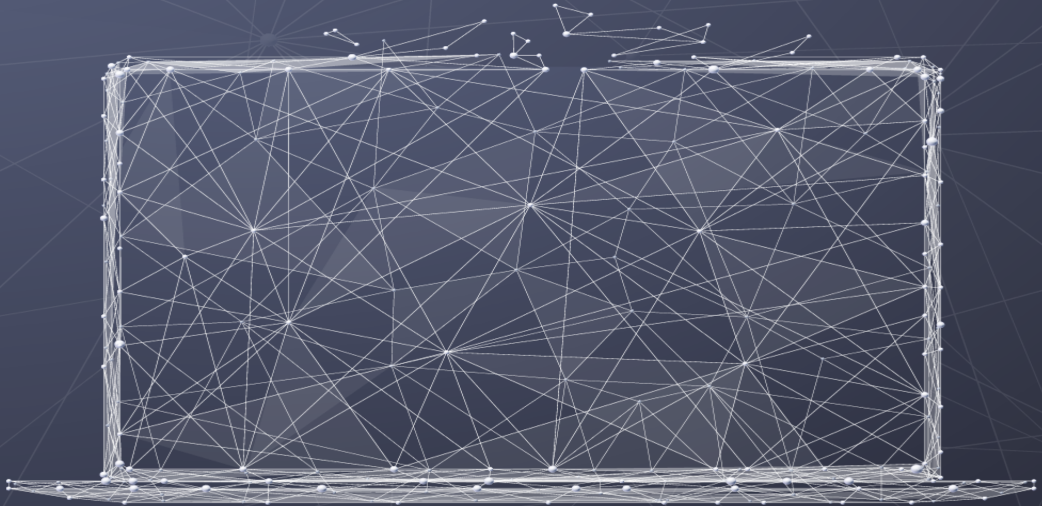
Data Science
“Data Engineering II”

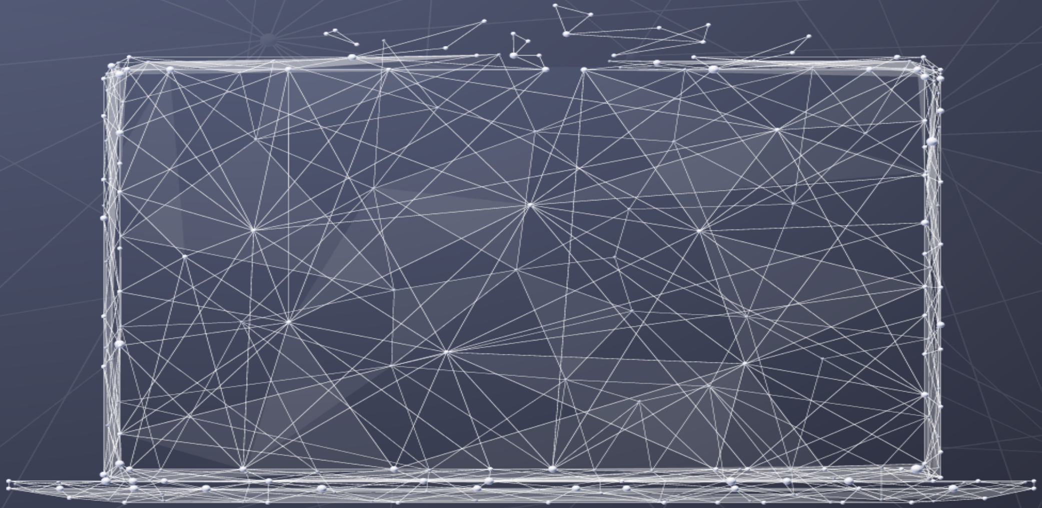
Data Engineering II



PURDUE UNIVERSITY | College of Science

Data Science
“Data Engineering II”

Entity Relationship Modeling



PURDUE UNIVERSITY® | College of Science

Entity-Relationship Modeling

Outcome

In this lecture, we will introduce the Entity-Relationship modeling tool used to model an enterprise database.

Database Design

Conceptual Design

Logical Design

Physical Design

Creating an Entity Relationship Diagram (ERD) which describe the real world enterprise entities, attributes and the relationships among them.

Transforming ERD to relational model: tables, keys (constraints), etc.

