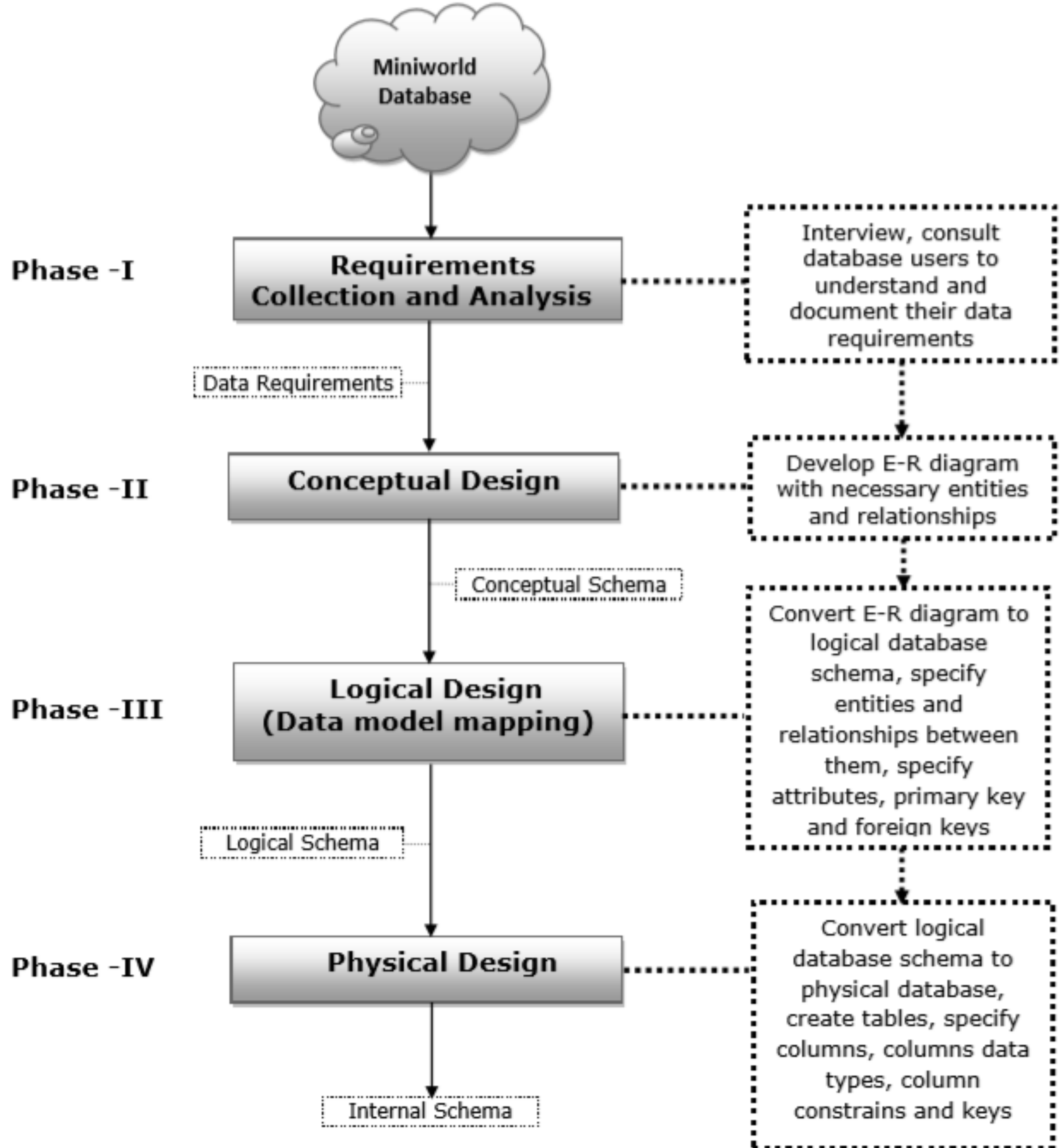
ENTITY RELATIONSHIP (ER)

Database Management System (DBMS)
and Programming

Database Design Process



Miniworld Database

Phase -I

Requirements Collection and Analysis

Interview, consult database users to understand and document their data requirements

Data Requirements

Phase -II

Conceptual Design

Develop E-R diagram with necessary entities and relationships

Conceptual Schema

Phase -III

Logical Design (Data model mapping)

Convert E-R diagram to logical database schema, specify entities and relationships between them, specify attributes, primary key and foreign keys

Logical Schema

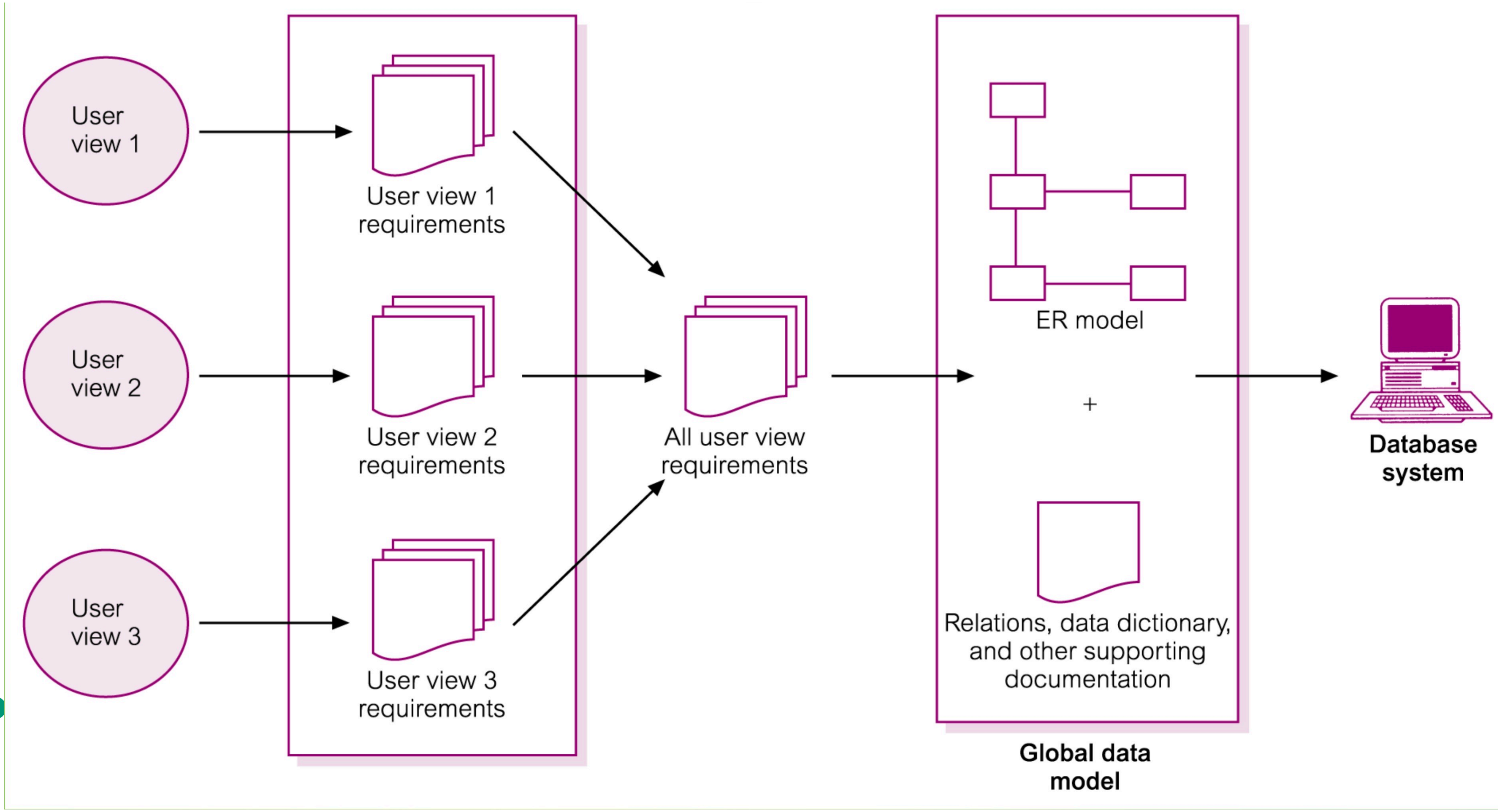
Phase -IV

Physical Design

Convert logical database schema to physical database, create tables, specify columns, columns data types, column constrains and keys

Internal Schema

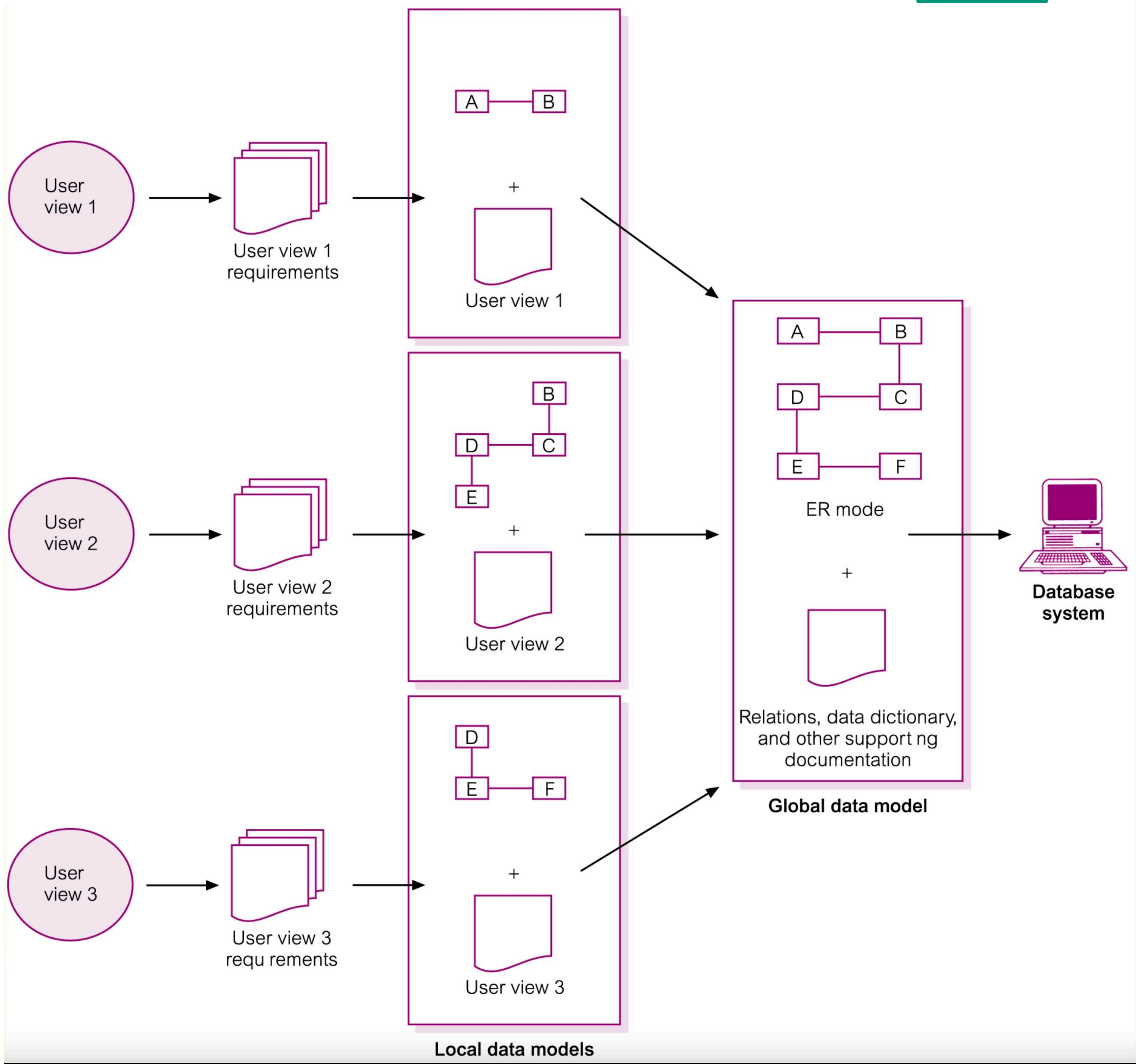
Requirements Collection and Analysis



The centralized approach

Requirements Collection and Analysis

The view integrated approach



Entity-Relationship (ER) Modeling

Conceptual Design Using the Entity-Relationship (ER) Model

- **Entity-Relationship (ER) model**
  - Popular high-level conceptual data model
  - Diagrammatic notation associated with the ER model
- **Unified Modeling Language (UML)**

ER model

- ER model describes data as:
  - Entities
  - Attributes
  - Relationships

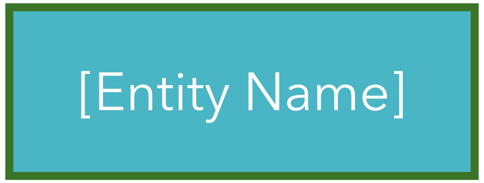
ENTITY

Entity

  - A group of objects with the same properties which are identified by the enterprise having an independent existence
  - Object in the “real world”
- ER Diagram symbol for entity
  - A rectangle with the name of the entity

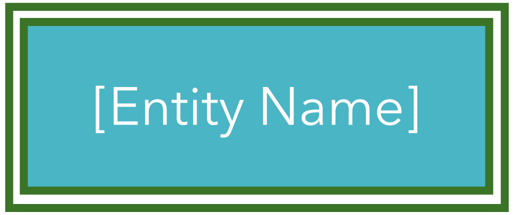
Entity

- 2 types
  - Strong entity
    - Independent entity
    - Can identify its characters

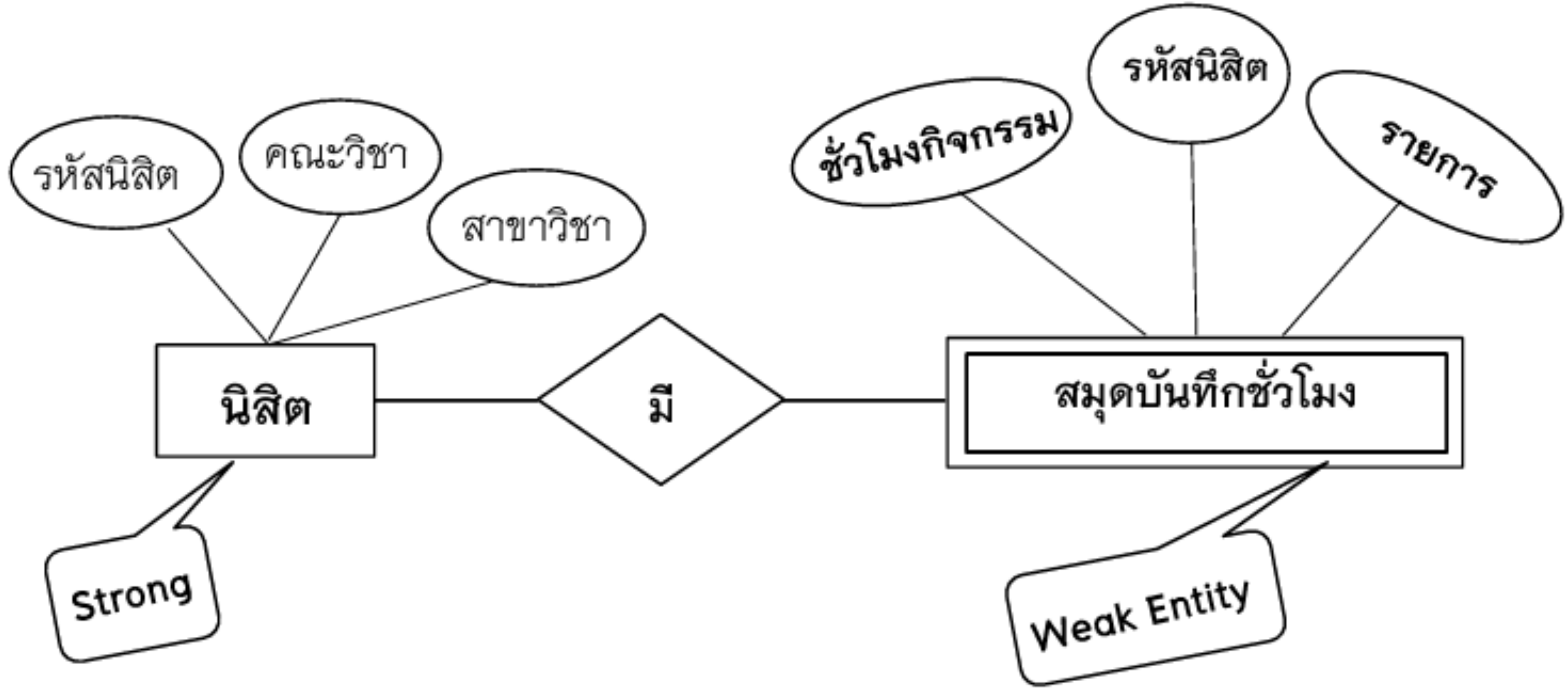


[Entity Name]

  - Weak entity
    - An entity that **cannot** be uniquely identified by its attributes alone
    - It must use a *foreign key* in conjunction with its attributes to create a primary key.
    - Dependent entity
    - If the main entity is deleted, the weak entity also be deleted



