
- Entity types and entity sets
- Key attributes
- Value Sets (Domains) of Attributes
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- Participation Constraints and Existence Dependencies (Total Participation)

The concepts of the ER diagram

Entities and Their Attributes:

- The basic concept that the ER model represents is an **entity**.
- An **entity** may be an object with a **physical existence** (for example, an employee) or an **object with a conceptual existence** (for instance, **a company, a course**).
- Each entity has **attributes**— properties that describe it.
- The attributes that describe each entity become a major part of the data stored in the database.
- Entities have **values** for each attributes which become rows or tuples after implementation of the logical schema.
- Entity is e
- For example, values for *Fname* attribute of **EMPLOYEE** entity could be '*Kassa*', '*Kefelegn*', etc.

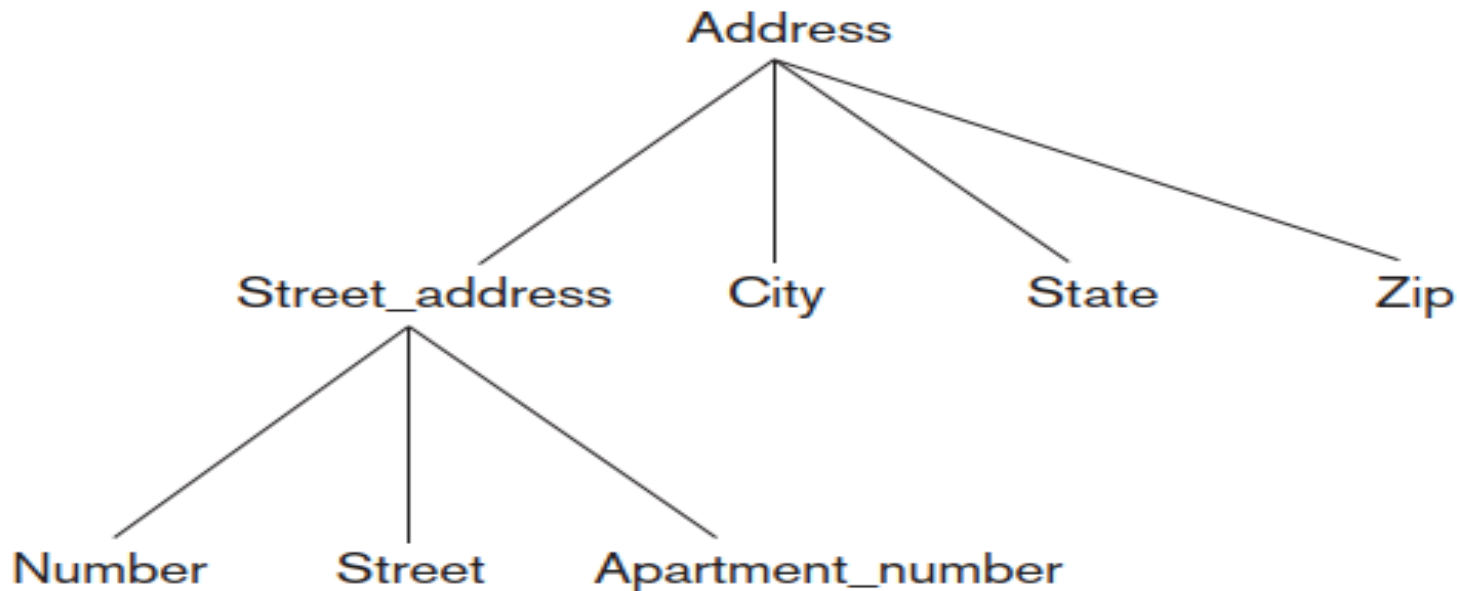


Attributes

The concepts of the ER diagram...

Attributes in the ER model could be simple versus composite, single-valued versus multivalued, and stored versus derived attributes.

Composite versus Simple (Atomic) Attributes.



The concepts of the ER diagram...

Single-Valued vs. Multivalued Attributes:

- Multivalued attribute is an attribute that can hold multiple values.
- It is represented with **double ovals** in an ER Diagram.
- For example: phone numbers
- **Double Ellipses**: Multivalued Attributes

Stored vs. Derived Attributes.

- e.g. age can be derived from two stored attributed (Birth_date attributes of a person and current (today's) date.
- Represented by **dashed ellipses**

The concepts of the ER diagram...

NULL Values:

- When the value in a column is **unknown** or **missing**.
- A null is neither an **empty string** (for character or datetime data types) **nor a zero value** (for numeric data types).
- An attribute assumes NULL if **not known (or missing)** or **not applicable**.

Complex Attribute:

- Formed by **nesting composite attributes** and **multi-valued attributes in arbitrary way**.
- For example, a person can have many phone numbers, e-mail, home addresses etc.
- An attribute of 'Contact detail' can nest all the above

Elements of ER Models

- Entity and their various types of attributes
- **Entity types and entity sets**
- Key attributes
- Value Sets (Domains) of Attributes
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- Participation Constraints and Existence Dependencies (Total Participation)

Entity Types and Entity Sets

- The **entity type** is a collection (or set) of entities having **similar attributes**.
- Each entity type in the database is described by its **name and attributes**.
- Figure 2 shows two entity types: **EMPLOYEE** and **COMPANY** and a list of some of the attributes for each.
- The collection of all entities of a particular entity type in the database at any point in time is called an **entity set** or **entity collection**.
- Entity types and attributes are represented in ER diagram with rectangular geometric shapes

Entity Type Name:

EMPLOYEE

COMPANY

Name, Age, Salary

Name, Headquarters, President

**Entity Set:
(Extension)**

e_1 •

(John Smith, 55, 80k)

e_2 •

(Fred Brown, 40, 30K)

e_3 •

(Judy Clark, 25, 20K)

•
•
•

c_1 •

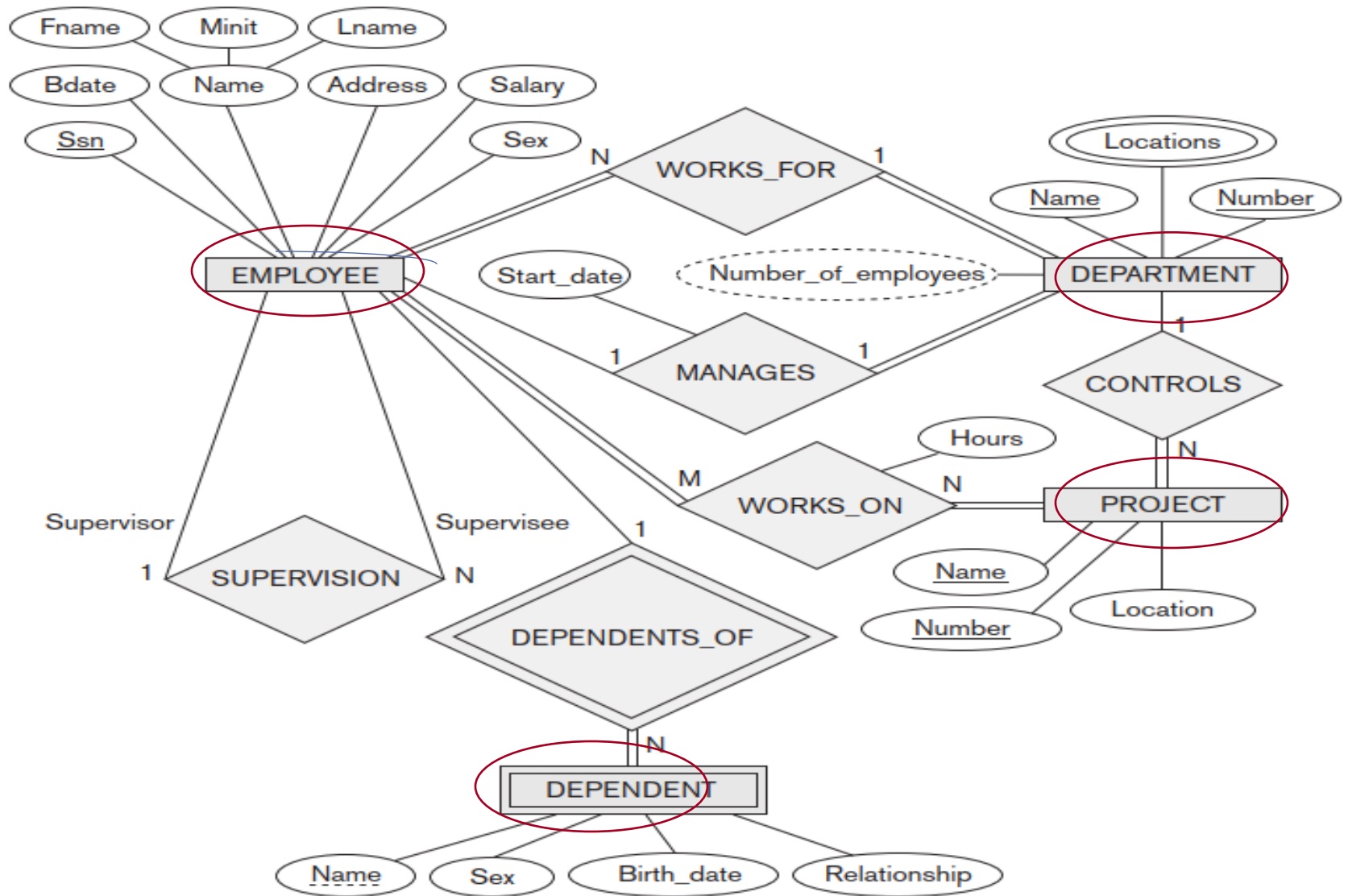
(Sunco Oil, Houston, John Smith)

c_2 •

(Fast Computer, Dallas, Bob King)

•
•
•

Figure 2: EMPLOYEE and COMPANY entity types and various entities in each entity type