Creating the database and other supporting structures based on a specific DBMS. E.g. Mysql

Conceptual Design

Entity Relationship (ER) Models

- What aspects of the enterprise will be modeled?
- What are the key concepts that need to be represented?
    - What entities do we care about?
    - What relationships do they have with each other?
- What constraints does the application impose?
    - Data type and Integrity constraints
    - Policies to be enforced
- Often done pictorially using an E-R Diagram

Sample Application

Company Database

- 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.
  A department may have several locations.

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

Example adapted from “Fundamentals of Database Systems” by Elmasri.

Sample Application

Company Database

- 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 the employee

ER Model Concepts

Entities

An entity: is a real-world “object” distinguishable from other objects which is represented in the database. E.g., employee, department, course, account, …

Entity Set: A collection of similar entities. E.g., all employees.
All have the same types of attributes

Attributes are properties used to describe an entity.
A specific entity will have a value for each of its attributes.

Each attribute has a **data type**, or **domain** that defines allowable values for that attribute. E.g., integer, string, subrange, …

ER Model Concepts

Types of Attributes

Simple

Each entity has a single atomic value for the attribute. E.g., SSN or Gender.

Composite

The attribute may be composed of several components. E.g., Address, name

Composition may form a hierarchy where some components are themselves composite.

Multi-valued

An entity may have multiple values for that attribute. E.g., Color of a CAR or PreviousDegrees of a STUDENT.

ER Model Concepts

Key Attributes

- In any valid instance of a database, each entity set should not have duplicates — I.e., no two rows have the same values for all attributes.
- However, for many applications we require uniqueness over other groups of attributes — e.g., SSN or {SSN, StartDate}
- Any combination of attributes that must be unique in any valid instance is called a **key**

ER Model Concepts

Key Attribute

- Each entity must have at least one key attribute

- A key attribute may be composite
  - VehicleTagNumber is a key of the CAR entity type with components (Number, State).

- An entity type may have more than one key

  The CAR entity type may have two keys:

  VehicleIdentificationNumber (popularly called VIN)

  VehicleTagNumber (Number, State), aka license plate number.
- Each key is underlined

PURDUE UNIVERSITY | College of Science

ER Model Concepts

Displaying an Entity

- In ER diagrams, an entity is displayed as a rectangular box
- Attributes are displayed in ovals
- Each attribute is connected to its entity type
- Each key attribute is underlined
- Multivalued attributes are displayed in double ovals