[Entity Name]


รหัสนิสิต
คณะวิชา
สาขาวิชา
นิสิต
มี
ชั่วโมงกิจกรรม
รหัสนิสิต
รายการ
สมุดบันทึกชั่วโมง
Strong
Weak Entity

Entity Naming Convention

- Should be a meaningful word
- Should be a singular noun

EMPLOYEE

DEPARTMENT

PROJECT

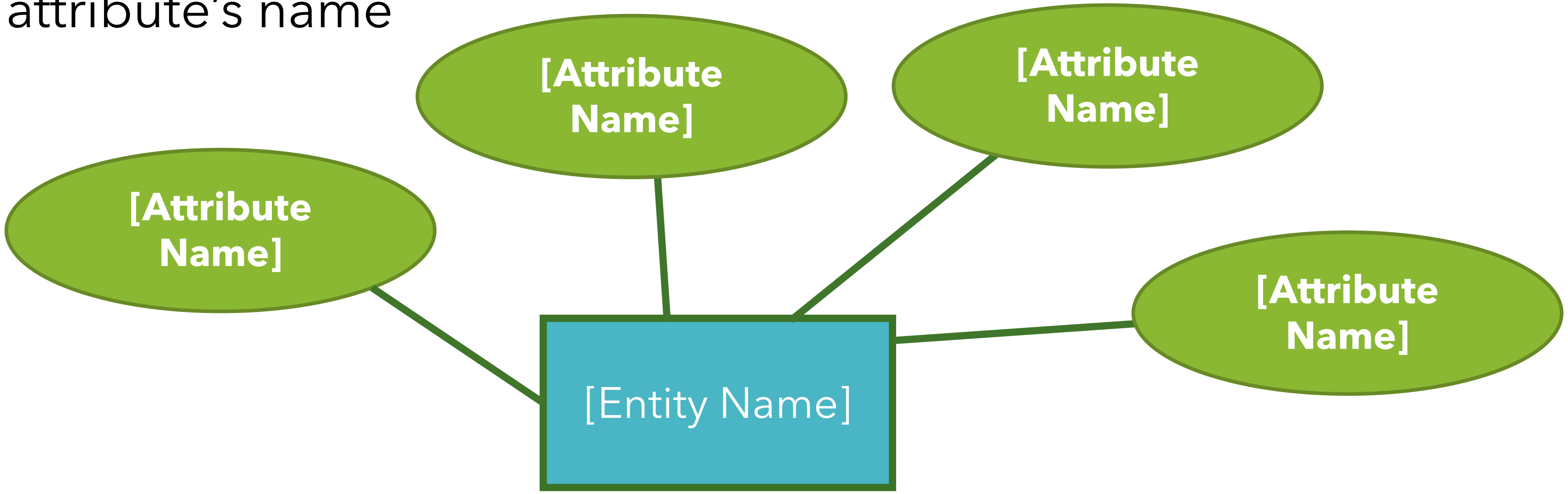
Relation vs Entity

- **Entity** is a real-world concept, such as "a person" in your example.
- **Relation** is a set of tuples (records), each representing a model of an entity.

ATTRIBUTE

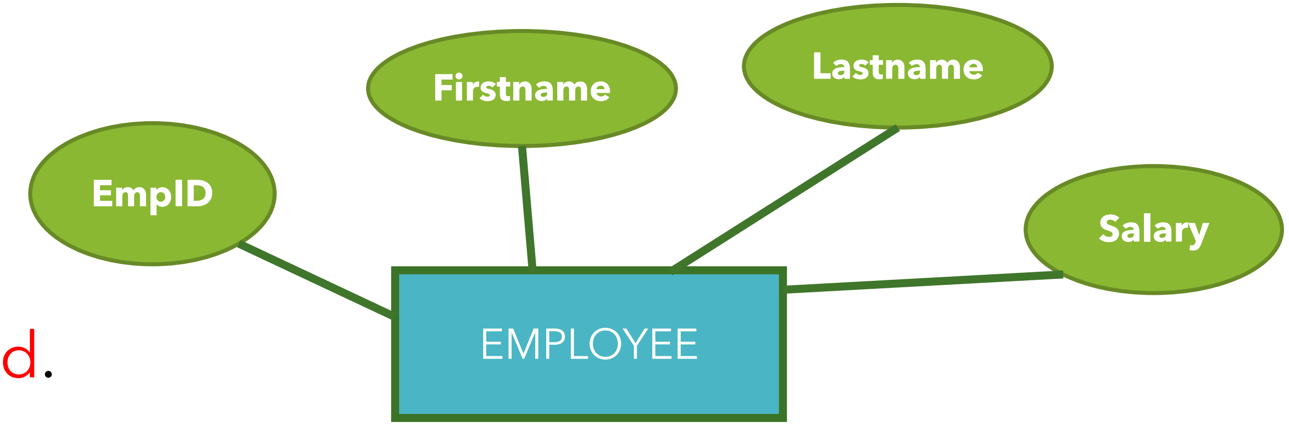
Attributes

- A property of an entity type.
- Something that describes the entity.
- ER diagram symbol for the attribute.
  - An oval with the attribute's name



Attributes

- Naming convention
  - Should be noun.
  - Should be a meaningful word.
  - Must be unique in the same entity
  - Sometime contain the entity name to avoid confusion
    - EmpID, EmpFirstname
  - Must not include the special character.

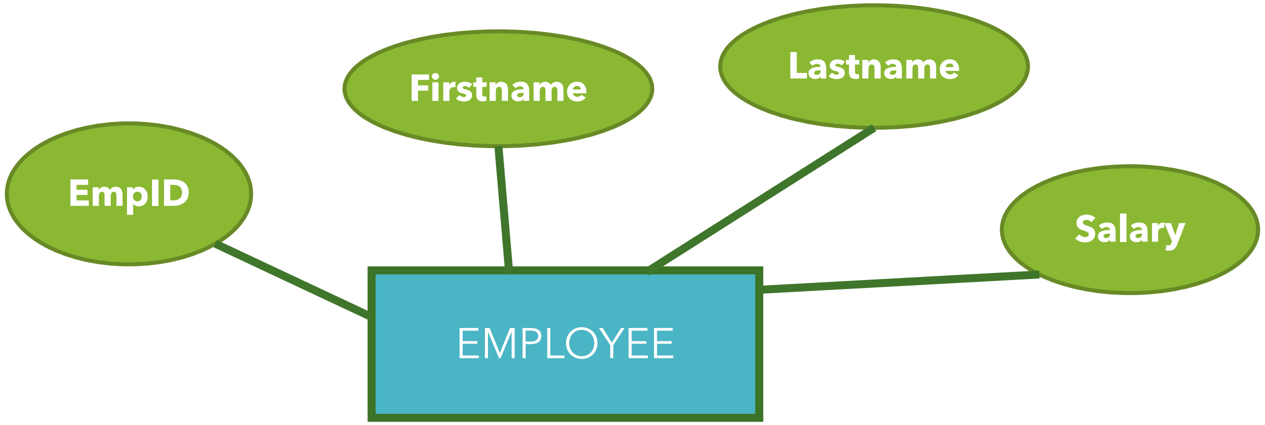


Attributes

- Type of Attribute
  - **Composite** VS **Simple (atomic)** attributes
  - **Single-valued** VS **Multivalued** attributes
  - **Stored** VS **Derived** attributes

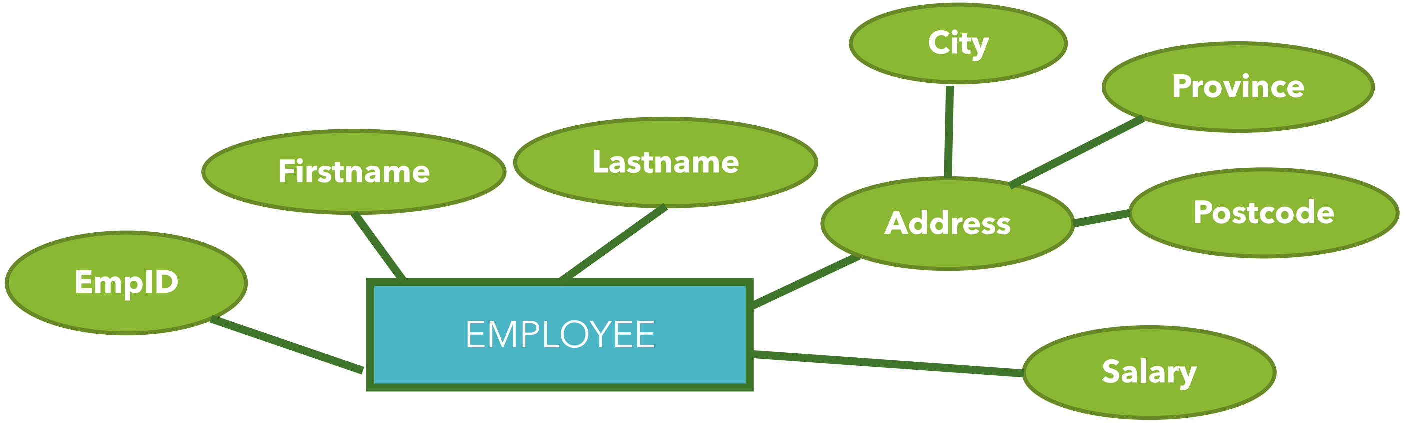
Simple (atomic) vs Composite attributes

- **Simple (atomic) attributes**
  - An attribute composed of a single component with an independent existence.
  - Represented by oval
  - Simple attributes cannot be further subdivided into smaller components.
    - Firstname
    - Lastname
    - Salary



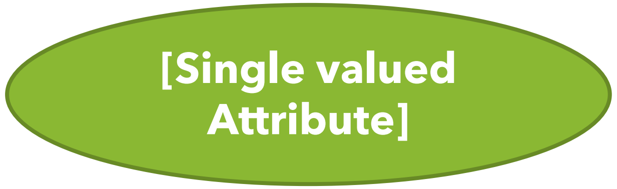
Simple (atomic) vs Composite attributes

- **Composite attributes**
  - An attribute composed of a multiple component with an independent existence.
HNIN
  - Represented by oval with attributes attached to it
  - Can be further subdivided into smaller components.
    - Address
      - City
      - Province
      - Postcode

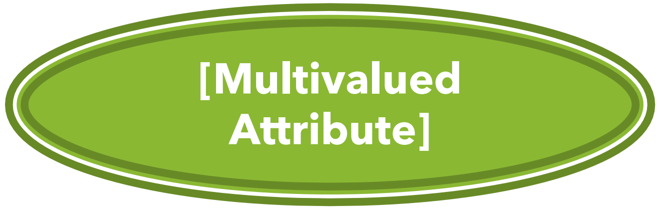


Single-valued vs Multivalued Attributes

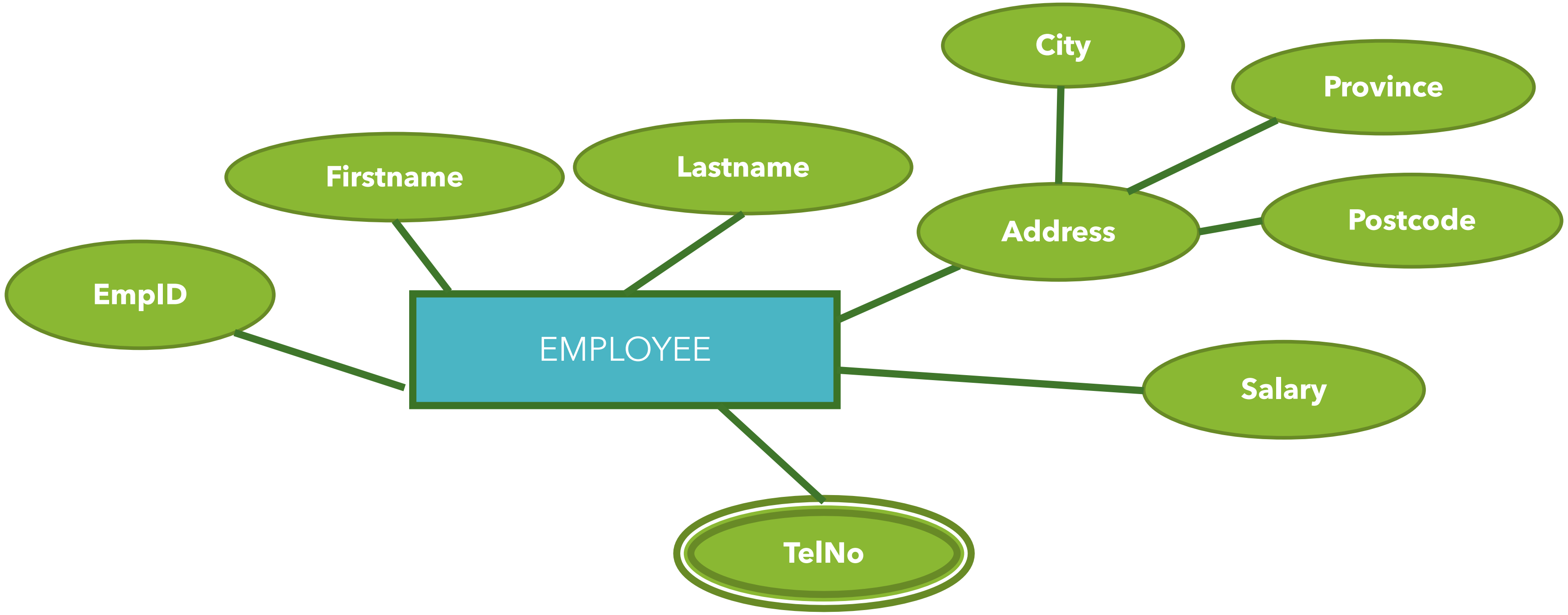
- **Single-valued**
  - An attribute that holds a single value for each occurrence of an entity type
  - Represented by a normal oval (like Simple Attribute)
  - E.g. Firstname, Lastname, Salary


[Single valued Attribute]

- **Multivalued**
  - An attribute that holds multiple values for each occurrence of an entity type
    Represented by a double oval
  - E.g. TelNo
    - 081-1231234, 053-123123


[Multivalued Attribute]

Single-valued vs Multivalued Attributes

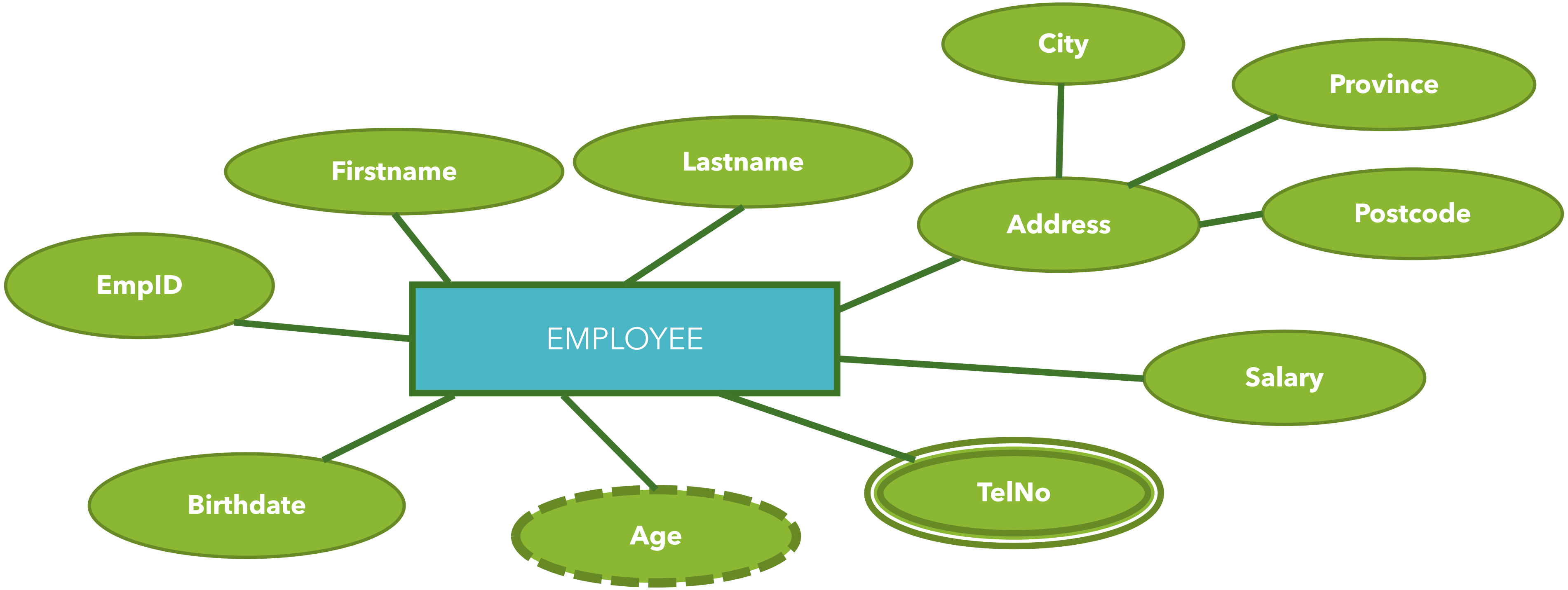


Stored vs Derived attributes

- **Stored Attribute**
  - An attribute that represents a value that is independent from another attribute.
  - Represented by a normal oval (like Simple Attribute)
  - E.g. Firstname, Lastname, Salary