Elements of ER Models

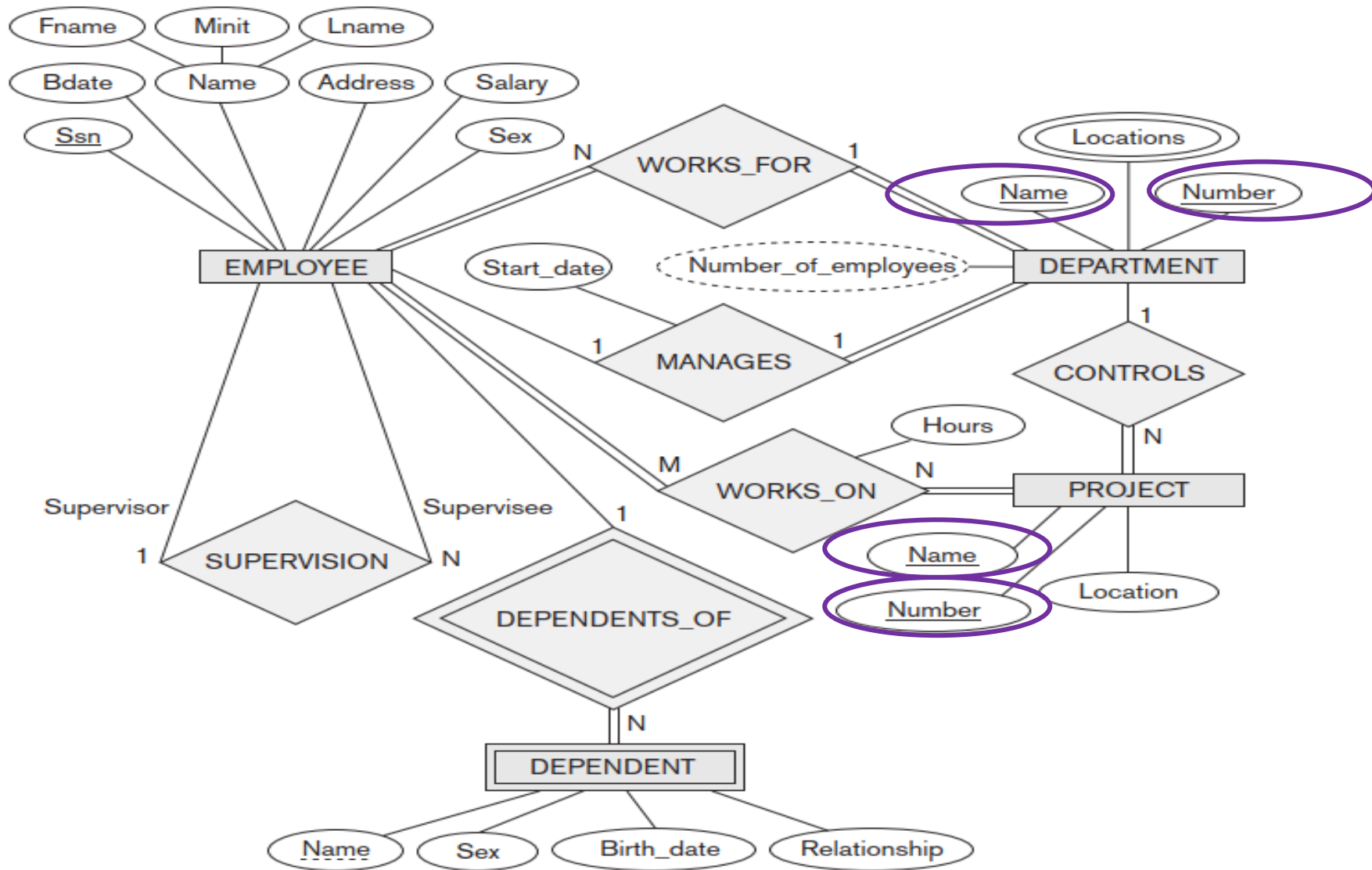
- Entity and their various types of attributes
- Entity types and entity sets
- **Key attributes**
- Value Sets (Domains) of Attributes
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- Participation Constraints and Existence Dependencies (Total Participation)

Key Attributes of an Entity Type

- Key attributes are attributes whose values are **unique for each individual entity in the entity set**.
- The values of such attributes can be used to **identify each entity uniquely**.
- Example: the **Name** attribute is a key of the COMPANY entity type in Figure 2
- Sometimes several attributes together form a key, meaning that the combination of the attribute values must be distinct for each entity (**composite keys**)

Key Attributes of an Entity Type...

- In ER diagrammatic notation, each key attribute has its **name underlined inside the oval**
- A key attribute is a constraint that prohibits **any two entities from having the same value for the key attribute at the same time.**
- An entity type may also have **no key**, in which case it is called a **weak entity type.**
- Notice that in ER diagrams, there may be more than one key attributes because there is **no a concept of primary key in ER.**



Elements of ER Models

- Entity and their various types of attributes
- Entity types and entity sets
- Key attributes
- **Value Sets (Domains) of Attributes**
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- Participation Constraints and Existence Dependencies (Total Participation)

Value Sets (Domains) of Attributes

- **Rules** that describe the available values of a field type.
- They are used to constrain the values **allowed in any particular attribute**
- They provide a method for **enforcing data integrity** by limiting what can be placed on a field to a valid list or range of choices.
- If the range of ages allowed for employees is between 16 and 70, we can specify the value set of the Age attribute of EMPLOYEE to be the set of integer numbers between 16 and 70.
- **Value sets are not typically displayed in basic ER**

Elements of ER Models

- Entity and their various types of attributes
- Entity types and entity sets
- Key attributes
- Value Sets (Domains) of Attributes
- **Relationship Types and Structural Constraints**
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- Participation Constraints and Existence Dependencies (Total Participation)

Relationship Types and Structural Constraints

- A relationship type among entity types represents the associations or relationship sets among entities from these entity types.
- Each of the entity types $E_1, E_2, \dots, E_n$ is said to participate in the relationship type
- In ER diagrams, relationship types are displayed as diamond-shaped boxes, which are connected by straight lines to the rectangular boxes representing the participating entity types.

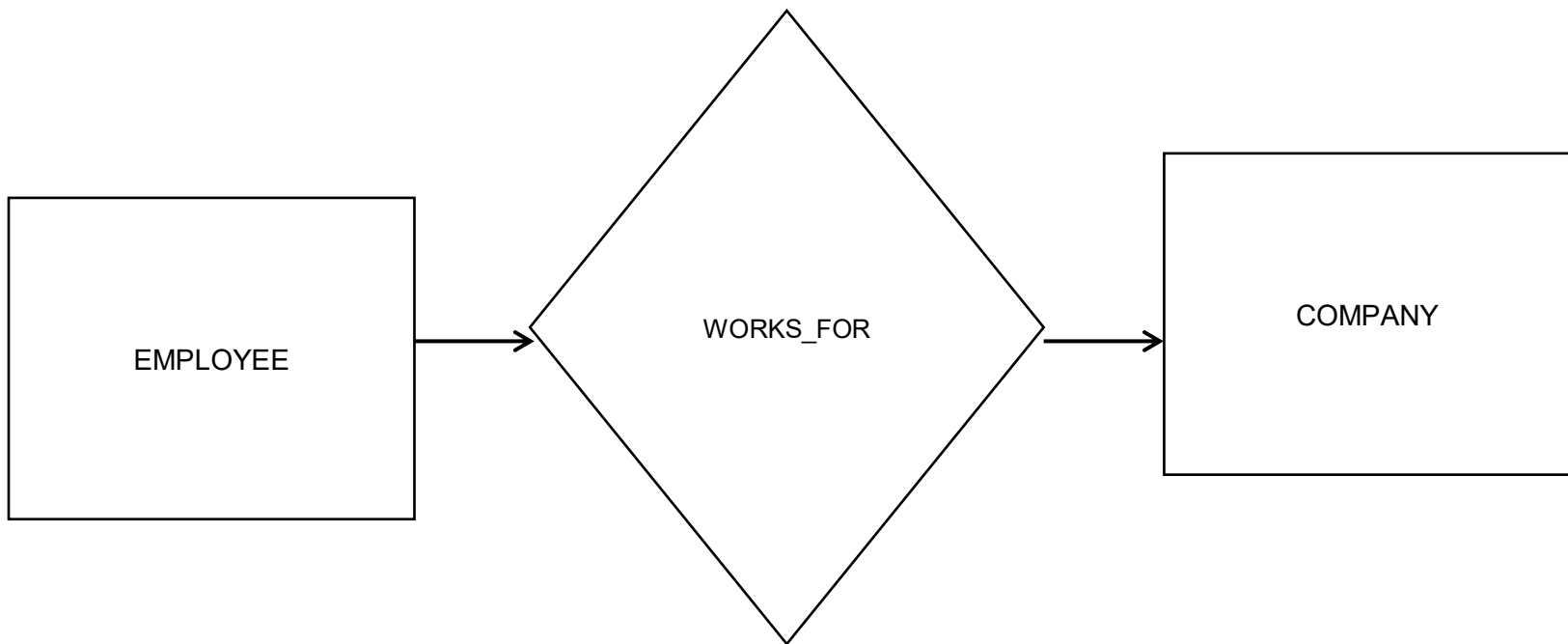
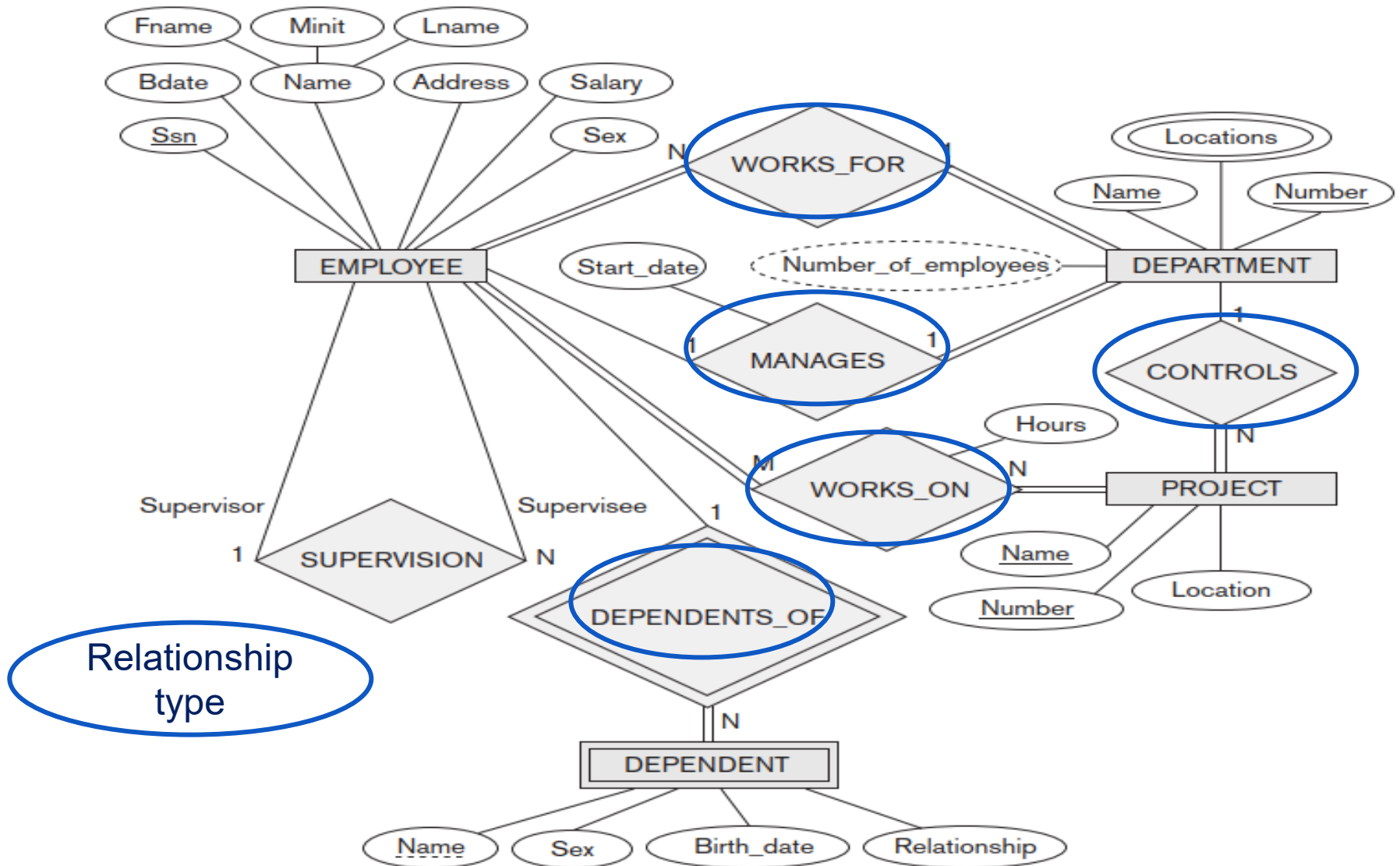


Figure: `WORKS_FOR` relationship type connecting `EMPLOYEE` and `COMPANY` entity types