Company Database

The DEPARTMENT Entity

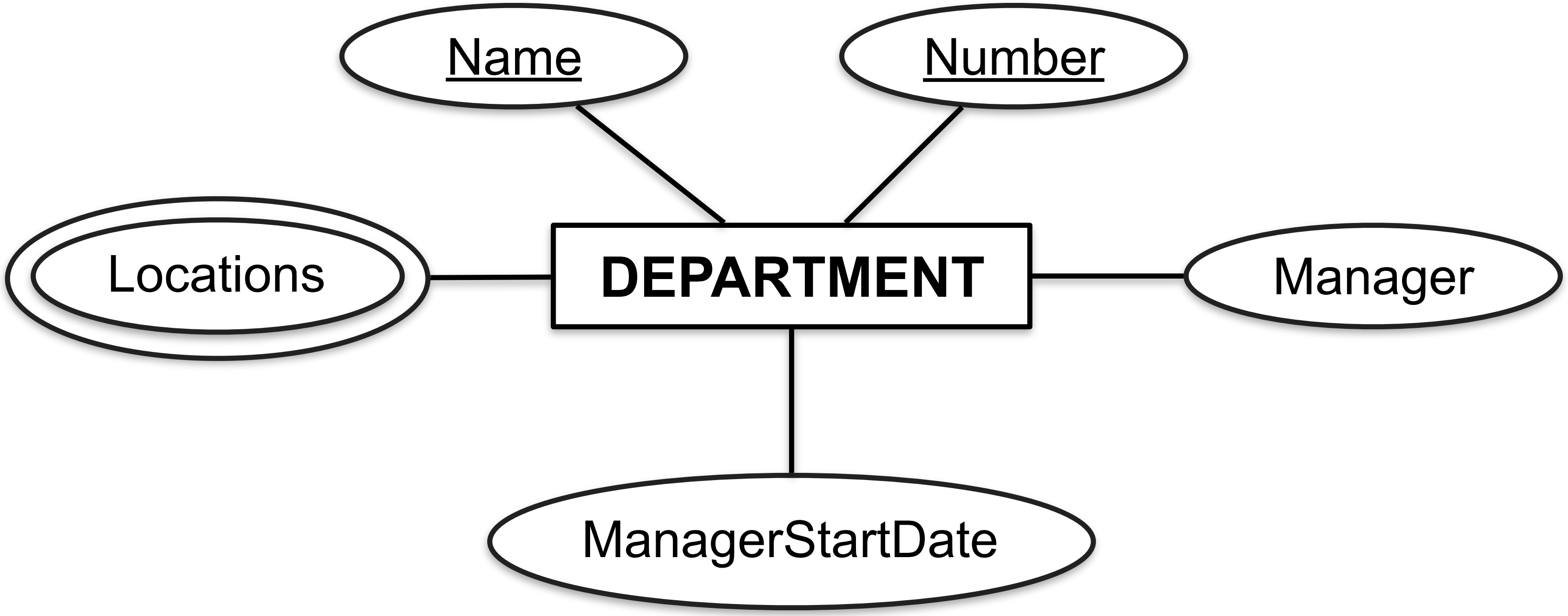


*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