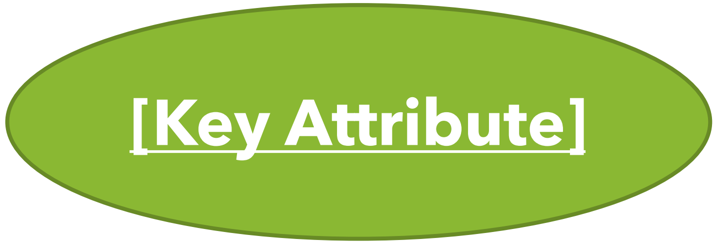
- **Derived Attribute**
  - An attribute that represents a value that is derivable from the value of a related attribute or set of attributes, not necessary in the same entity type.
  - Represented by a dashed oval
  - E.g. Age (age is calculated from date of birth)

Stored vs Derived attributes



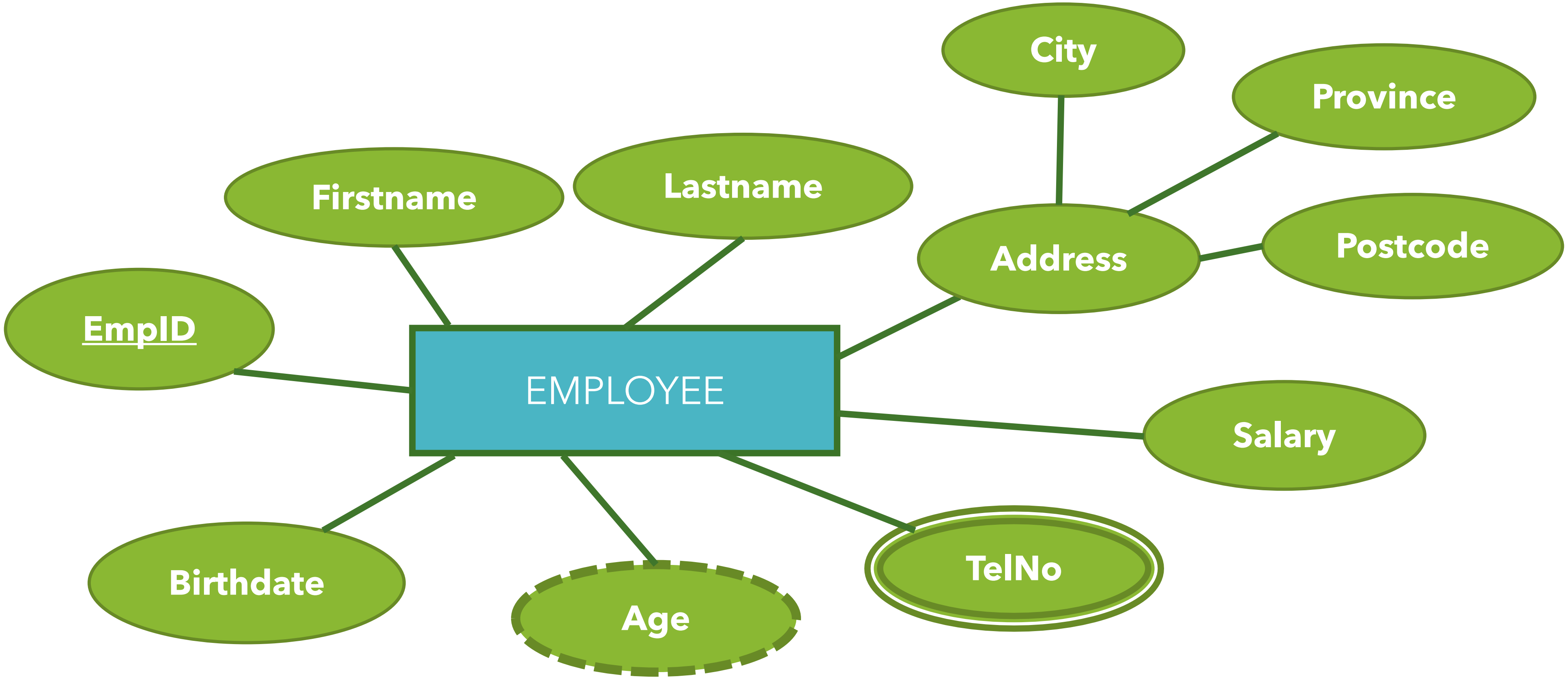
Key Attribute

- Attributes whose values are distinct and unique for each individual entity
- Represent by an oval with underlined name
- Usually has one or more key attributes in the entity



[Key Attribute]

Key Attribute



Create ER Diagram

1. Determine the Entities
    - Identifying the "what" is in your system
    - Entities usually be nouns
    - *Thing* or *object* in the real world
        - Departments
        - Employee
        - Projects
        - Supervisor
        - Manager
2. Add Attributes to Each Entity
    - Attributes that you need to describe each entity
    - Attributes usually be nouns
3. Determine the primary key(s)

Activity 1

Design the entities and attributes of the COMPANY database

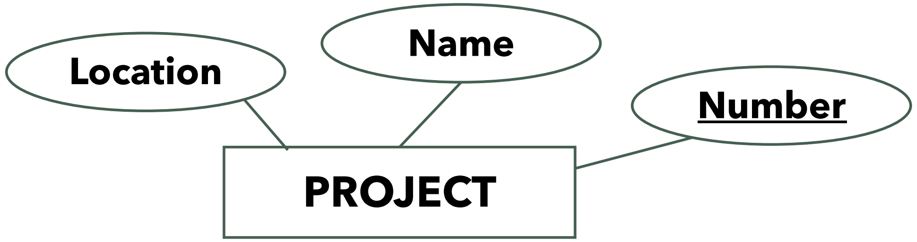
A Sample Database Application

- The database designers provide the following description of the *mini world*—the part of the company that will be represented in the database.
  - The company is organized into departments.
  - Each department has a unique name, a unique number
  - A department may have several locations.
  - A department controls a number of projects, each of which has a unique name, a unique number, and a single location.
  - The database will store each employee's name, Social Security number, address, salary, gender and birthdate.
  - An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department.
  - It is required to keep track of the current number of hours per week that an employee works on each project, as well as the direct supervisor of each employee (who is another employee).
  - A particular employee who manages the department. We keep track of the start date when that employee began managing the department.

A Sample Database Application

- The database designers provide the following description of the *mini world*—the part of the company that will be represented in the database.
  - The company is organized into departments.
  - Each department has a unique name, a unique number
  - A department may have several locations.
  - A department controls a number of projects, each of which has a unique name, a unique number, and a single location.
  - The database will store each employee’s name, Social Security number, address, salary, gender and birthdate.
  - An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department.
  - It is required to keep track of the current number of hours per week that an employee works on each project, as well as the direct supervisor of each employee (who is another employee).
  - A particular employee who manages the department. We keep track of the start date when that employee began managing the department.

Activity

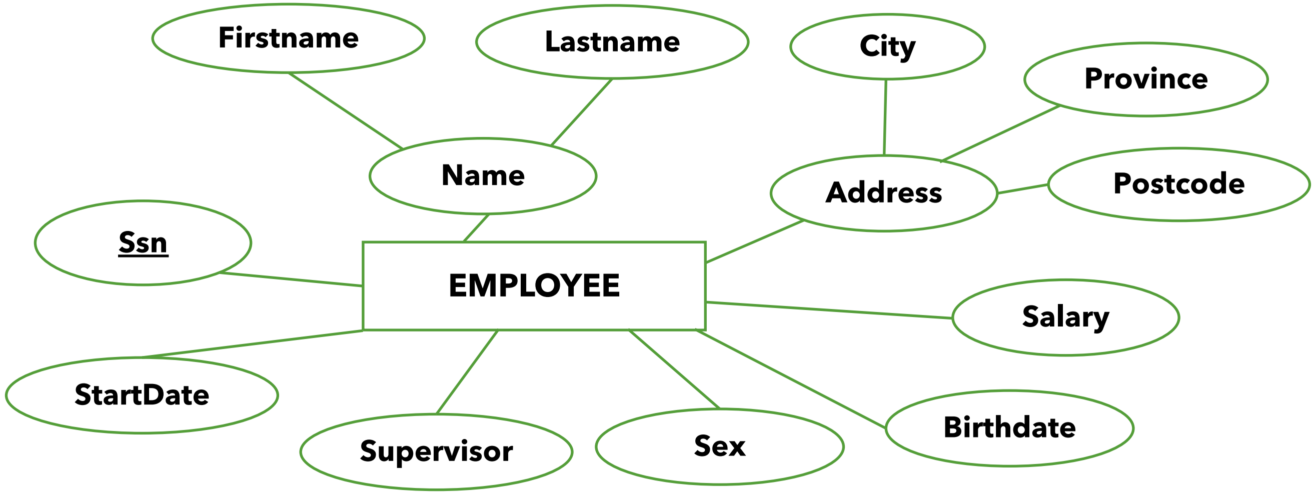


Location

Name

Number

PROJECT



Firstname

Lastname

City

Province

Name

Address

Postcode

Ssn

EMPLOYEE

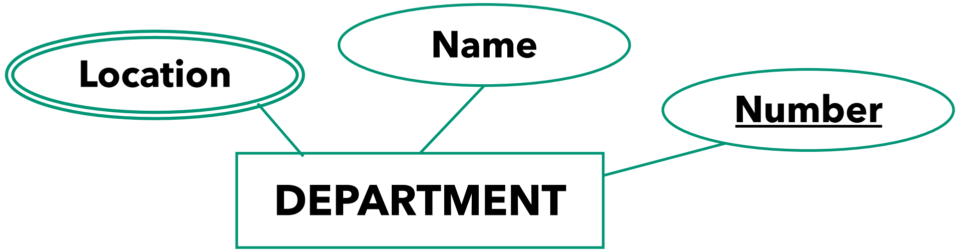
Salary

StartDate

Supervisor

Sex

Birthdate



Location

Name

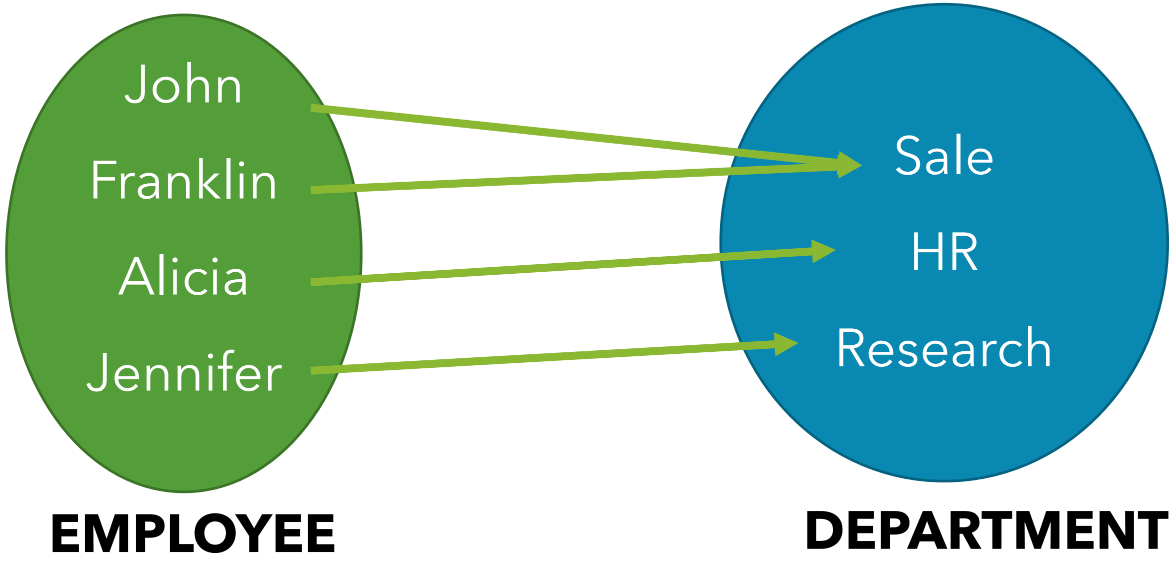
Number

DEPARTMENT

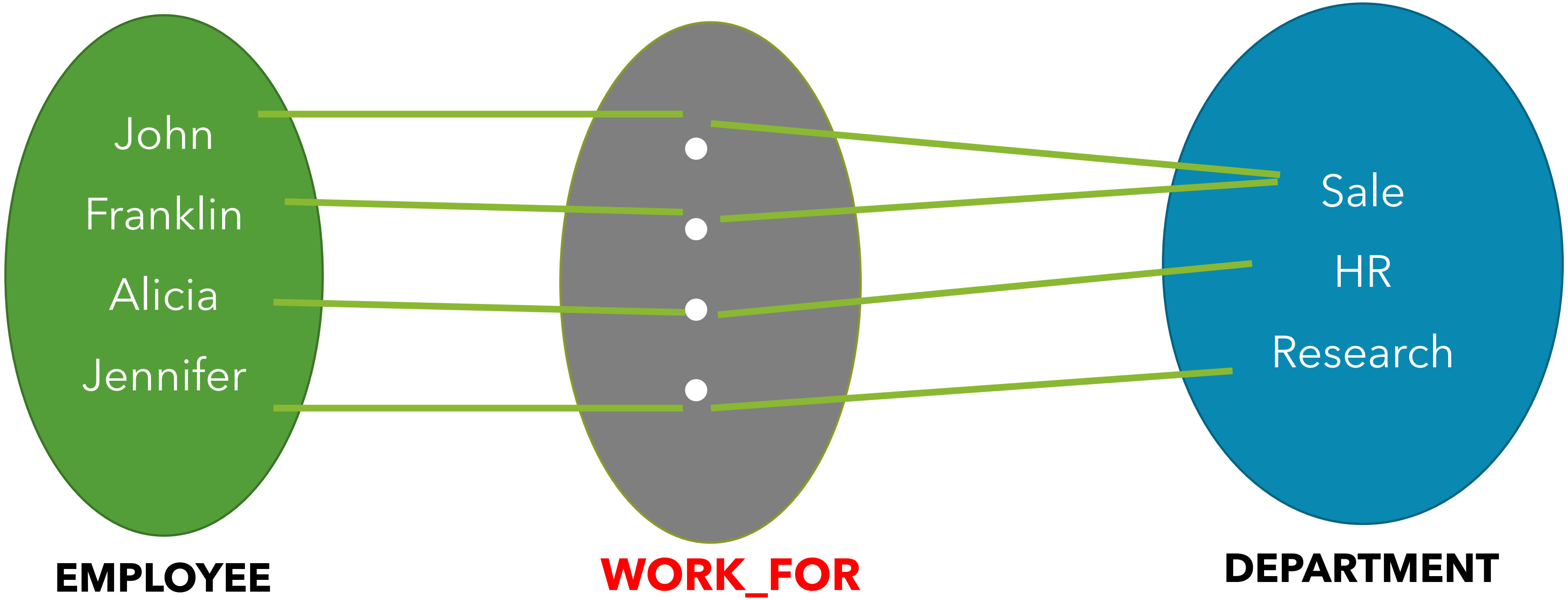
RELATIONSHIP

Relationship

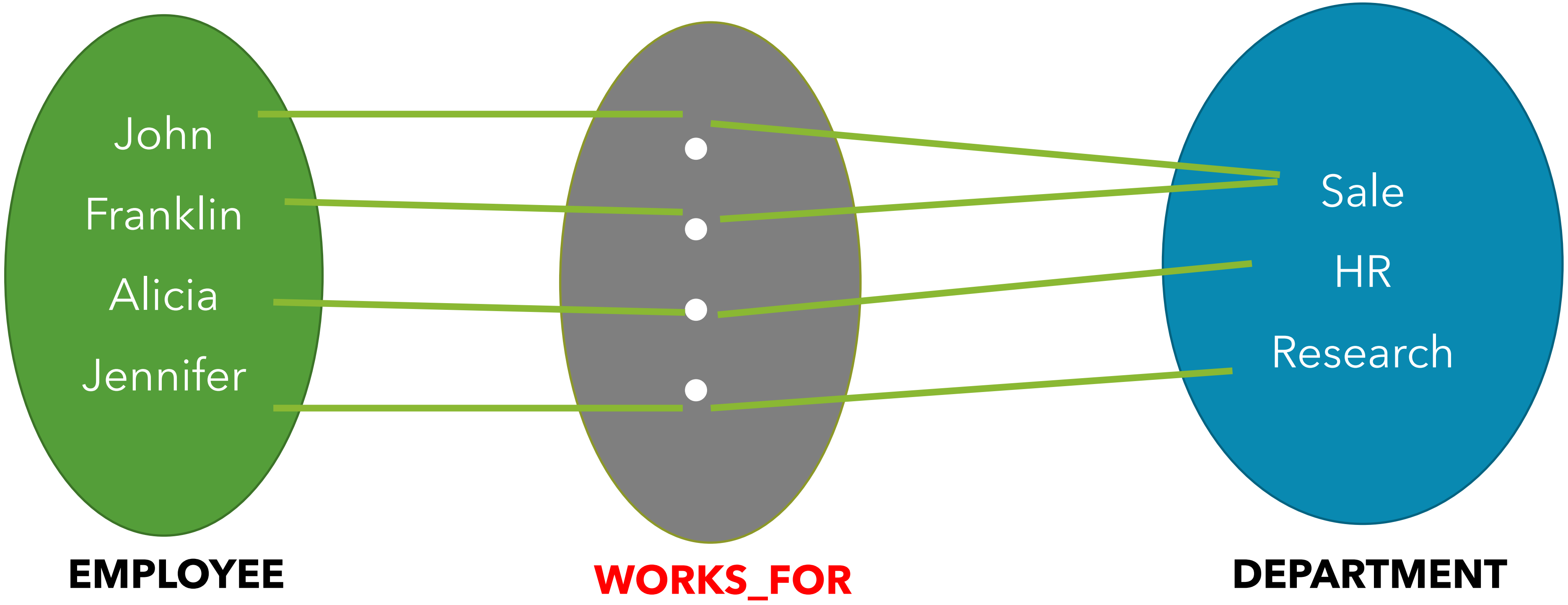
- Relation between 2 or more entities.
- When an attribute of one entity type refers to another entity type
- Represent references as relationships not attributes