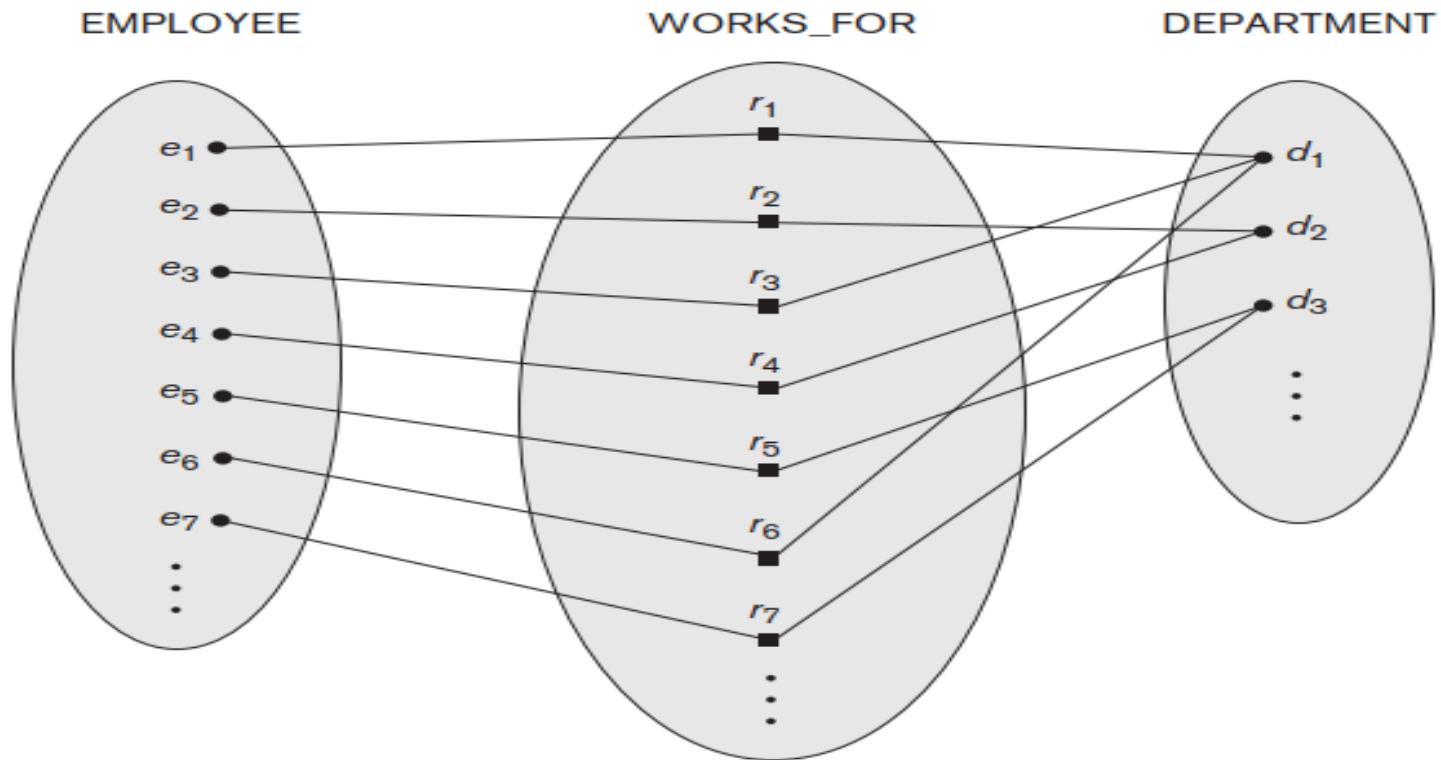


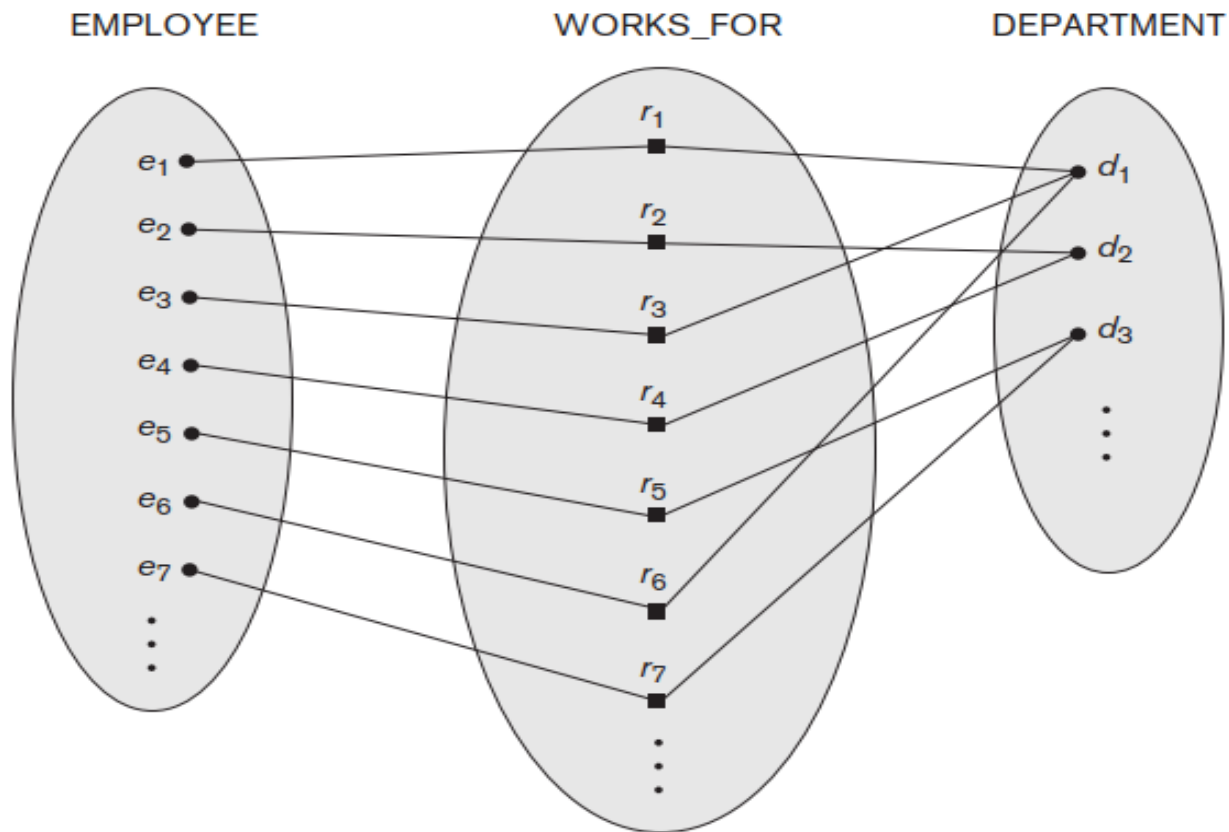
Figure: Some instances in the **WORKS_FOR** relationship type, which represents a relationship type **WORKS_FOR** between **EMPLOYEE** and **DEPARTMENT**.

Relationship Types and Structural Constraints...

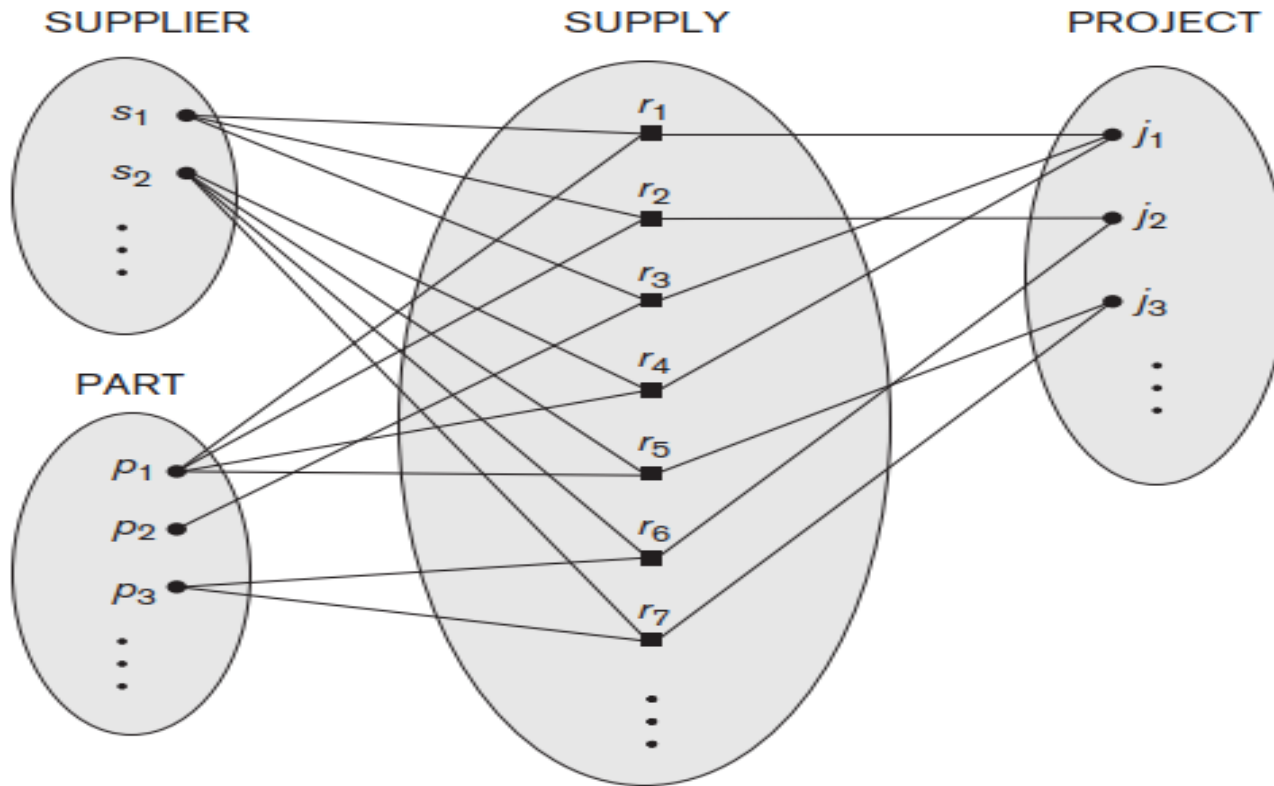
- The above figure shows that each relationship instance is shown connected to the EMPLOYEE and DEPARTMENT entities that participate in the relationship.
- In the representation, the employees e_1 , e_3 , and e_6 work for department d_1 ; the employees e_2 and e_4 work for department d_2 ; and the employees e_5 and e_7 work for department d_3 .