Company Database

The PROJECT Entity

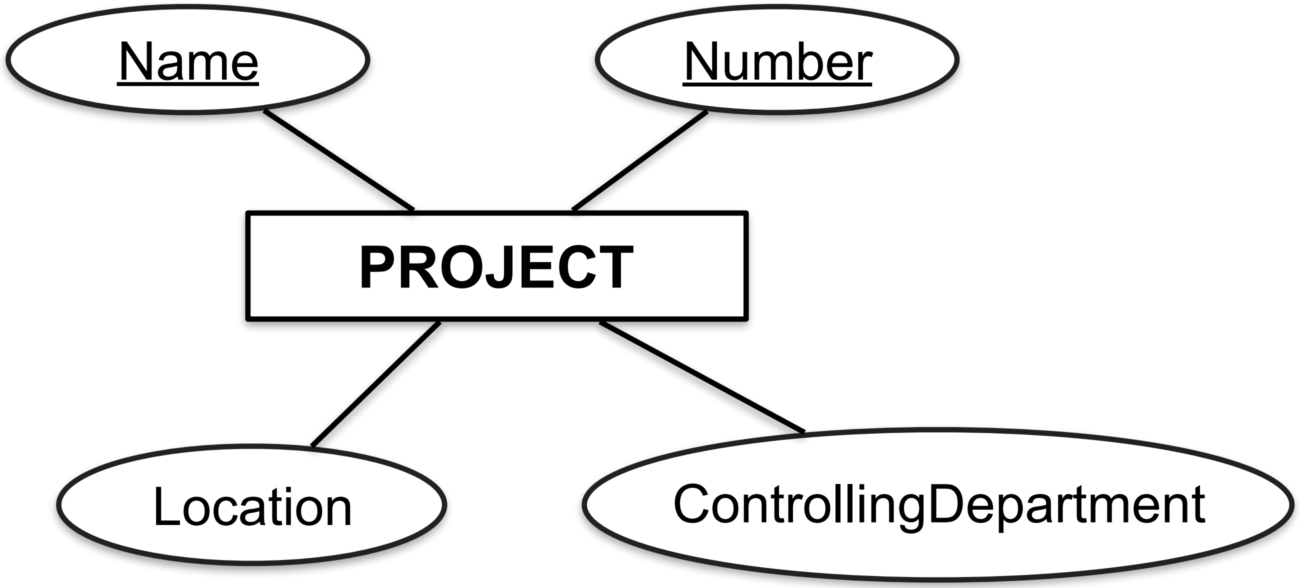
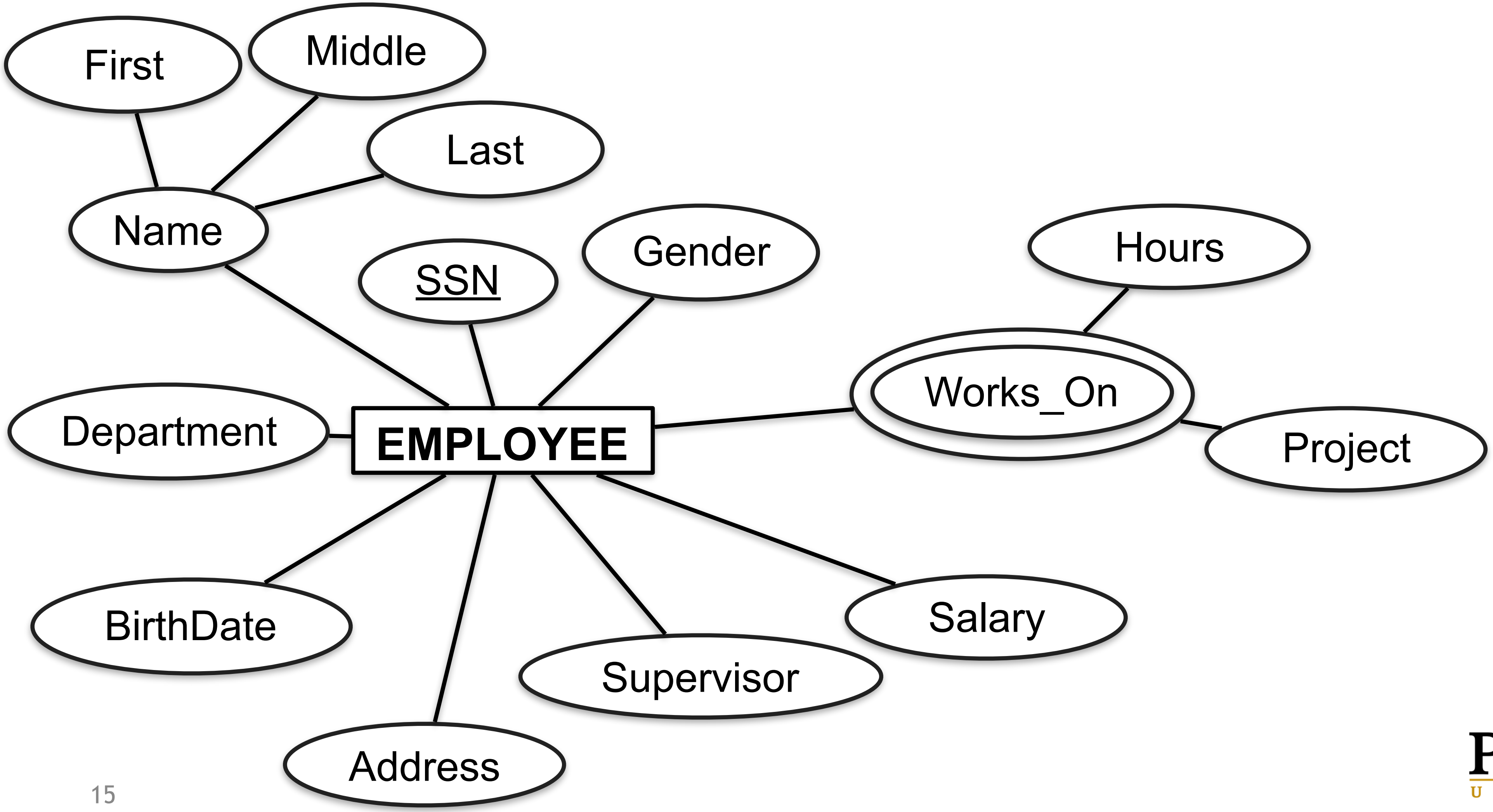