Relationship



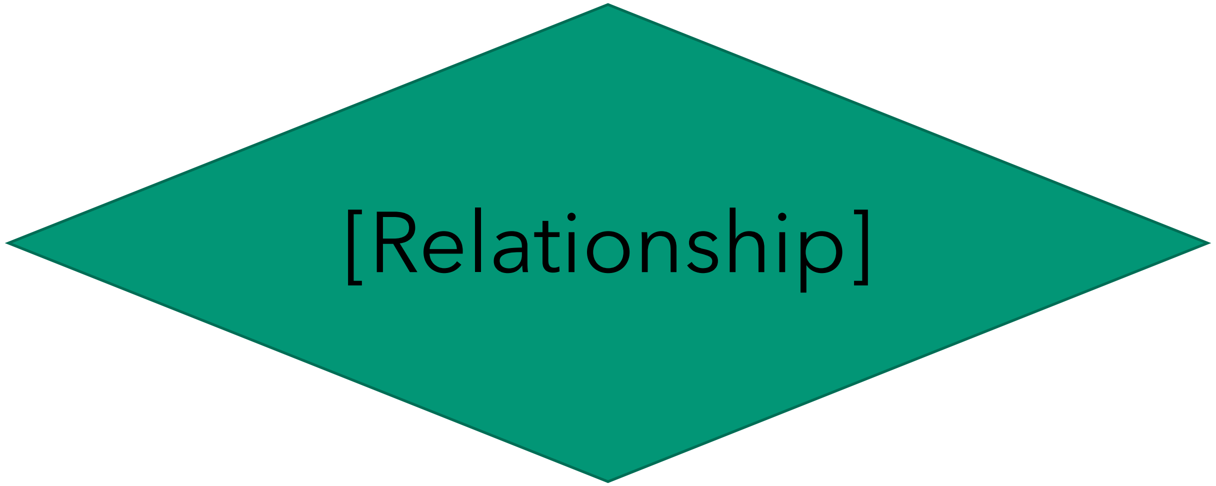
Relationship



- John <u>works for</u> Sale department
- Franklin <u>works for</u> Sale department
- Alicia <u>works for</u> HR department
- Jennifer <u>works for</u> Research department

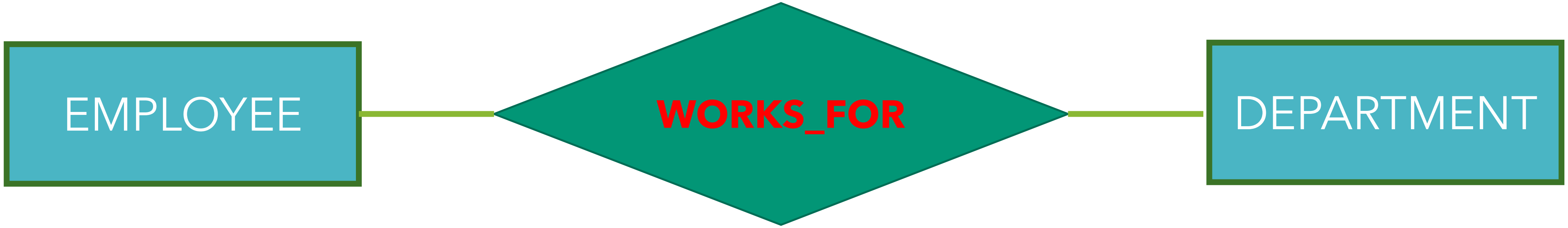
Relationship

- Represented by diamond box in ER diagram
- Relationship naming convention
    - tend to be verbs



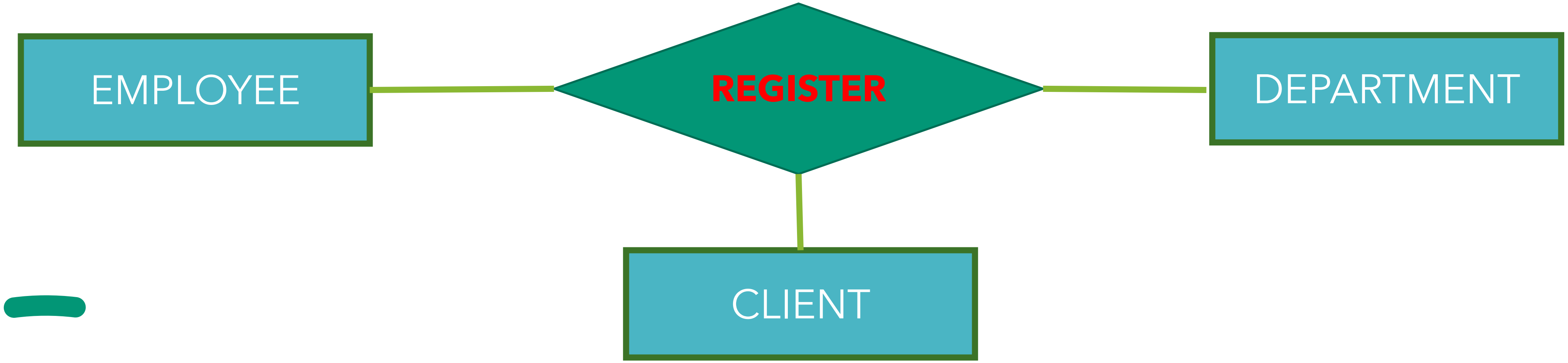
[Relationship]

Relationship



Relationship Degree

- **Degree** of a relationship type
  - Number of participating entity types
  - Binary, ternary, quaternary



Constraints on Binary Relationship Types

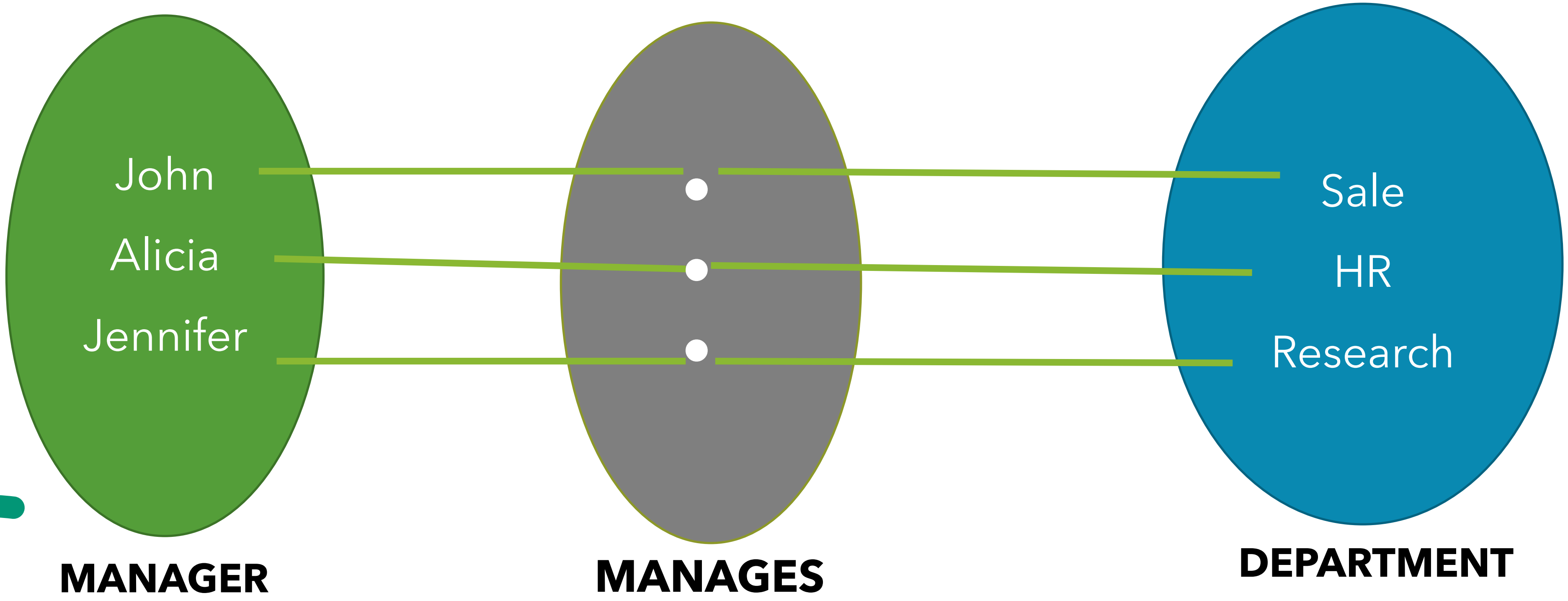
- **Cardinality ratio** for a binary relationship
  - Specifies **maximum** number of relationship instances that entity can participate in
- **Participation constraint**
  - Specifies whether existence of entity depends on its being related to another entity
  - Types: **total** and **partial**

Cardinality ratio

- There are 3 types
  - One-to-one (1:1)
  - One-to-many (1:N)
  - Many-to-many (M:N)

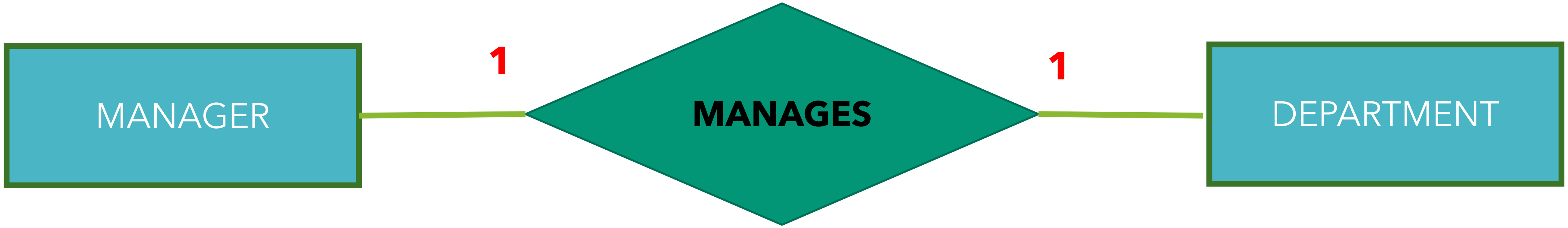
One-to-one (1:1)

- A record in one entity (table) is associated with exactly one record in another entity (table)



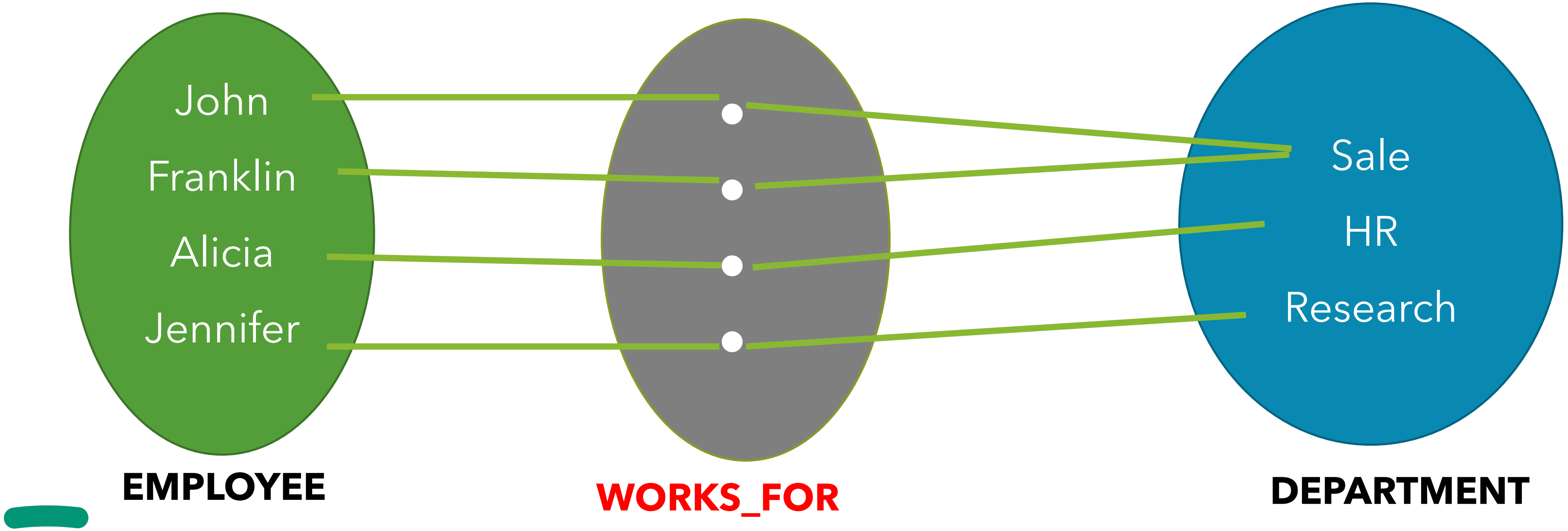
One-to-one (1:1)

- Cardinality ratio = 1:1



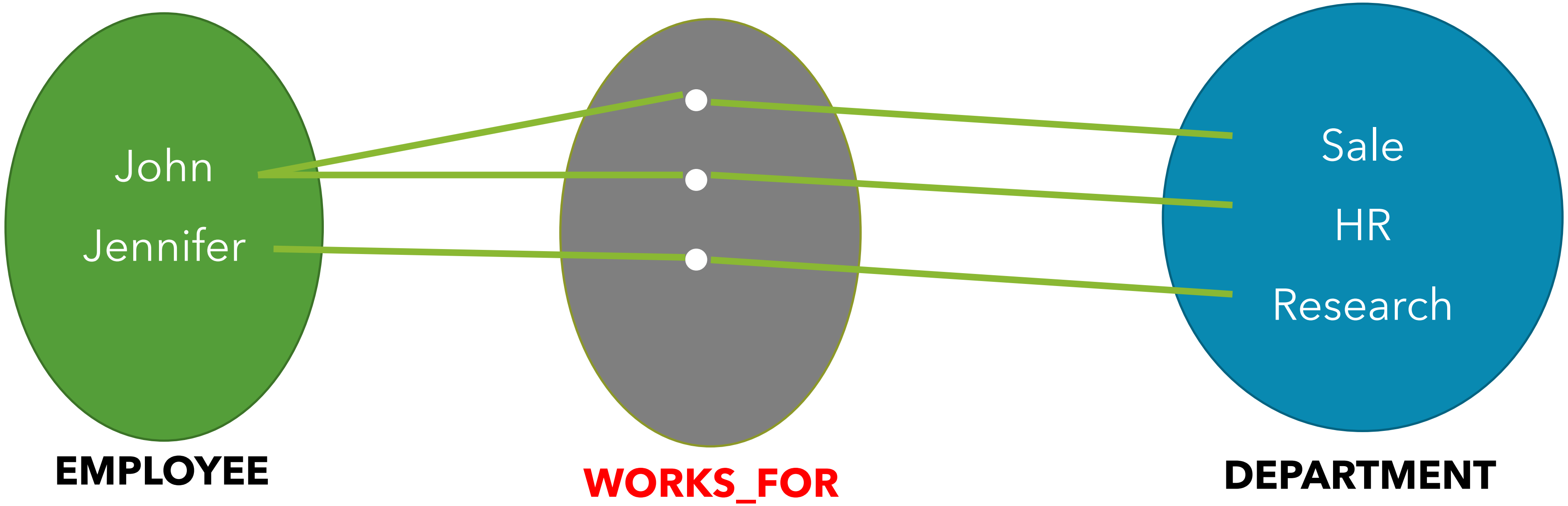
Question

- Is this relationship a one-to-one relationship?