Degree of a Relationship Type

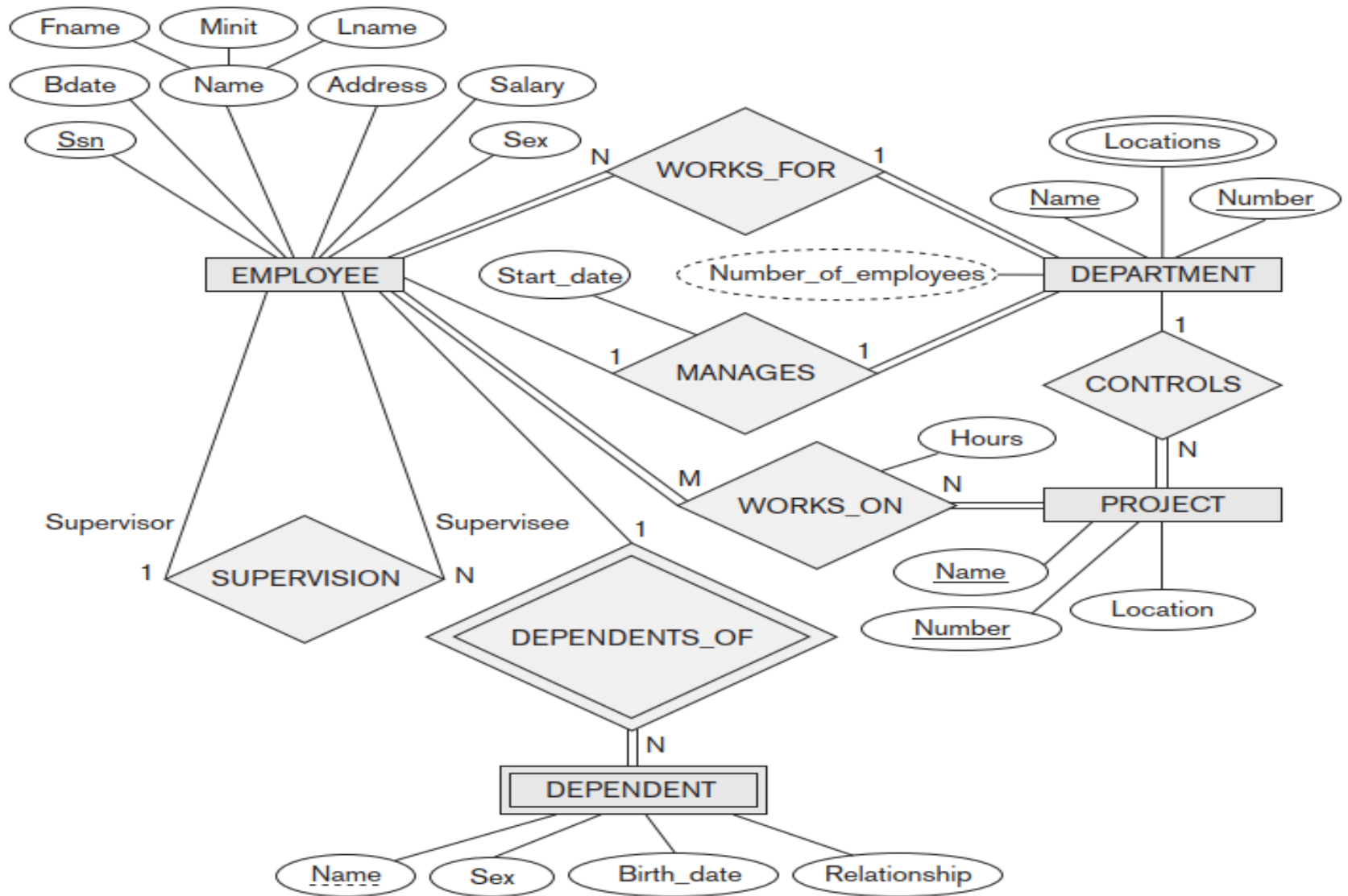
- The degree of a relationship type is the **number of participating entity types**.
- The degree of a relationship types could be
degree one (unary),
two (binary) degree
three (ternary).
- There could be higher degree relationship types.

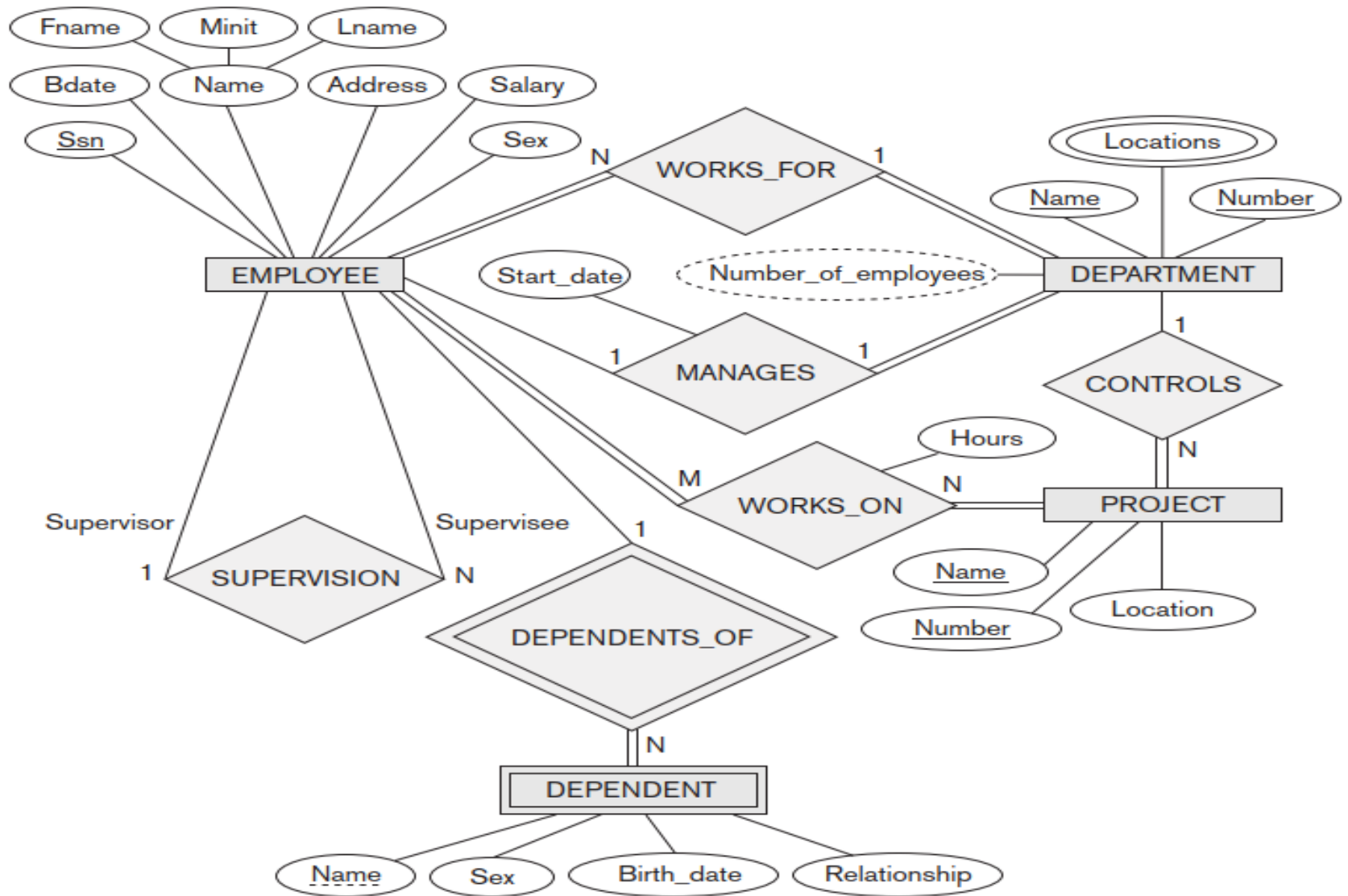


WORKS_FOR relationship is of degree two (binary)



SUPPLY relationship is of degree three (ternary)



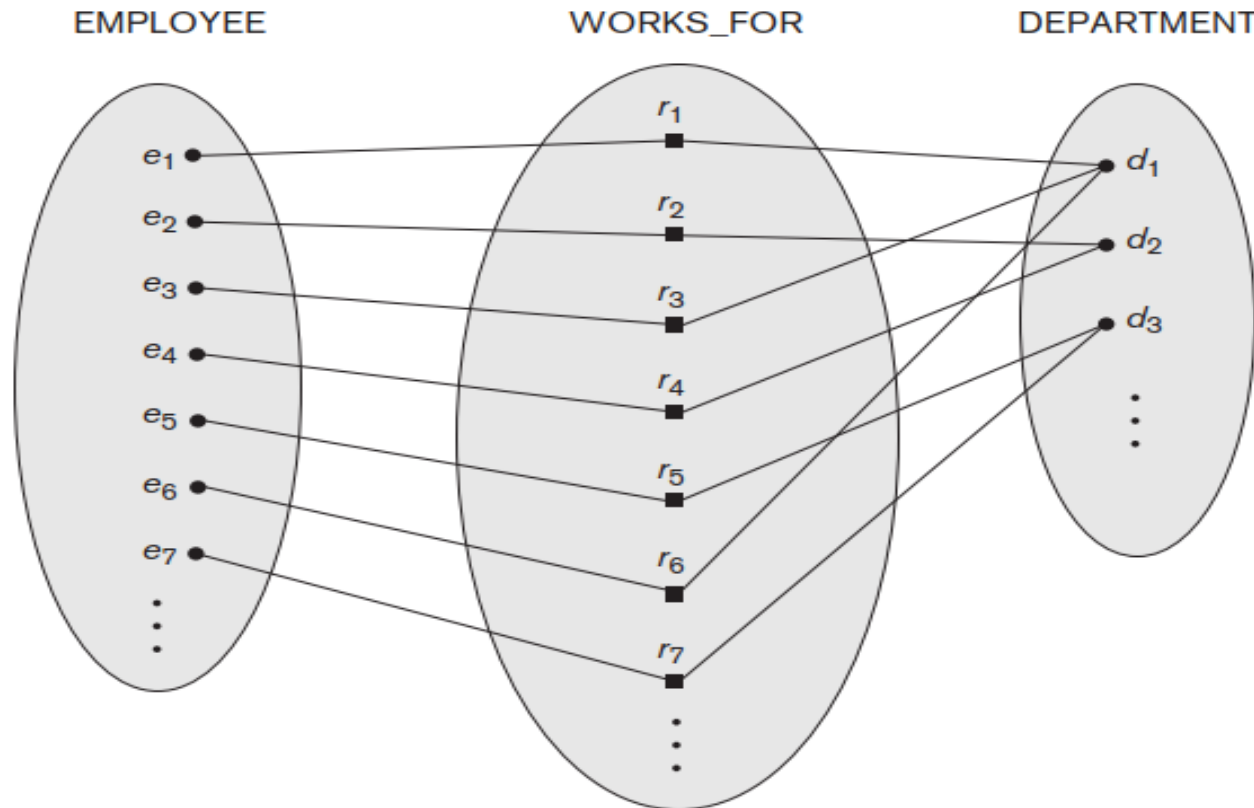


Elements of ER Models

- Entity and their various types of attributes
- Entity types and entity sets
- Key attributes
- Value Sets (Domains) of Attributes
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- **Cardinality ratio and participation constraints**
- Participation Constraints and Existence Dependencies (Total Participation)

Constraints on Binary Relationship Types

- Relationship types have constraints that limit the possible combinations of entities that may participate in the corresponding relationship set.
- These constraints are determined from the real world situation that the relationships represent.
- For example, in figure below, if the company has a rule that **each employee must work for exactly one department**, then we would like to describe this constraint in the schema.
- We can distinguish two main types of binary relationship constraints: **cardinality ratio** and **participation constraints**.