*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

EMPLOYEE entity



PURDUE UNIVERSITY College of Science

*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

Company Database

The DEPENDENT Entity

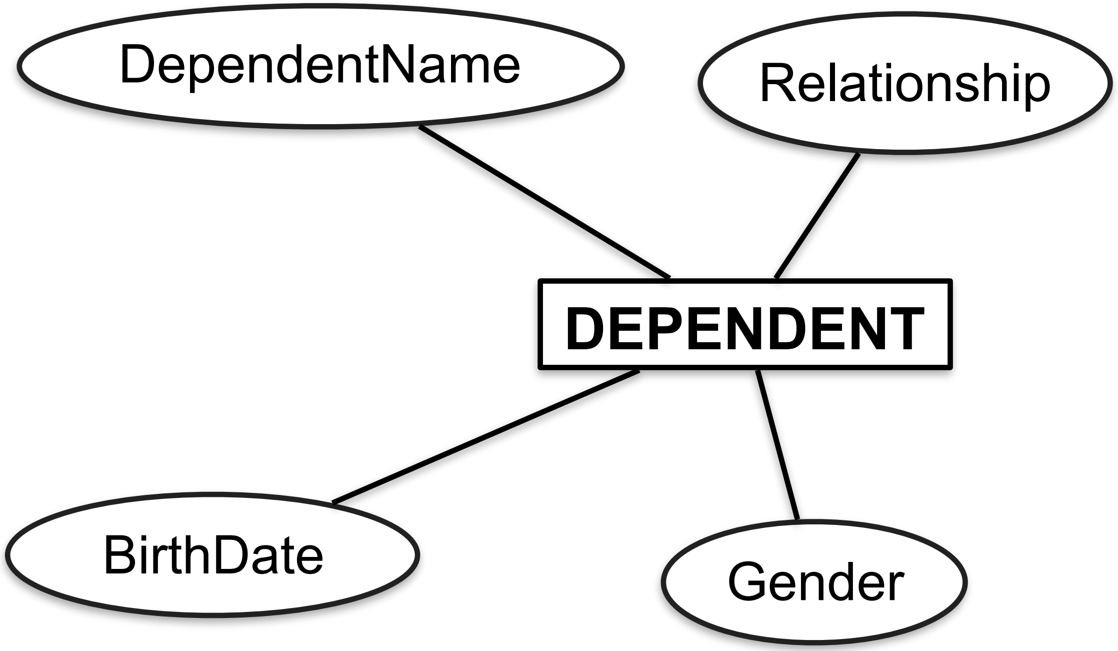


*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

ER Model Concepts

Relationships

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

  E.g., DEPARTMENT Software CONTROLS the Server PROJECT, or EMPLOYEE Franklin Wong MANAGES the Research DEPARTMENT.

- Relationships of the same type are grouped into a **Relationship Type**.

  For example, the WORKS_ON relationship type in which EMPLOYEEs and PROJECTs participate, or the MANAGES relationship type in which EMPLOYEEs and DEPARTMENTs participate.

- A Relationship Type can connect two or more Entity types together

  Binary, tertiary, etc.

Sample Use of Relationships

Company Database Relationships

- By examining the requirements, six relationship types are identified

    All are binary relationships(degree 2)

- Listed below with their participating entity types:

    WORKS_FOR (between EMPLOYEE, DEPARTMENT)

    MANAGES (also between EMPLOYEE, DEPARTMENT)

    CONTROLS (between DEPARTMENT, PROJECT)

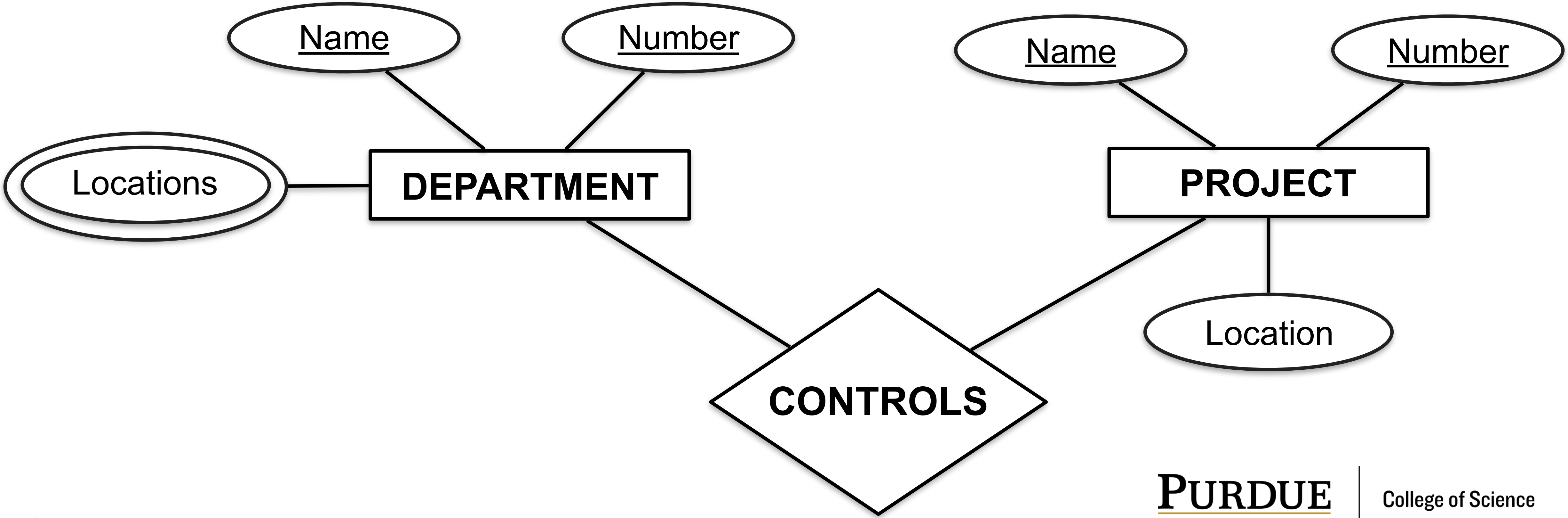
    WORKS_ON (between EMPLOYEE, PROJECT)