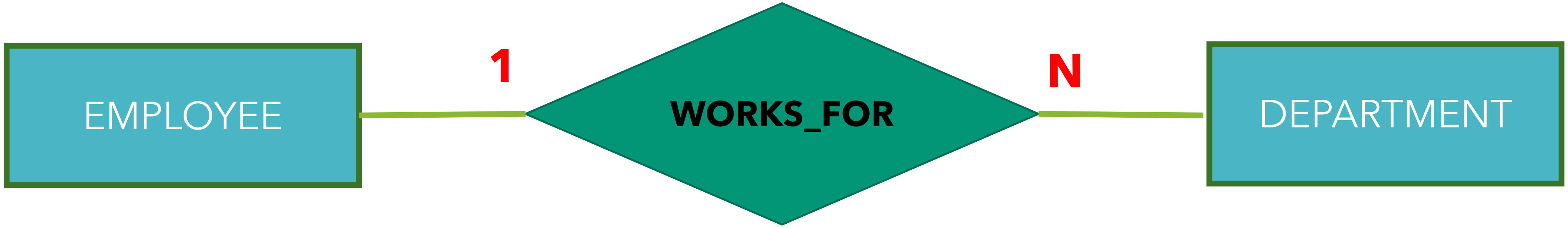
One-to-Many (1:N) Relationship

- A relationship between two entities where there is a single record from one entity associated with multiple records from another entity.



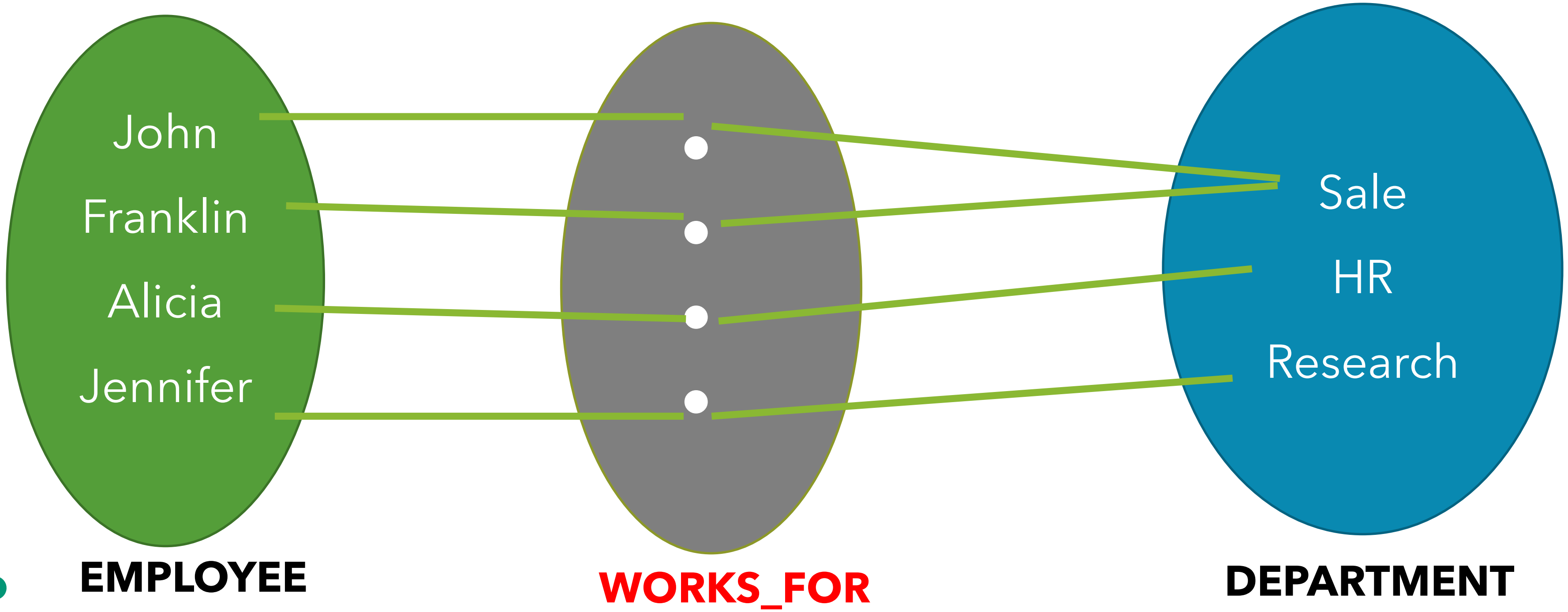
One-to-Many (1:N)

- Cardinality ratio 1:N
- ER Diagram of One-to-Many relationship

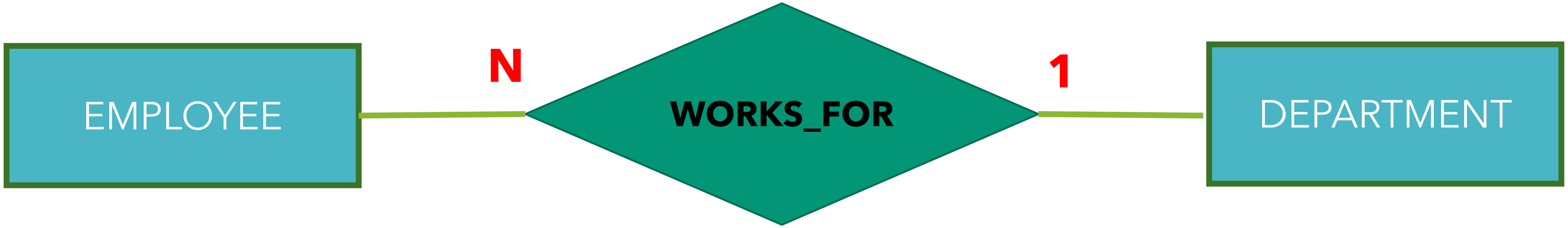


Question

- Is this relationship a one-to-many relationship?

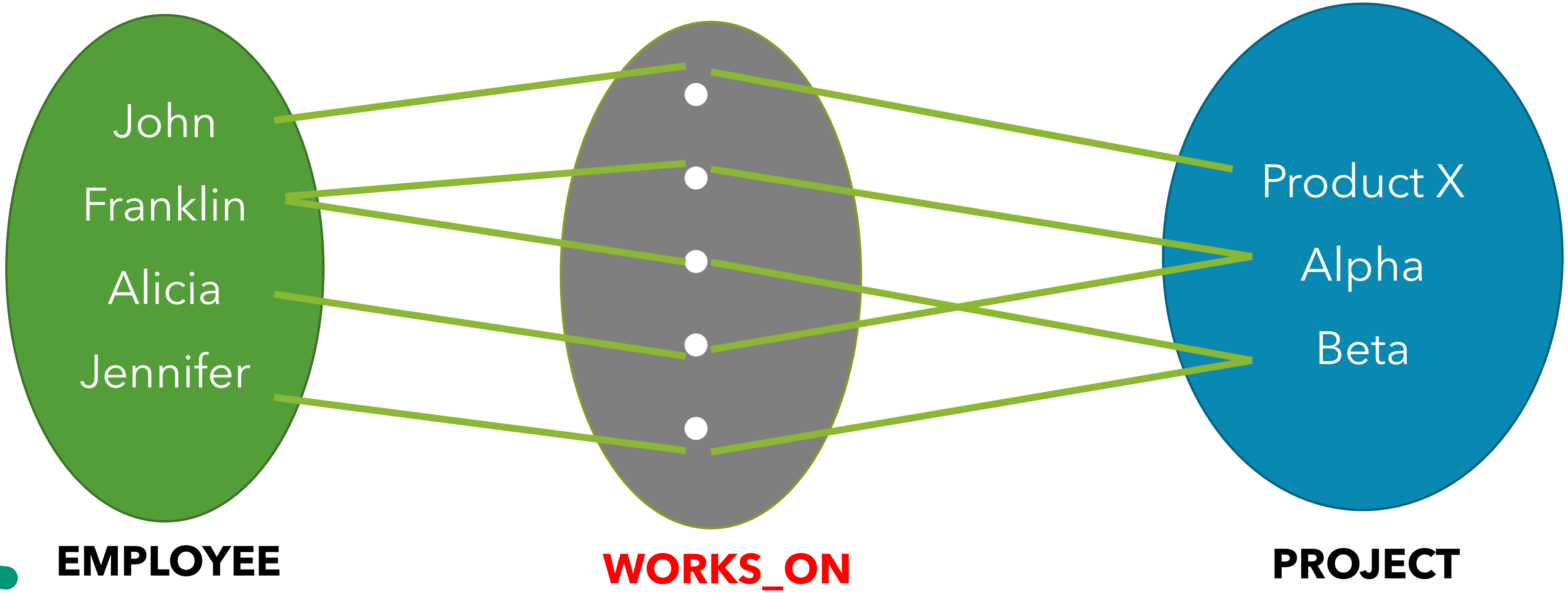


One-to-Many (1:N)



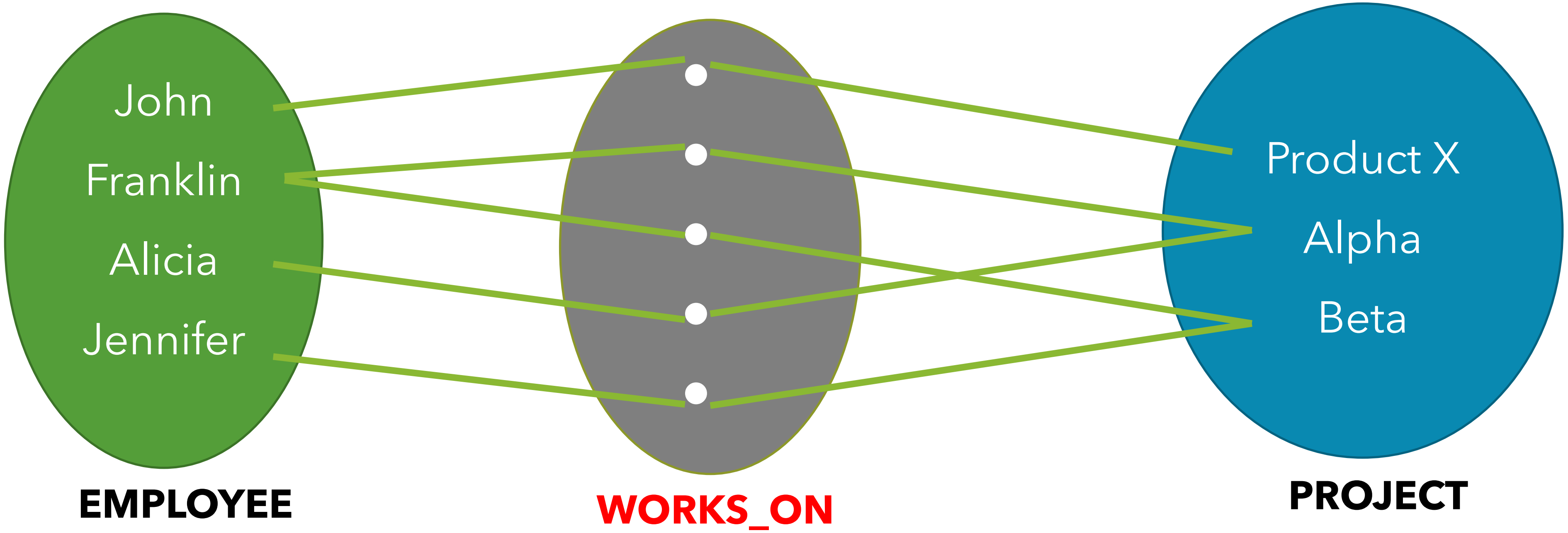
Question

- Is this relationship a one-to-many relationship?



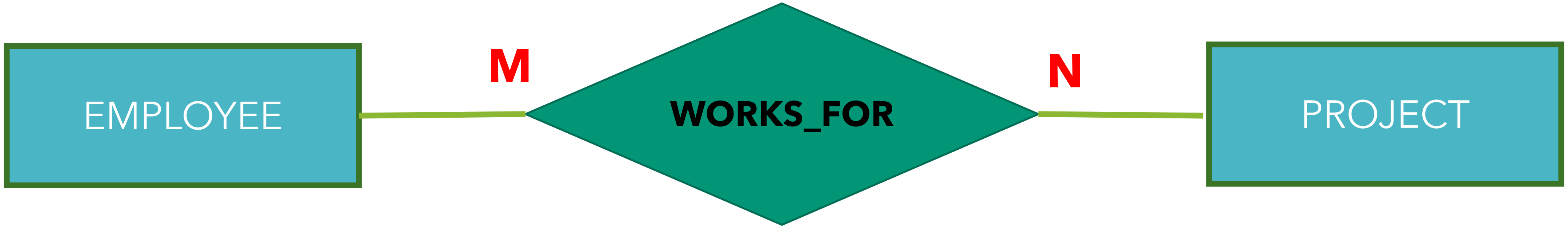
Many-to-Many (M:N) Relationship

- A relationship between two entities where there are multiple records from one entity associated with multiple records from another entity.



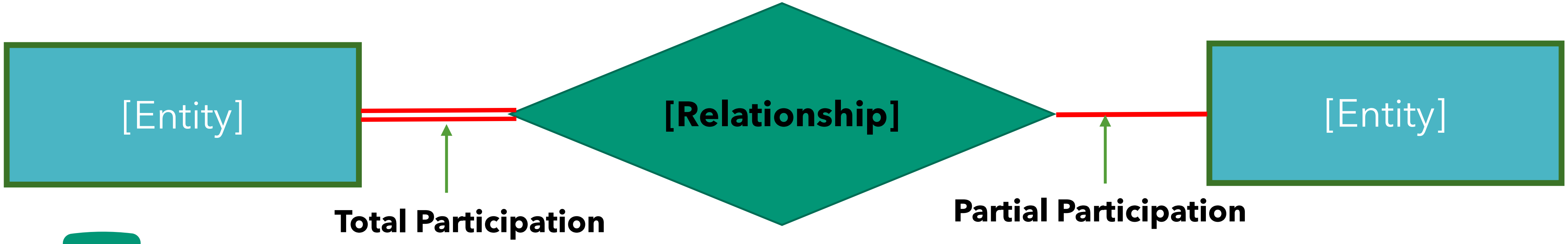
Many-to-Many (M:N)

- Cardinality ratio M:N
- ER Diagram of One-to-Many relationship

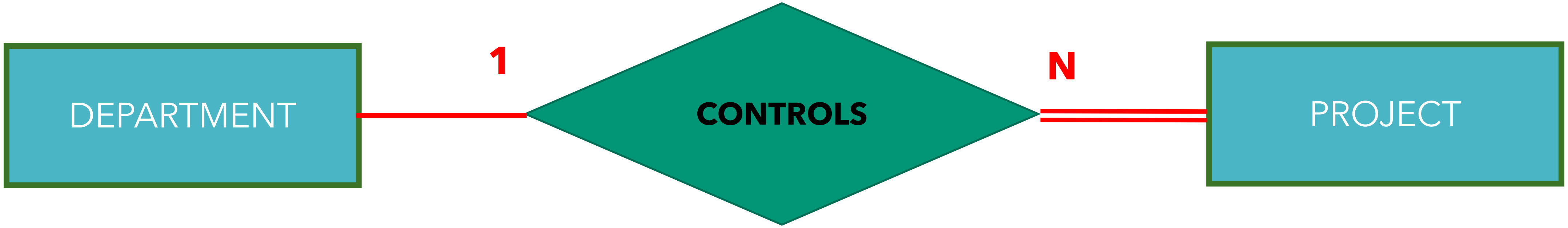


Participation Constraint

- **Total Participation**
  - All entity are involved in the relationship.
  - Total participation is represented by double lines.
- **Partial participation**
  - Not all entities are involved in the relationship
  - Partial participation is represented by single lines.



Participation Constraint



DEPARTMENT

1

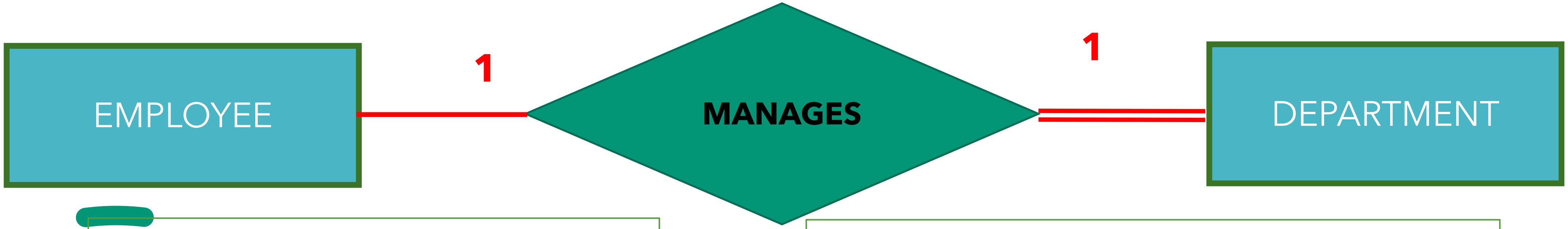
CONTROLS

N

PROJECT

Not all department controls projects

Every project is controled by a department



EMPLOYEE

1

MANAGES

1

DEPARTMENT

Not all employee manges department

Every department is managed by an employee

Create ER Diagram

4. Determine the relationships between the entities
    - Relationships tend to be verb.

Activity 2

Design the relationships and cardinality of the COMPANY database

A Sample Database Application

- The database designers provide the following description of the *mini world*—the part of the company that will be represented in the database.
  - The company is organized into departments.
  - Each department has a unique name, a unique number
  - A department may have several locations.
  - A department controls a number of projects, each of which has a unique name, a unique number, and a single location.
  - The database will store each employee's name, Social Security number, address, salary, gender and birthdate.
  - An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department.
  - It is required to keep track of the current number of hours per week that an employee works on each project, as well as the direct supervisor of each employee (who is another employee).
  - A particular employee who manages the department. We keep track of the start date when that employee began managing the department.