Each employee working only for one department, but the department can have more than one employee (**many-to-one relationship of EMPLOYEE:DEPARTMENT**)

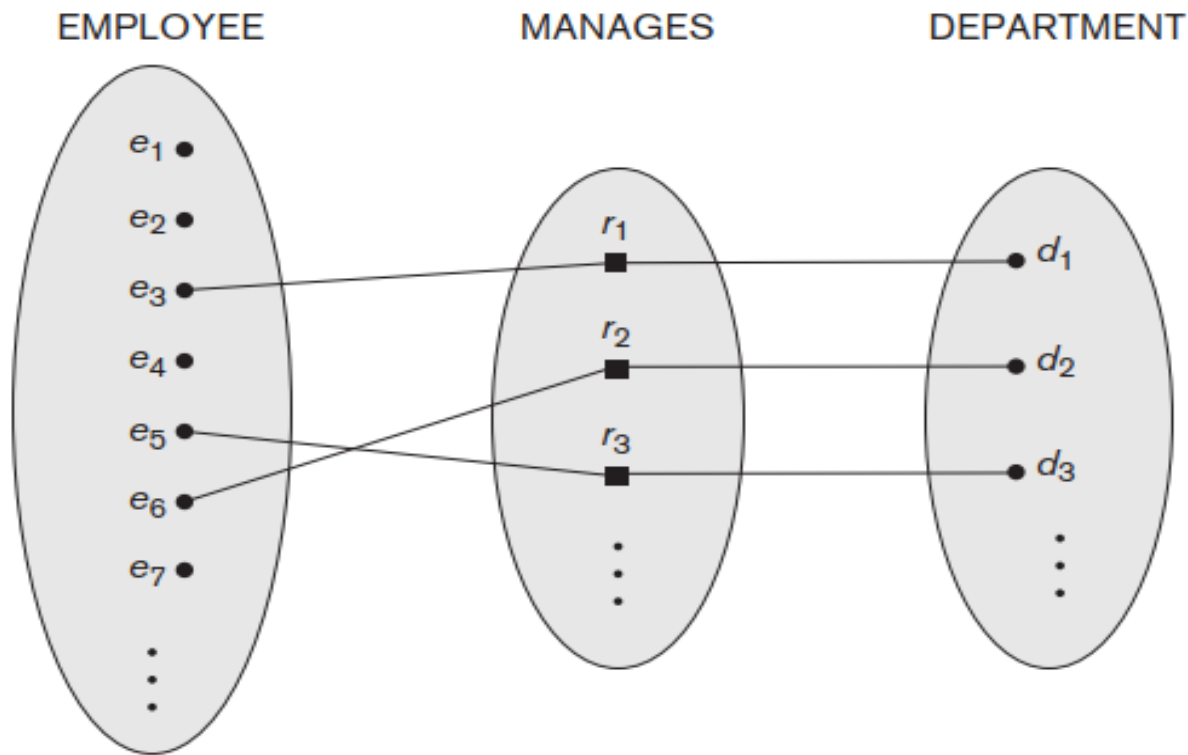
Cardinality Ratios for Binary Relationships.

- This specifies the maximum number of relationship instances that an entity can participate in.
- The possible cardinality ratios for binary relationship types are
 - 1:1,
 - 1: M,
 - M: 1, and
 - N: M.

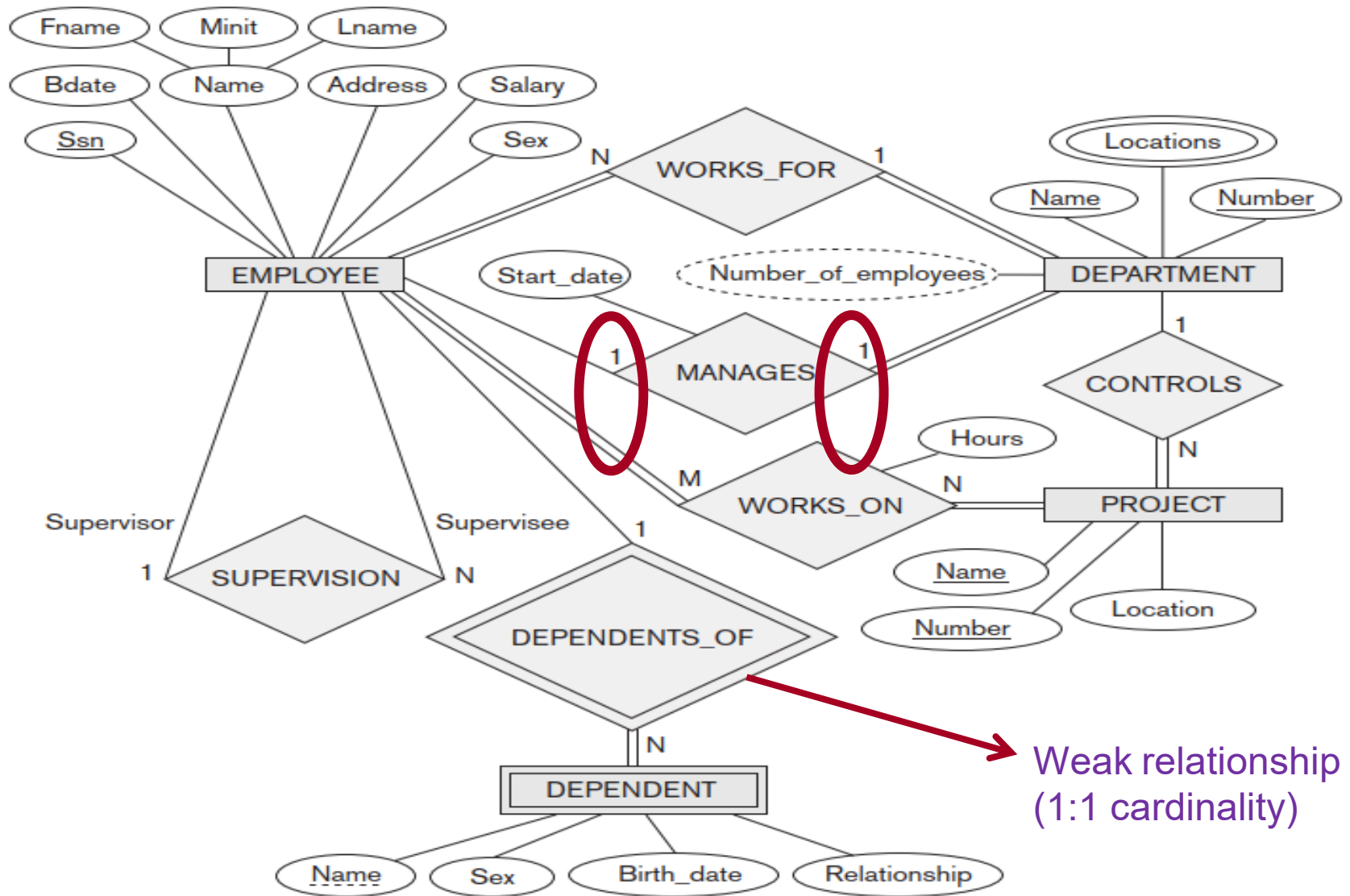
Cardinality Ratios for Binary Relationships....

One to One (1:1) cardinality ratio of relationships:

- When one entity in each entity set takes part **at most once in the relationship**.
- Such entity types do not have any key attribute of their own are called **weak entity type**.
- Weak entity type is represented in ER diagram by **double rectangle** as (DEPENDENTS_OF relationship type).



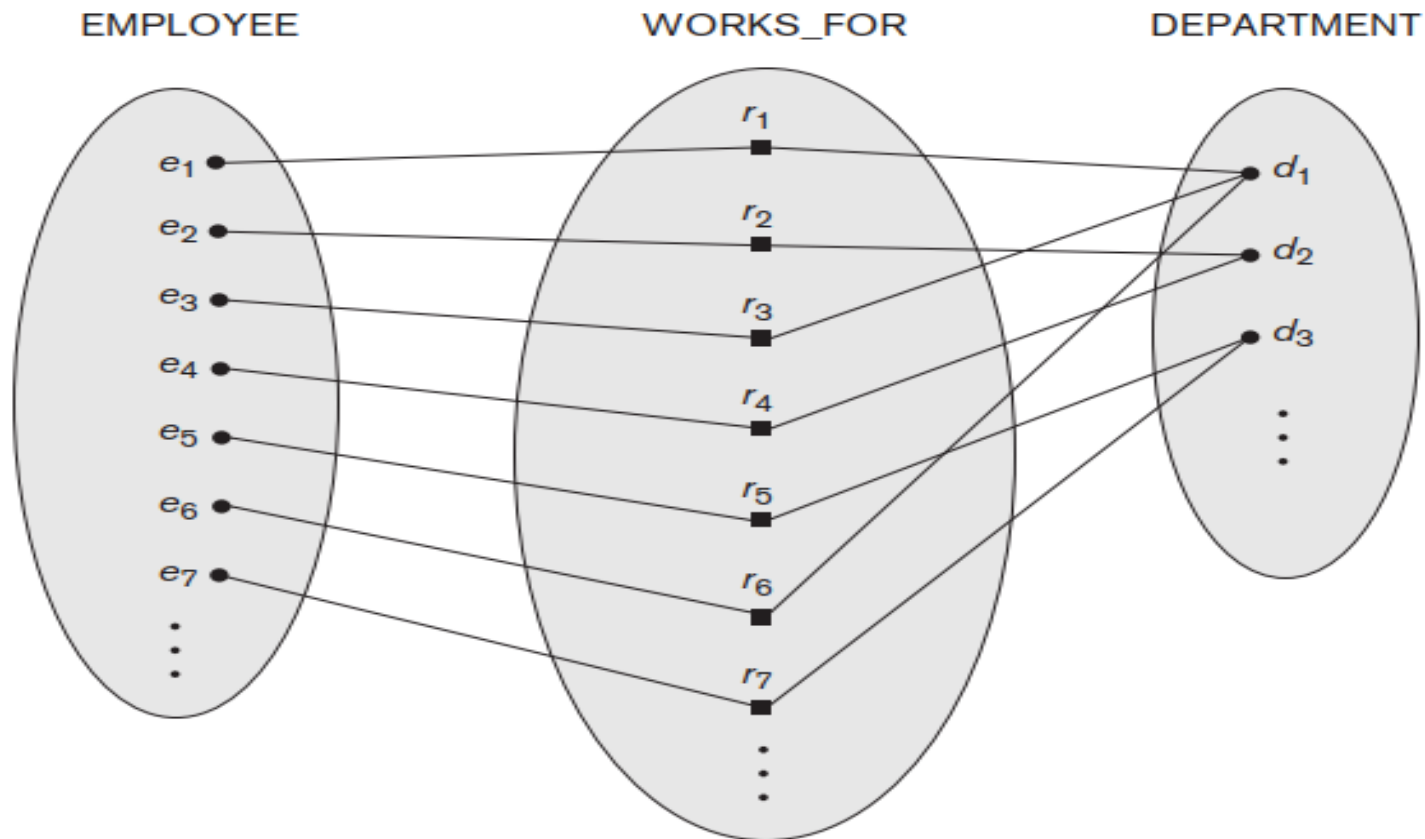
1:1 cardinality ratio of relationships. Each entity of entity type EMPLOYEE associates with only one entity in entity type DEPARTMENT and vice-versa (MANAGES is weak entity)