Create ER Diagram

	EMPLOYEE	DEPARTMENT	PROJECT	EMPLOYEE (Manager)	EMPLOYEE (Supervisor)
EMPLOYEE	▨				
DEPARTMENT	▨	▨			
PROJECT	▨	▨	▨		
EMPLOYEE (Manager)	▨	▨	▨	▨	
EMPLOYEE (Supervisor)	▨	▨	▨	▨	▨

Create ER Diagram

	EMPLOYEE	DEPARTMENT	PROJECT	EMPLOYEE (Manager)	EMPLOYEE (Supervisor)
EMPLOYEE		Works for (Assigned to)	Works on		Supervised by
DEPARTMENT			Controls	Managed by	
PROJECT					
EMPLOYEE (Manager)					
EMPLOYEE (Supervisor)					

Activity

- The database designers provide the following description of the *miniworld*—the part of the company that will be represented in the database.
  - The company is organized into departments.
  - Each department has a unique name, a unique number
  - A department may have several locations.
  - A department controls a number of projects, each of which has a unique name, a unique number, and a single location.
  - The database will store each employee's name, Social Security number, address, salary, gender and birthdate.
  - An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department.
  - It is required to keep track of the current number of hours per week that an employee works on each project, as well as the direct supervisor of each employee (who is another employee).
  - A particular employee who manages the department. We keep track of the start date when that employee began managing the department.

Activity

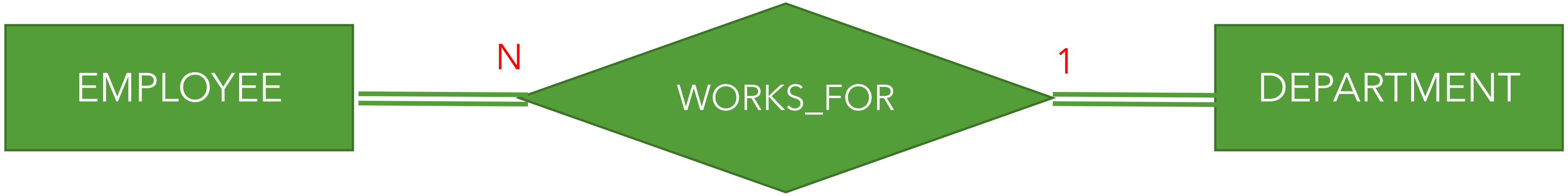
- The database designers provide the following description of the *miniworld*—the part of the company that will be represented in the database.
  - The company is organized into departments.
  - Each department has a unique name, a unique number
  - A department may have several locations.
  - A department controls a number of projects, each of which has a unique name, a unique number, and a single location.
  - The database will store each employee's name, Social Security number, address, salary, sex (gender), and birth date.
  - An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department.
  - It is required to keep track of the current number of hours per week that an employee works on each project, as well as the direct supervisor of each employee (who is another employee).
  - A particular employee who manages the department. We keep track of the start date when that employee began managing the department.

Create ER Diagram

5. Determine the cardinality of each relationship
   - 1:1
   - 1:N
   - M:N

6. Determine participation constraints
   - Total participation
   - Partial participation

Create ER Diagram



one

many

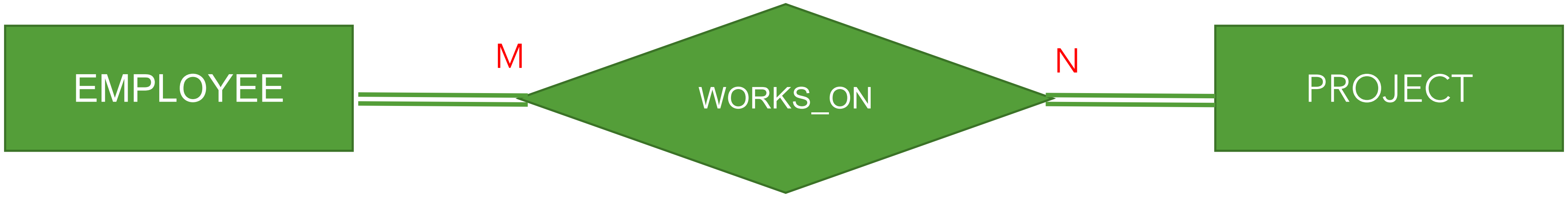
Each employee works for only a department.
Each department is worked by many employee.

Total

Total

Every employee works for a department.
Every department is worked by employees.

Create ER Diagram



Many

Many

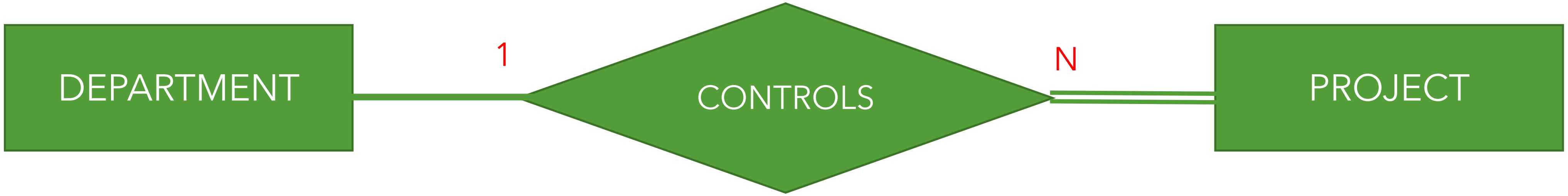
Each employee works on many projects.
Each project is worked by many employees.

Total

Total

Every employee works on projects.
Every project is worked by employees.

Create ER Diagram



many

one

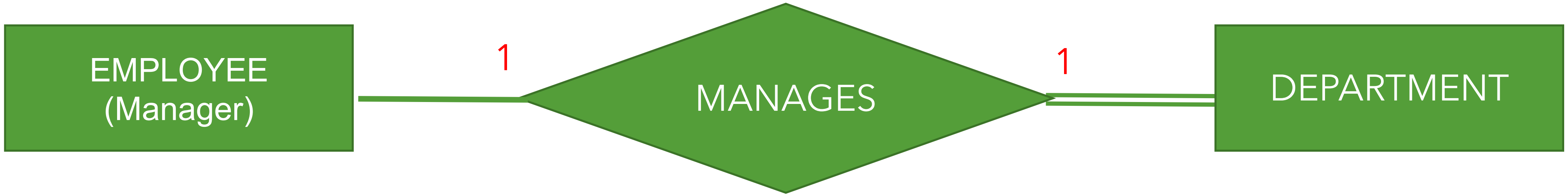
Each department controls many projects.
Each project is controlled by a department.

Partial

Total

Not all department controls projects
Every project is controlled by a department.

Create ER Diagram



one

one

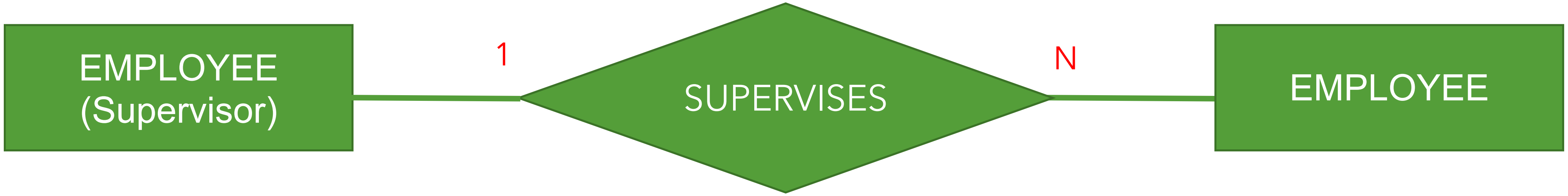
Each manager manages a department.
Each department is managed by a manager.

Partial

Total

Not all manager manages a department.
Every department is managed by a manager.

Create ER Diagram



one

man
y

Each supervisor supervises many employee.
Each employee is supervised by a supervisor.

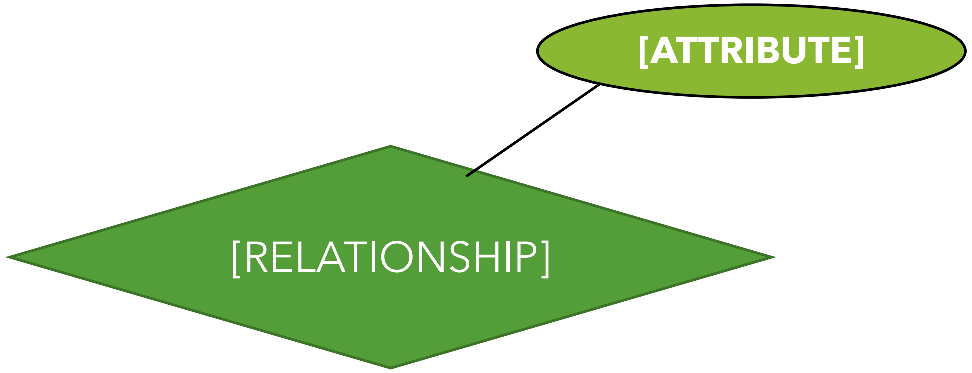
Partial

Partial

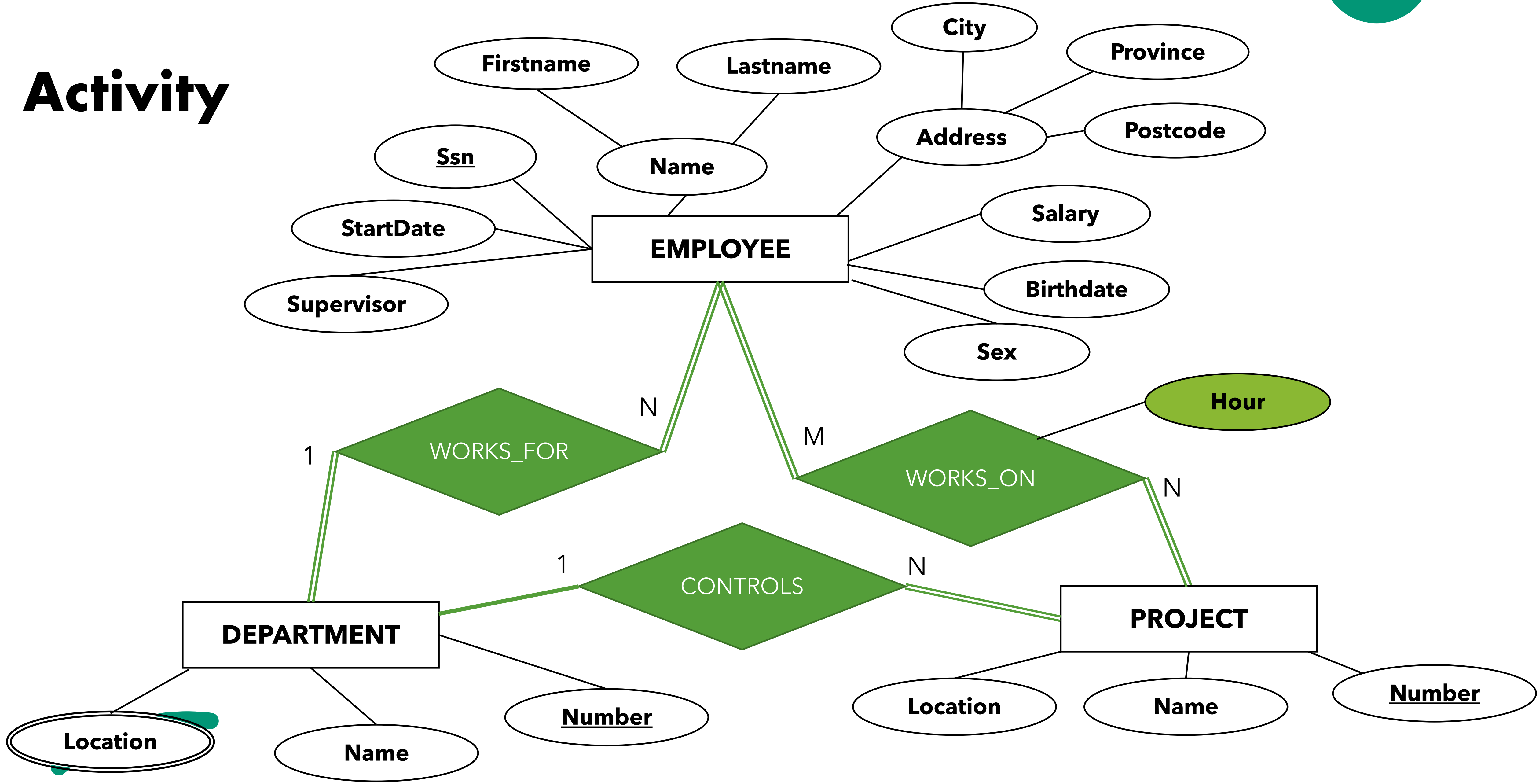
Not all supervisor supervises many employee.
Not all employee is supervised by a supervisor.

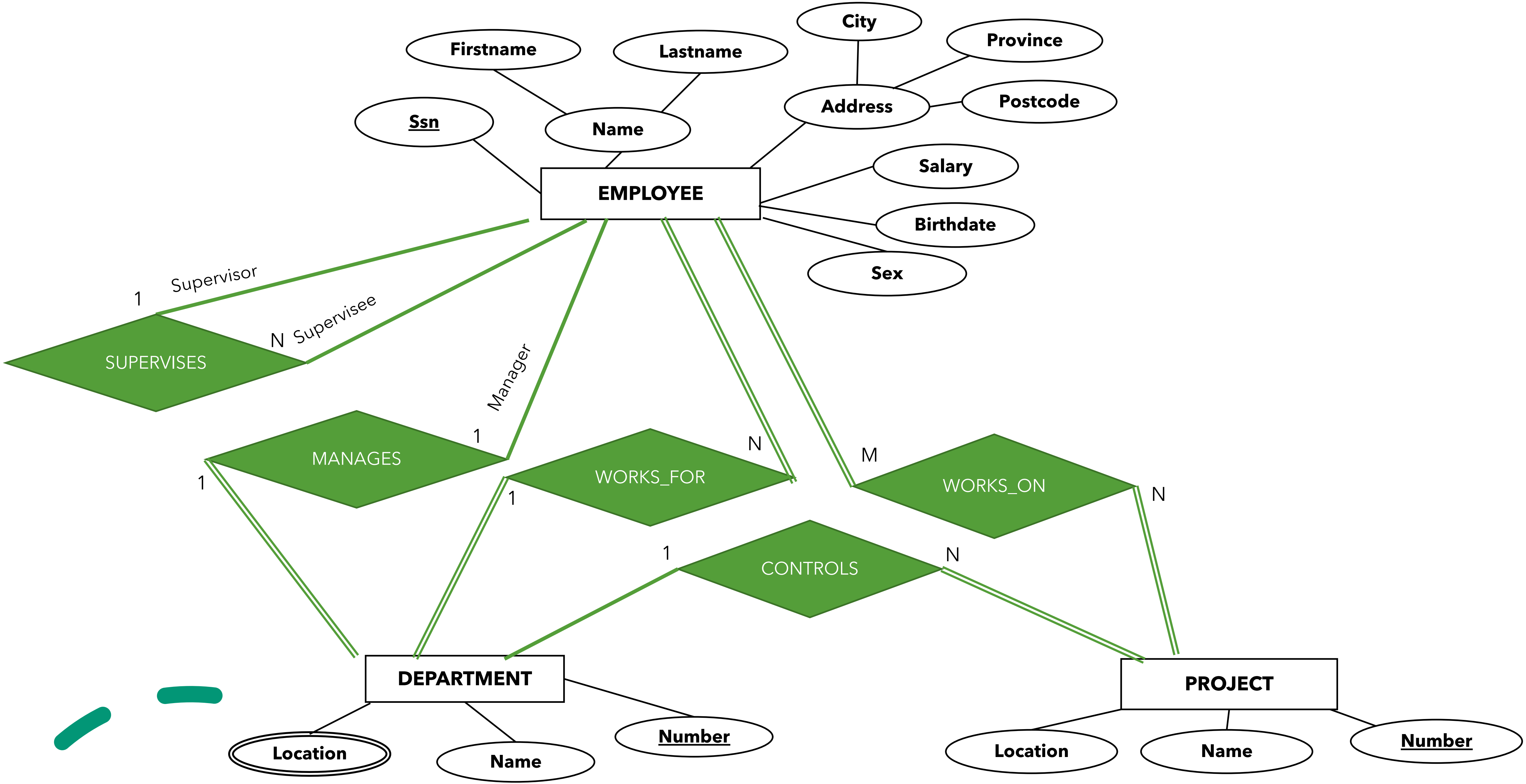
Attributes of Relationship

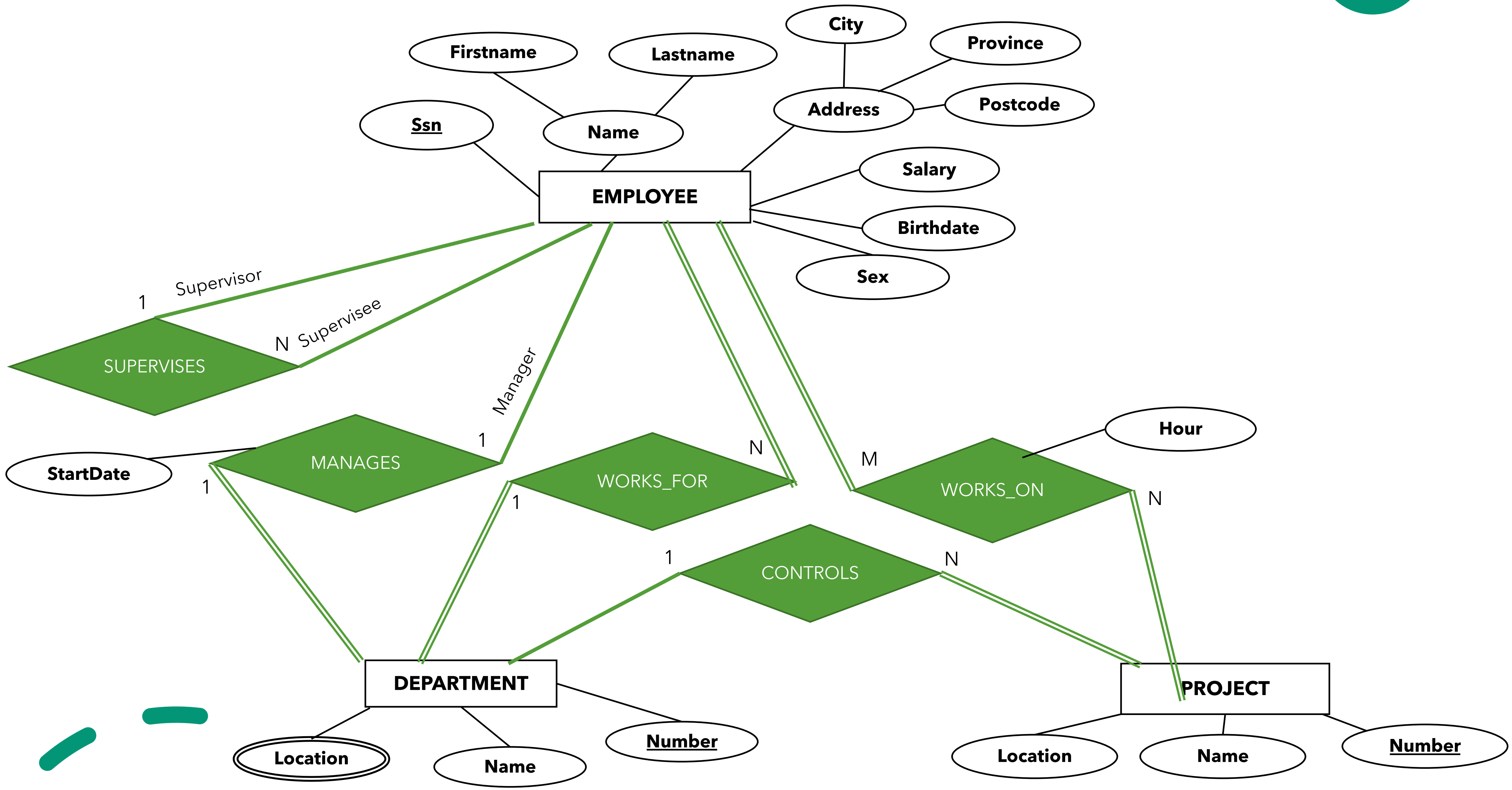
- Relationship can also have attributes.
- Attributes of relationships are nouns or adverbs.



Activity


Firstname
Lastname
City
Province
Postcode
Address
Ssn
Name
StartDate
Supervisor
EMPLOYEE
Salary
Birthdate
Sex
Hour
N
1
WORKS_FOR
M
WORKS_ON
N
1
CONTROLS
N
DEPARTMENT
PROJECT
Location
Name
Number
Location
Name
Number


City
Province
Firstname
Lastname
Address
Postcode
Ssn
Name
Salary
EMPLOYEE
Birthdate
Sex
1
Supervisor
N
Supervisee
SUPERVISES
Manager
1
MANAGES
1
WORKS_FOR
N
1
M
WORKS_ON
N
1
CONTROLS
N
DEPARTMENT
PROJECT
Location
Name
Number
Location
Name
Number

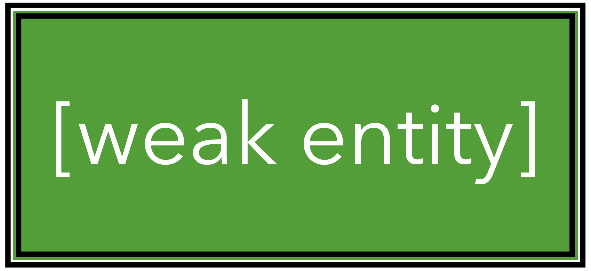

Firstname
Lastname
City
Province
Postcode
Address
Ssn
Name
EMPLOYEE
Salary
Birthdate
Sex
1
Supervisor
N
Supervisee
SUPERVISES
Manager
1
MANAGES
StartDate
1
WORKS_FOR
N
1
M
WORKS_ON
Hour
N
1
CONTROLS
N
DEPARTMENT
Location
Name
Number
PROJECT
Location
Name
Number

Additional Requirement of COMPANY DB

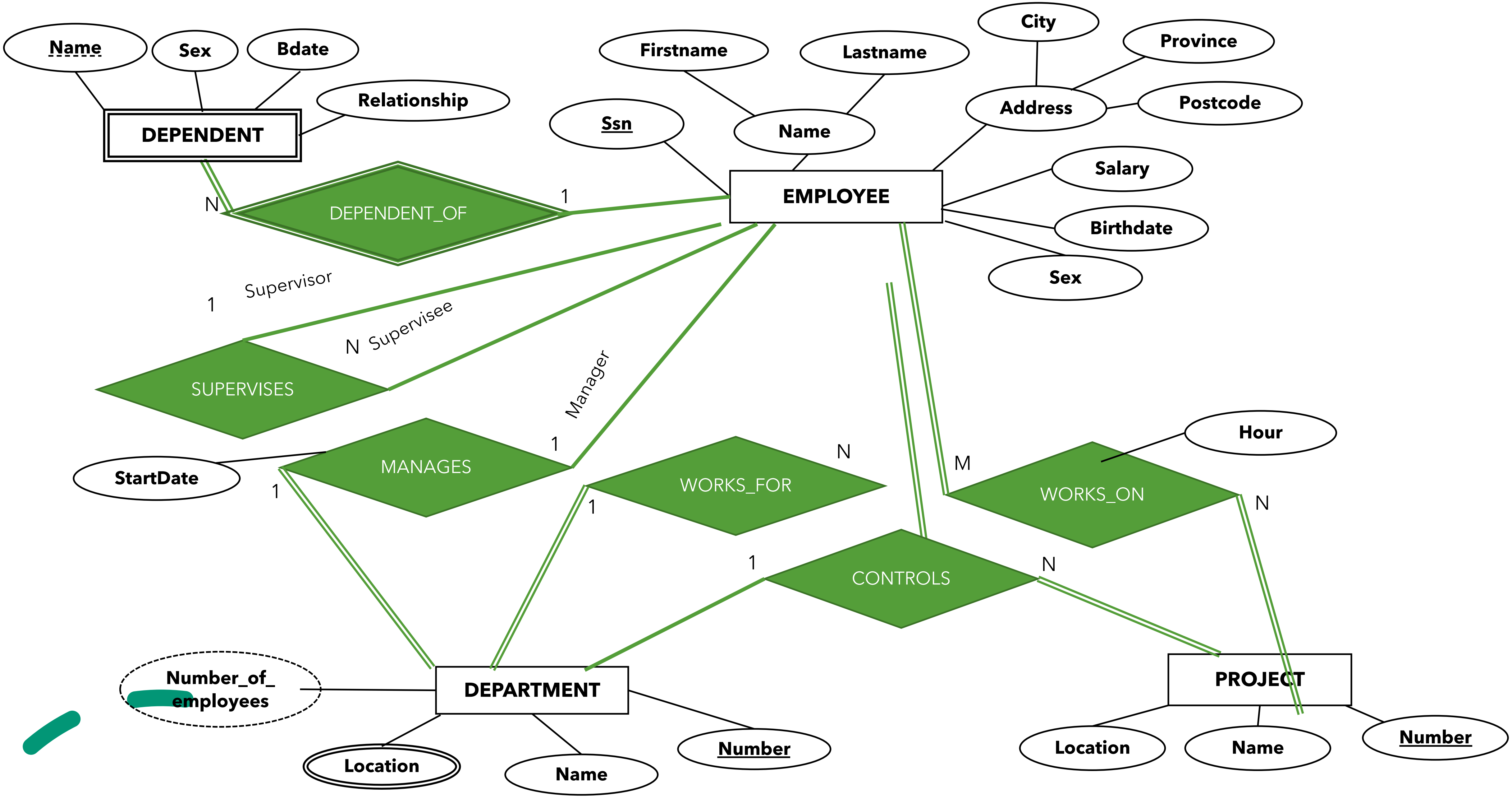
- The database will keep track of the dependents of each employee for insurance purposes, including each dependent’s first name, sex, birth date, and relationship to the employee.

Weak Entity

- An entity that cannot be uniquely identified by its attributes alone
- It must use a foreign key in conjunction with its attributes to create a primary key.

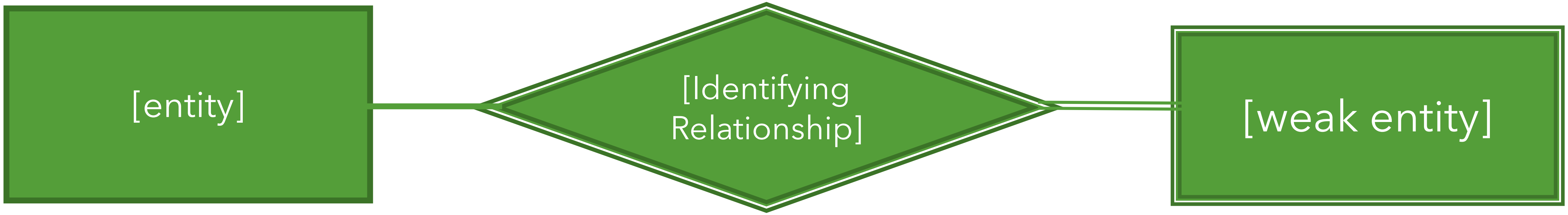


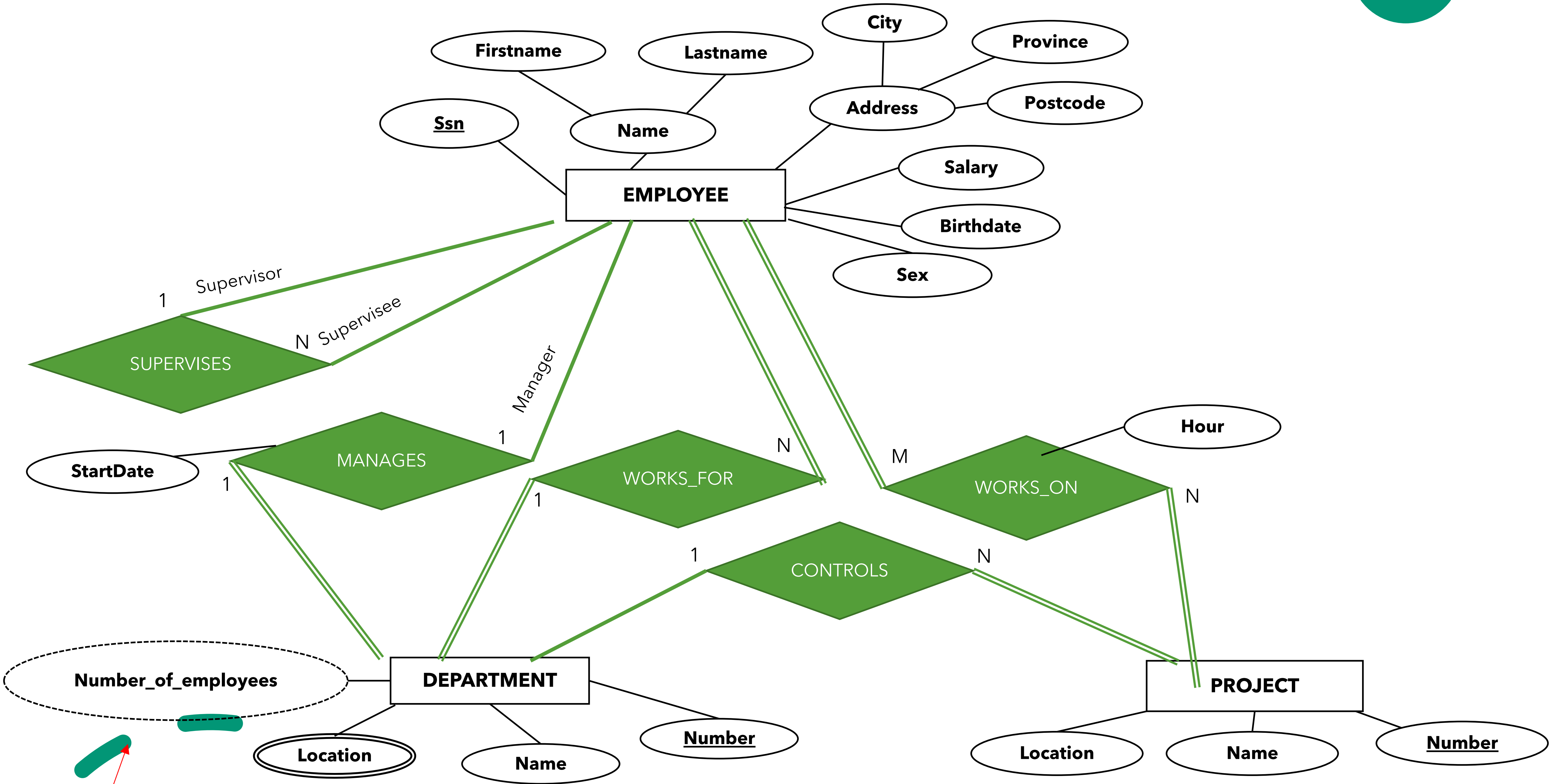
[weak entity]


Name
Sex
Bdate
Relationship
DEPENDENT
N
DEPENDENT_OF
1
Firstname
Lastname
City
Province
Address
Postcode
Ssn
Name
EMPLOYEE
Salary
Birthdate
Sex
1
Supervisor
N
Supervisee
SUPERVISES
Manager
1
MANAGES
StartDate
1
WORKS_FOR
N
1
M
WORKS_ON
Hour
N
1
CONTROLS
N
Number_of_
employees
DEPARTMENT
Location
Name
Number
PROJECT
Location
Name
Number