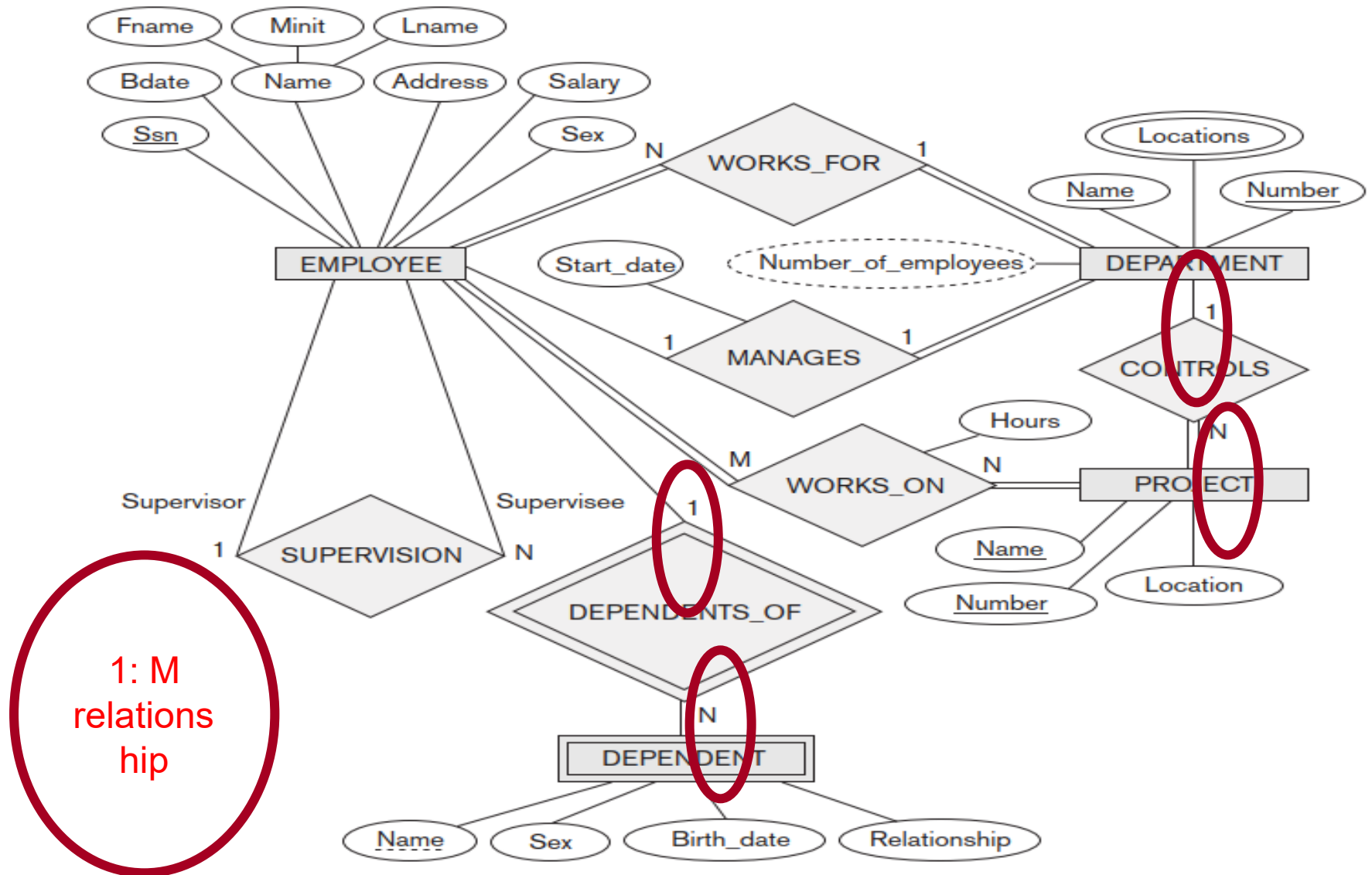
Cardinality Ratios for Binary Relationships...

One to Many (1:M) cardinality ratio of relationships:

- If entities in the first entity set take part in the relationship **set at most once** and entities in the second entity set take part many times (at least twice), the cardinality is said to be one-to-many.
- Example: in the WORKS FOR binary relationship type DEPARTMENT:EMPLOYEE is of cardinality ratio 1:M, meaning that each department can be related to (that is, employs) any number of employees (N), but an employee can be related to (work for) at most one department (1).



1:M cardinality ratio of relationships. Each entity of entity type EMPLOYEE associates with only one entity in entity type DEPARTMENT but each entity in entity type DEPARTMENT can be related to more than one entities in entity type EMPLOYEE



1: M
relations
hip

Cardinality Ratios for Binary Relationships....

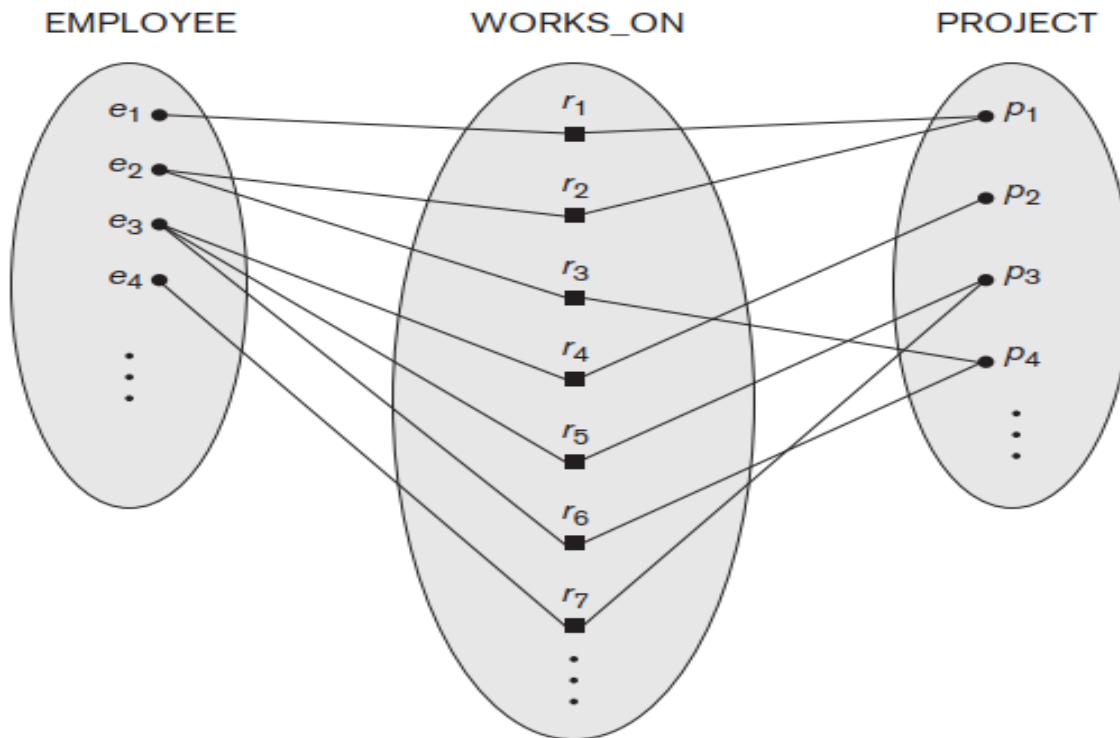
Many to One (N:1) cardinality ratio of relationships:

- If entities in the first entity set take part in the relationship set many times (at least twice), while entities in the second entity set take part at most once, the cardinality is said to be many-to-one.

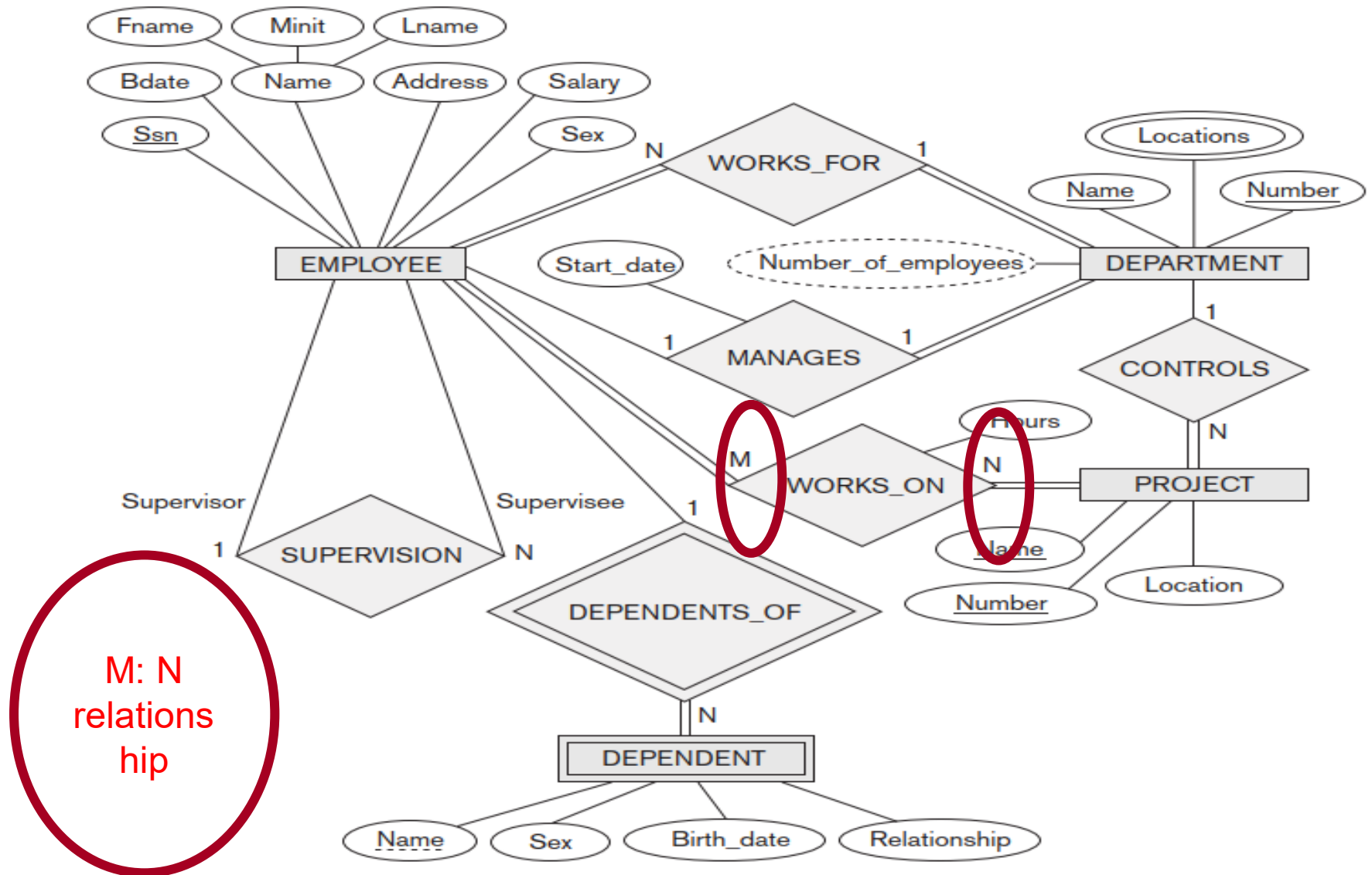
Cardinality Ratios for Binary Relationships...

Many to many (N:M) cardinality ratio of relationships:

- if entities in both the entity sets take part many times (at least twice) in the relationship set.
Example: the relationship type WORKS_ON is of cardinality ratio M:N, because in the real world, an employee can work on several projects and a project can have several employees.