Identifying Relationship

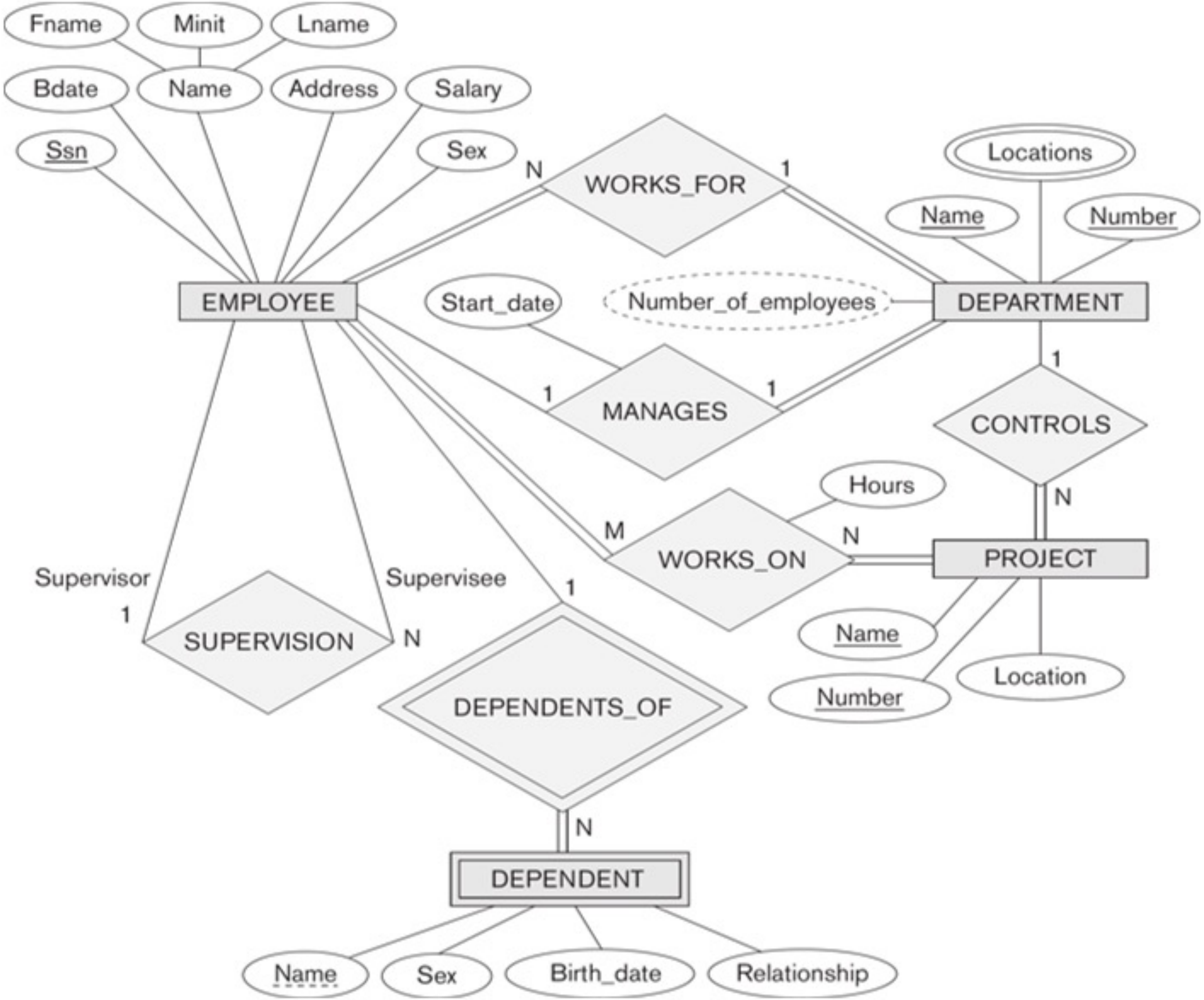
- The relationship that relates the weak entity type to an owner entity type



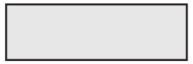
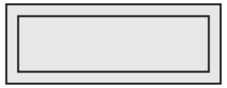
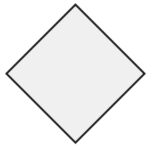
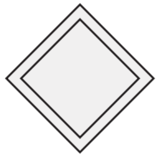
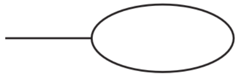
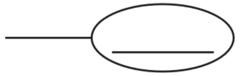
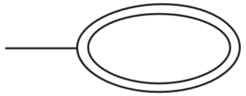


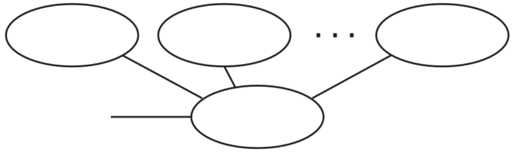
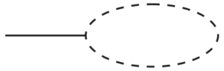
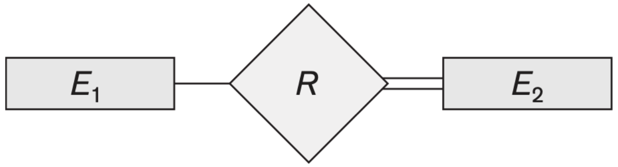
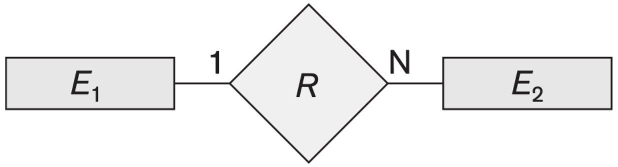
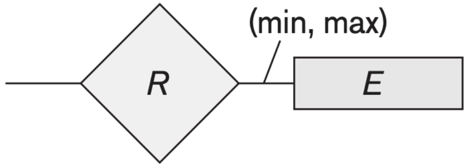
Firstname
Lastname
City
Province
Address
Postcode
Ssn
Name
EMPLOYEE
Salary
Birthdate
Sex
1
Supervisor
N
Supervisee
SUPERVISES
Manager
1
MANAGES
StartDate
1
N
WORKS_FOR
1
M
WORKS_ON
Hour
N
1
CONTROLS
N
Number_of_employees
DEPARTMENT
Location
Name
Number
PROJECT
Location
Name
Number

Derived attribute

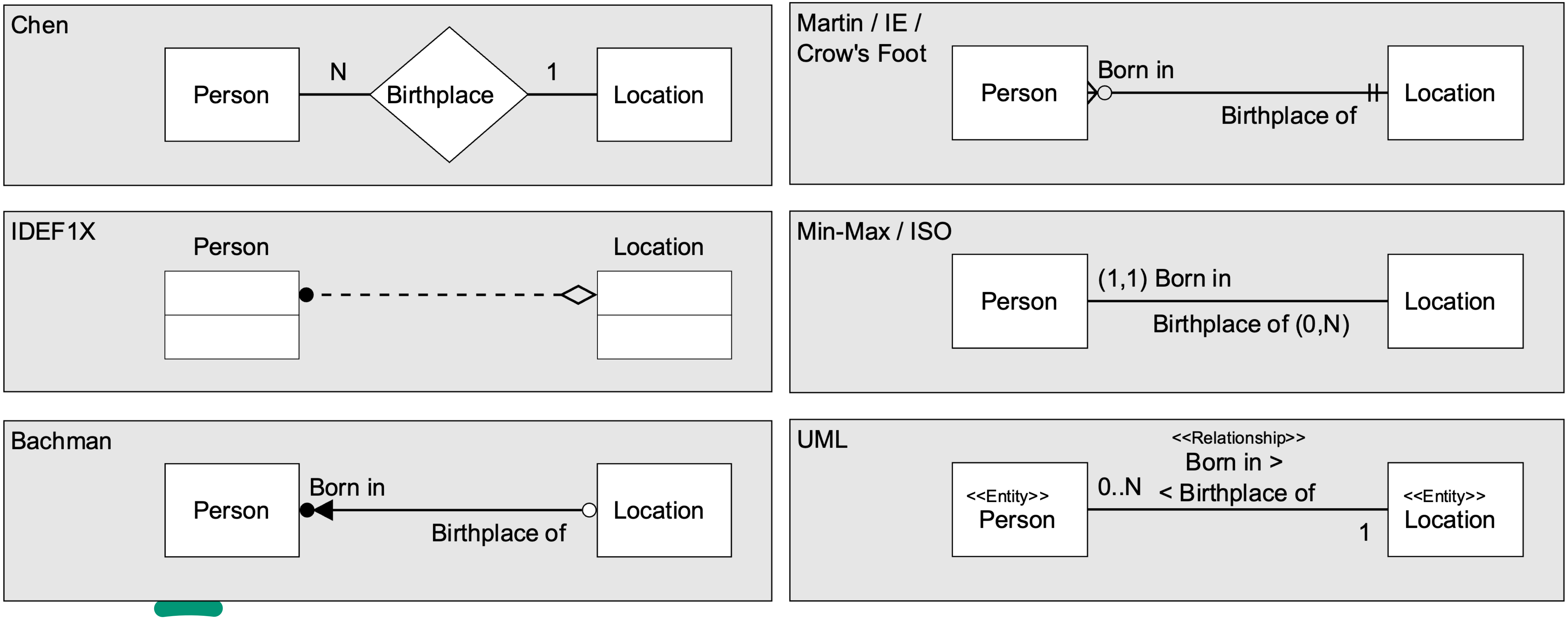

Fname
Minit
Lname
Bdate
Name
Address
Salary
Ssn
Sex
N
WORKS_FOR
1
Locations
Name
Number
EMPLOYEE
Start_date
Number_of_employees
DEPARTMENT
1
1
MANAGES
1
CONTROLS
Hours
N
M
N
WORKS_ON
PROJECT
Supervisor
Supervisee
1
1
SUPERVISION
N
Name
DEPENDENTS_OF
Location
Number
N
DEPENDENT
Name
Sex
Birth_date
Relationship

Summary of ER Diagram

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Indentifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute

Symbol	Meaning
	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 Styles



Chen
Person
N
Birthplace
1
Location

Martin / IE / Crow's Foot
Person
Born in
Birthplace of
Location

IDEF1X
Person
Location

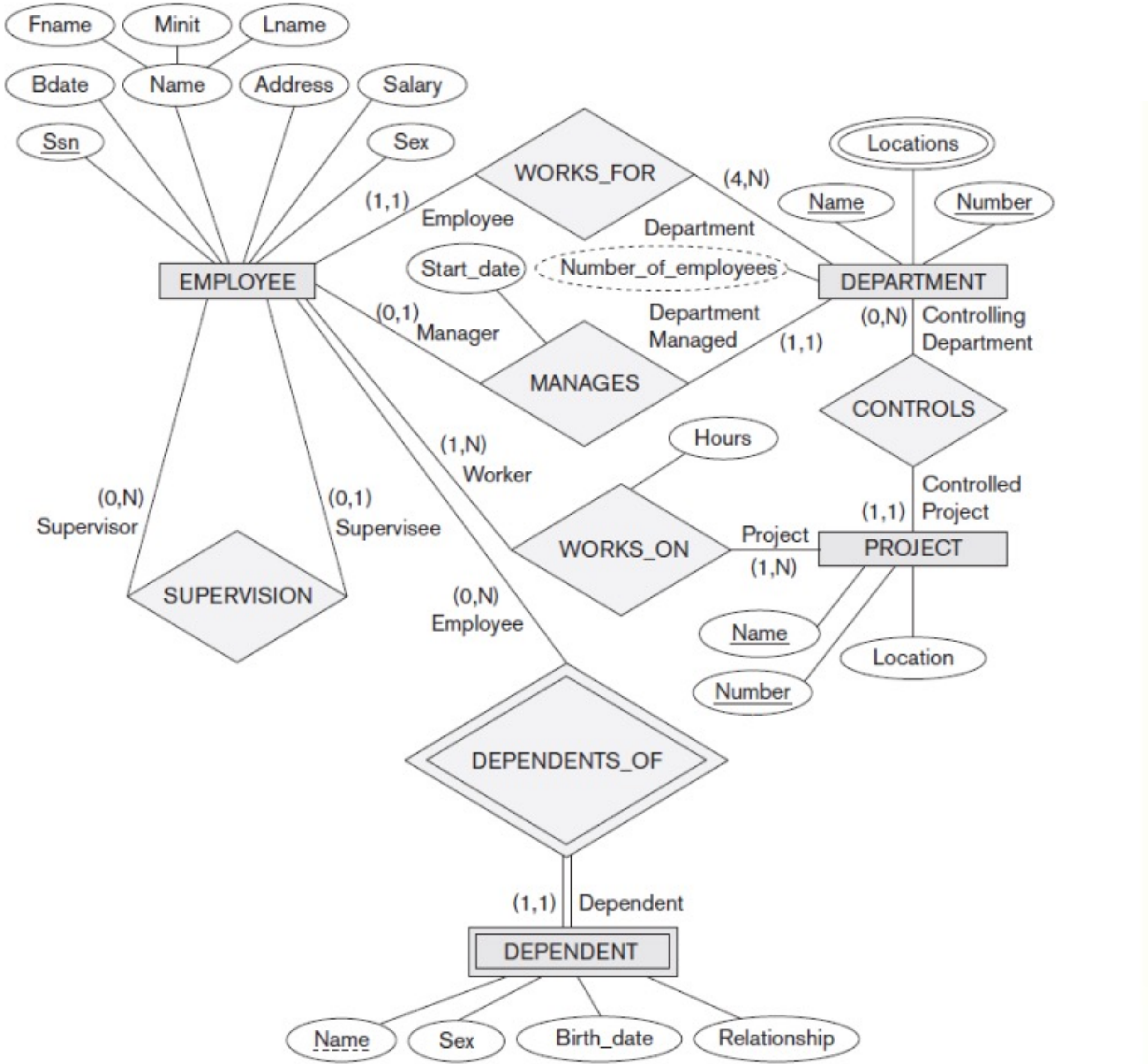
Min-Max / ISO
Person
(1,1) Born in
Birthplace of (0,N)
Location

Bachman
Person
Born in
Birthplace of
Location

UML
<<Relationship>>
Born in >
< Birthplace of
<<Entity>>
Person
0..N
1
<<Entity>>
Location

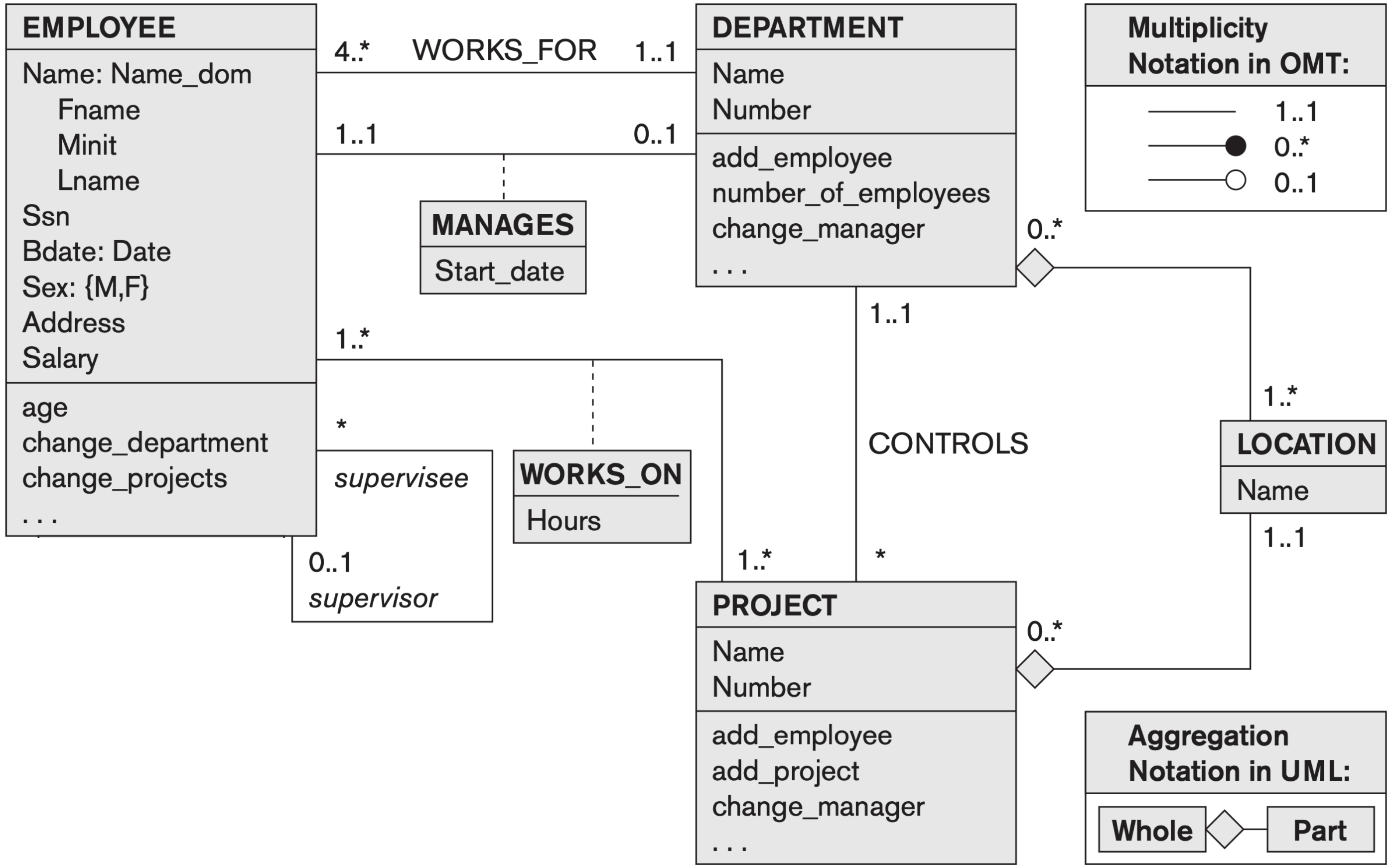
Alternative Notations for ER Diagrams

- Min-Max



Alternative Notations for ER Diagrams

- Unified Modeling Language (UML)


EMPLOYEE
Name: Name_dom
Fname
Minit
Lname
Ssn
Bdate: Date
Sex: {M,F}
Address
Salary
age
change_department
change_projects
. . .
4..*
WORKS_FOR
1..1
1..1
0..1
MANAGES
Start_date
DEPARTMENT
Name
Number
add_employee
number_of_employees
change_manager
. . .
Multiplicity
Notation in OMT:
1..1
0..*
0..1
0..*
1..*
*
supervisee
WORKS_ON
Hours
0..1
supervisor
1..1
CONTROLS
1..*
LOCATION
Name
1..1
1..*
*
PROJECT
Name
Number
add_employee
add_project
change_manager
. . .
0..*
Aggregation
Notation in UML:
Whole
Part