N:M cardinality ratio of relationships: WORKS_ON

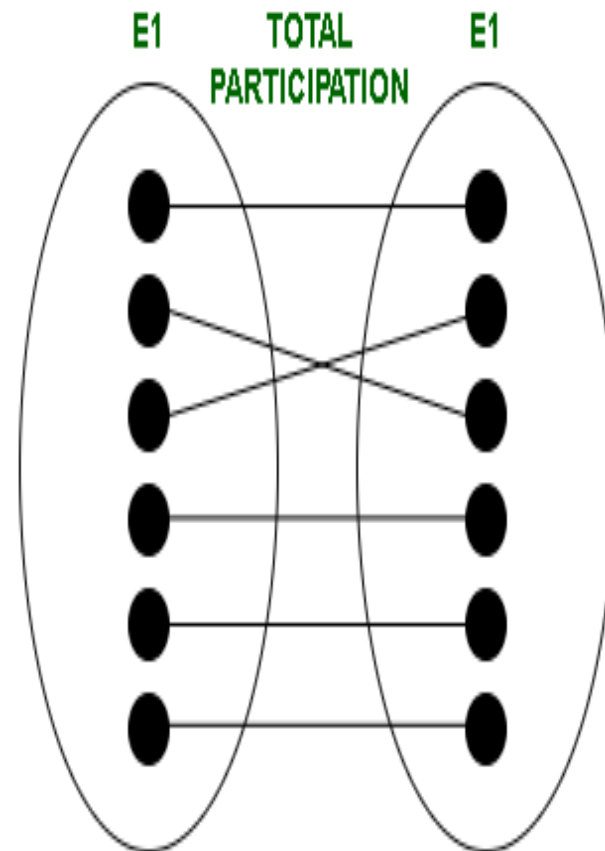
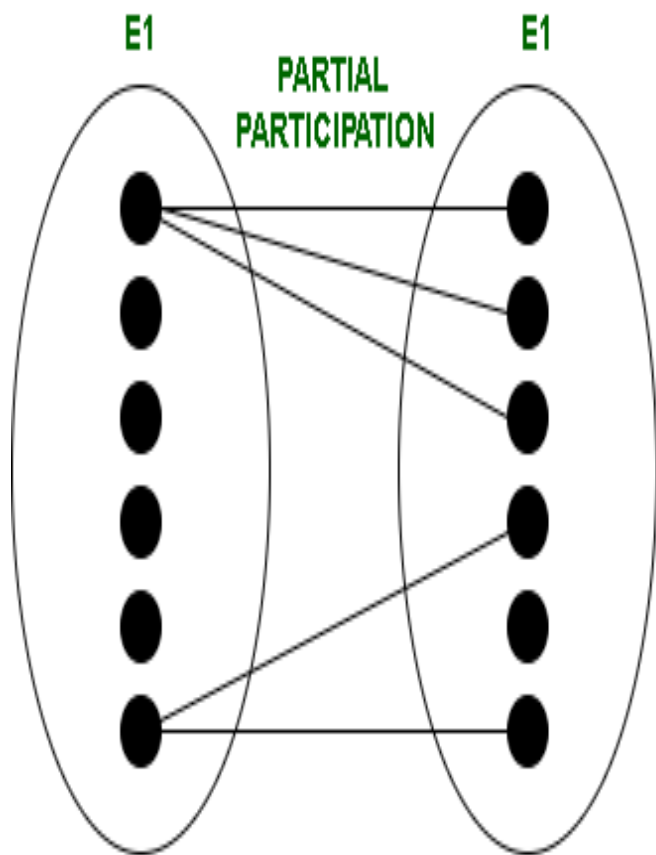


Elements of ER Models

- Entity and their various types of attributes
- Entity types and entity sets
- Key attributes
- Value Sets (Domains) of Attributes
- Relationship Types and Structural Constraints
- Weak attributes vs. strong attributes
- Relationship
- Cardinality ratio and participation constraints
- **Participation Constraints and Existence Dependencies (Total Participation)**

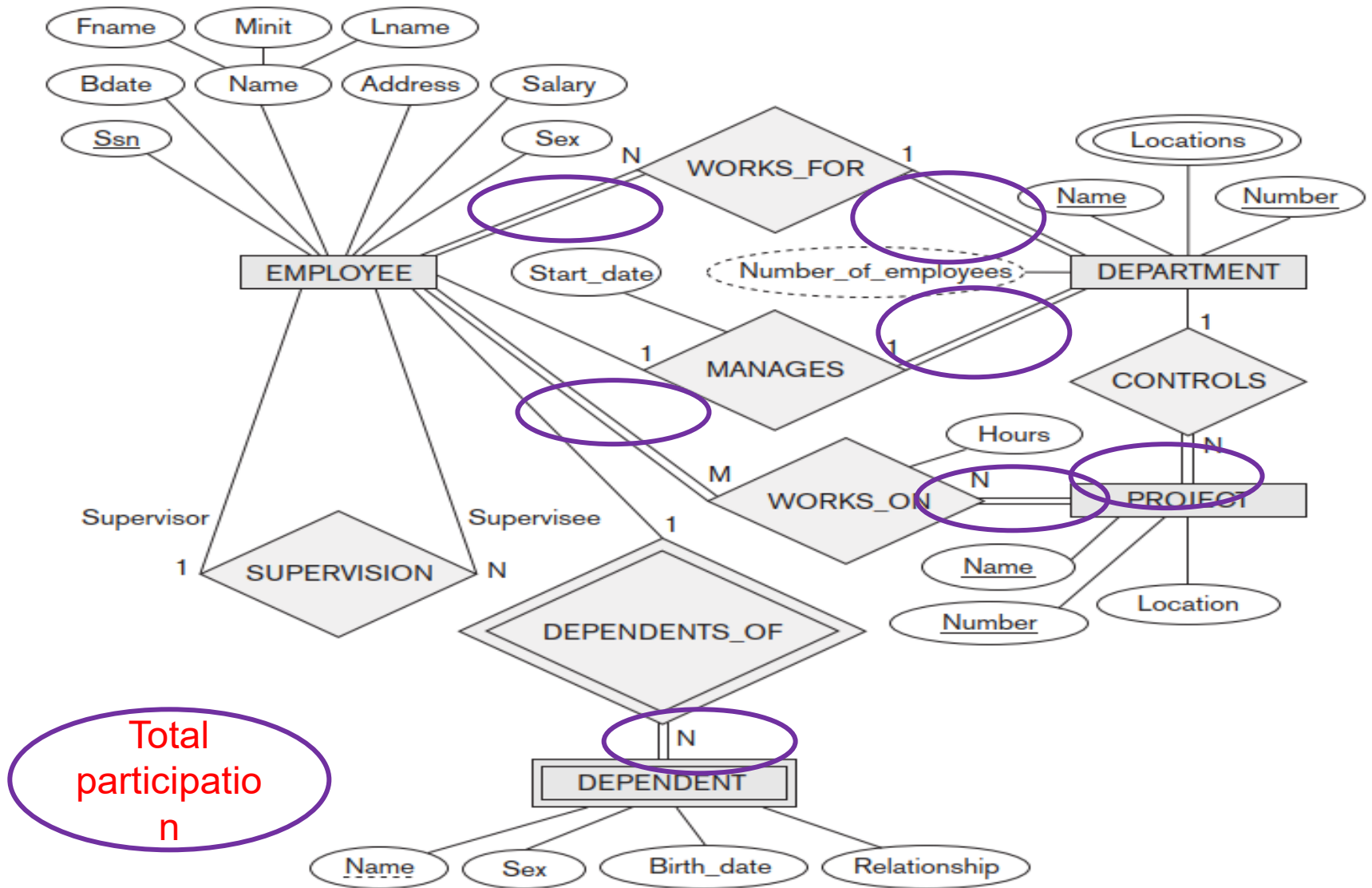
Participation Constraints and Existence Dependencies (Total Participation)

- **Participation Constraints** tell us that the participation in a relationship can either be **total** or **partial**.
- Example, if a company policy states that every employee must work for a department, then an employee entity can exist only if it participates in at least one WORKS_FOR relationship instance.
- Thus, the participation of EMPLOYEE in WORKS_FOR is called **total participation (existence dependency)**,
- However, when all entities in the given entity set do not participate in a relation, it is called **partial participation**



Structural Constraints (Structural properties of a DBMS).

- Cardinality ratios and participation constraints taken together are called **structural constraints**.
- The name constraints refer to the fact that such limitations must be imposed on the data, for the DBMS system to be consistent with the requirements.
- In ER diagrams, total participation (or existence dependency) is displayed as a **double line connecting** the participating entity type to the relationship, whereas partial participation is represented by a **single**.

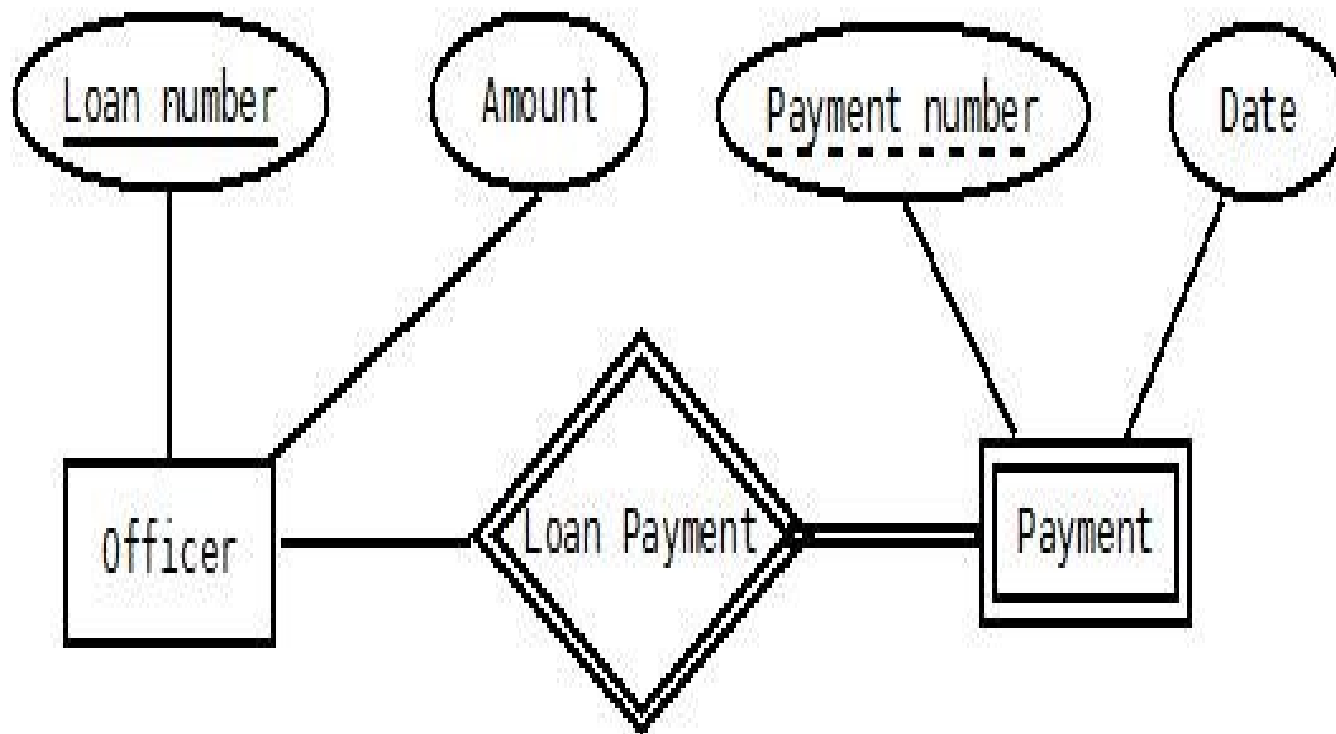


Weak Entity Types

- Entity types that do not have **key attributes of their own**.
- In contrast, regular entity types that do have a **key attributes** are called **strong entity types**.
- As the weak entities do not have any key attribute, they **cannot be identified on their own**, so they depend on some other entity (known as owner entity).
- A weak entity type always has a **total participation constraint** (existence dependency) with respect to its identifying relationship because a weak entity cannot be identified without an owner (parent) entity.

Weak Entity Types...

- For example, a DRIVER_LICENSE entity cannot exist unless it is related to a PERSON entity.
- Weak entity types have **partial keys**. Partial Keys are set of attributes with the help of which the tuples of the weak entities can be distinguished and identified.
- Weak entity depends on strong entity to ensure the existence of weak entity.
- Weak entity is represented by **double rectangle**.
- The relation between one strong and one weak entity is represented by **double diamond**.



Weak Entity Types...

Example: In the ER Diagram in Figure above, 'Payment' is the weak entity.

'Loan Payment' is the identifying relationship and Payment Number is the partial key.

Attribute key of the Loan along with the partial key would be used to identify the records.

- Other examples of weak and strong entities:

Strong entity

Order

Employee

Class

Host

Weak entity

Order Item

Dependent

Section

Logins

- **Note** – Strong-Weak entity set always has parent-child relationship.

Weak Entity Types...

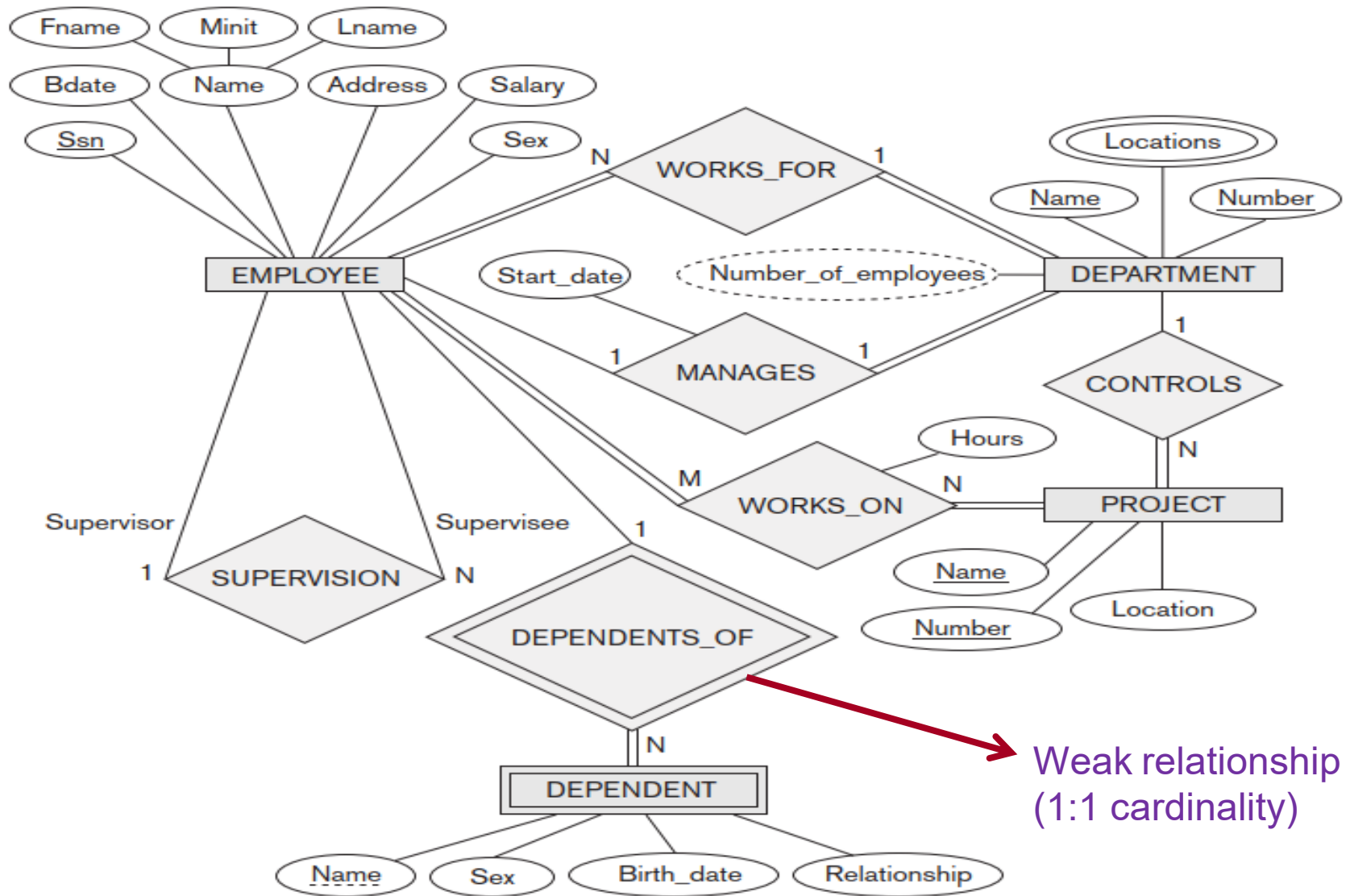
Example: In the ER Diagram in Figure above, 'Payment' is the weak entity.

'Loan Payment' is the identifying relationship and

Payment Number is the partial key.

Attribute key of the Loan along with the partial key would be used to identify the records.

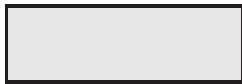
Strong-Weak entity set always has parent-child relationship.



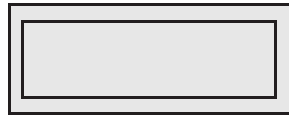
ER Diagrams, Naming Conventions

Symbol

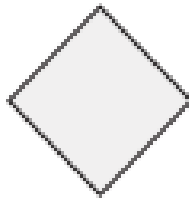
Meaning



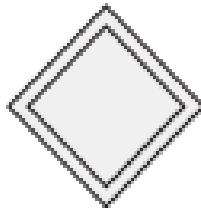
Entity



Weak Entity



Relationship



Identifying Relationship

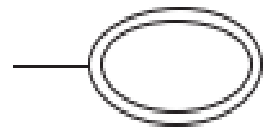
ER Diagrams, Naming Conventions



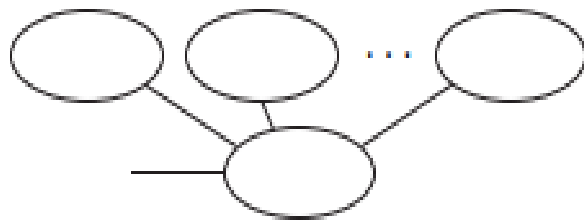
Attribute



Key Attribute



Multivalued Attribute

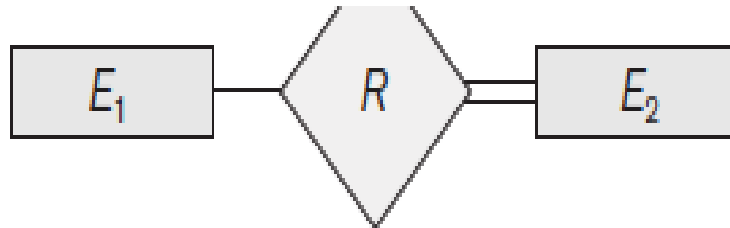


Composite Attribute

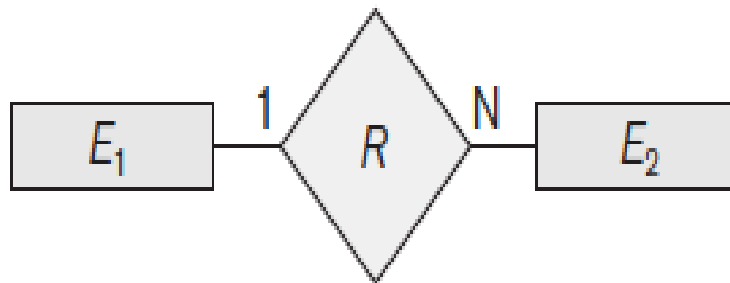


Derived Attribute

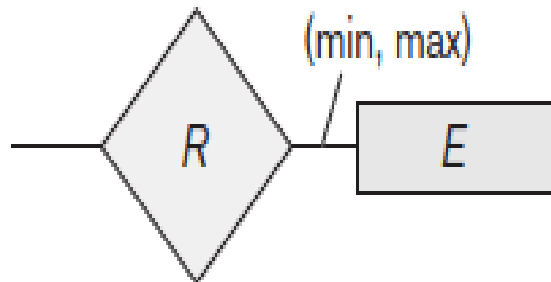
ER Diagrams, Naming Conventions



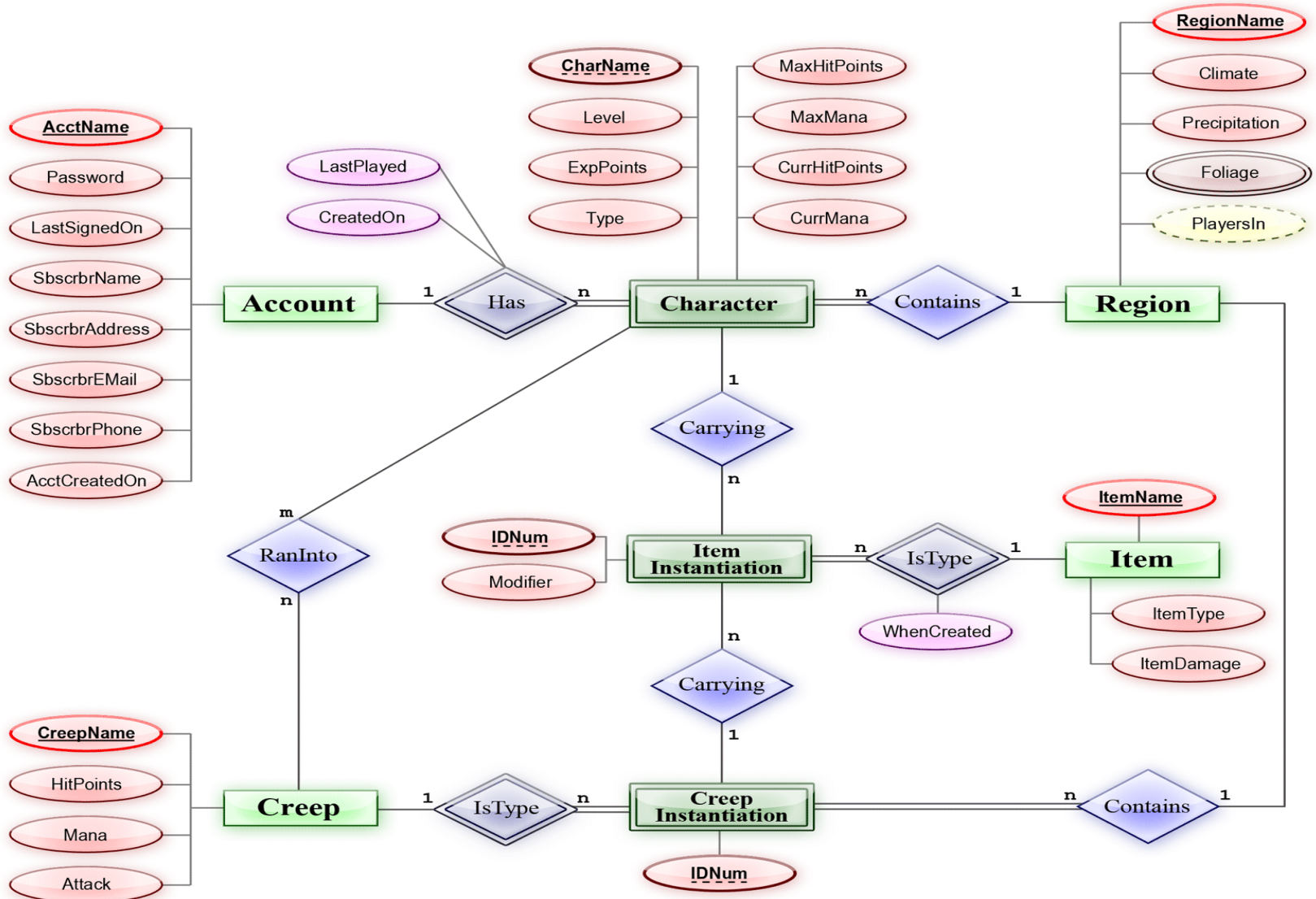
Total Participation of E_2 in R



Cardinality Ratio 1 : N for $E_1 : E_2$ in R



Structural Constraint (min, max)
on Participation of E in R



Abdulmelik Aman