    SUPERVISION (between EMPLOYEE (as subordinate), EMPLOYEE (as supervisor))

    DEPENDENTS_OF (between EMPLOYEE, DEPENDENT)

Relationships in ER diagrams

Displaying the CONTROLS Relationship



EMPLOYEE entity

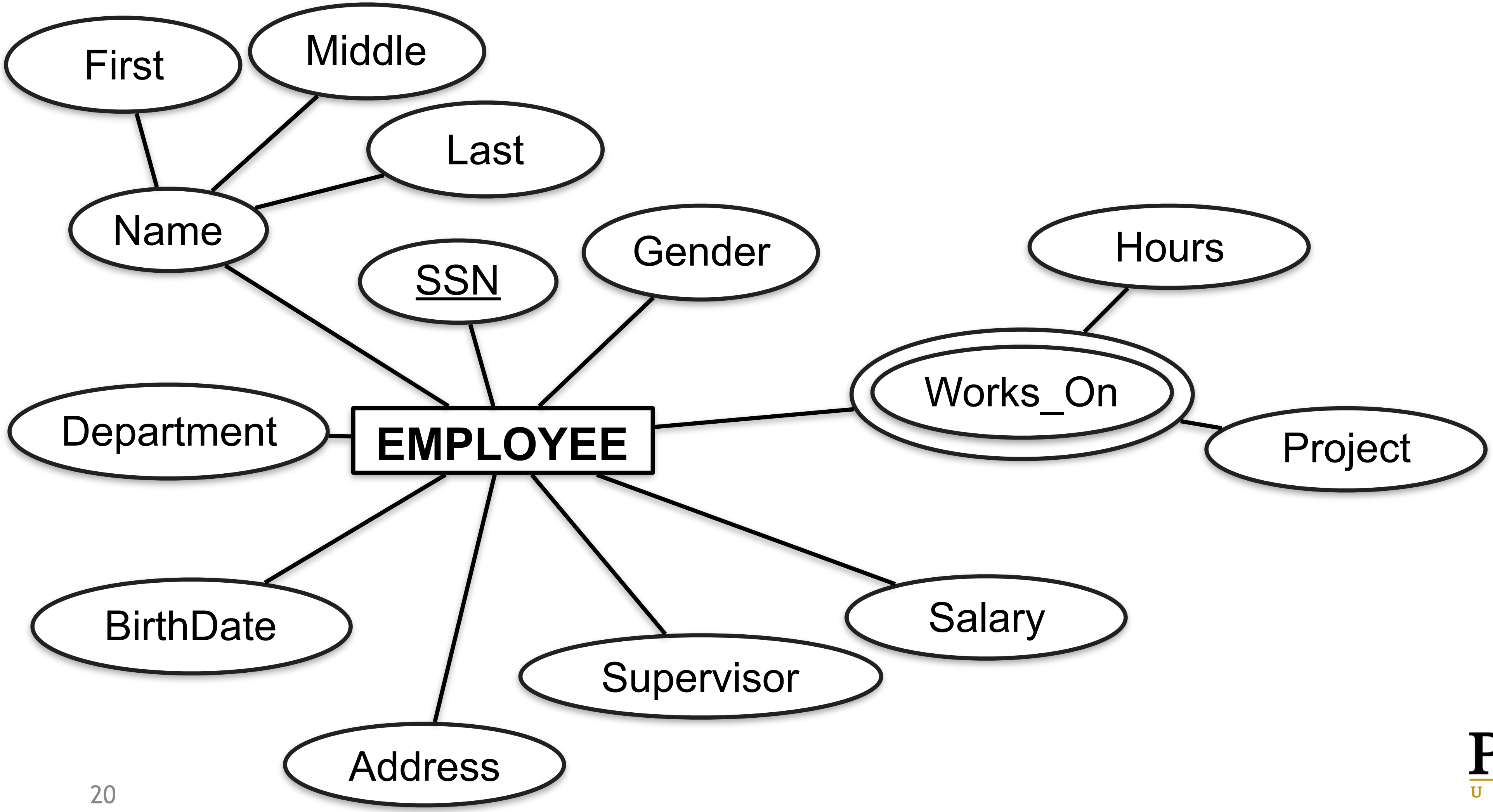
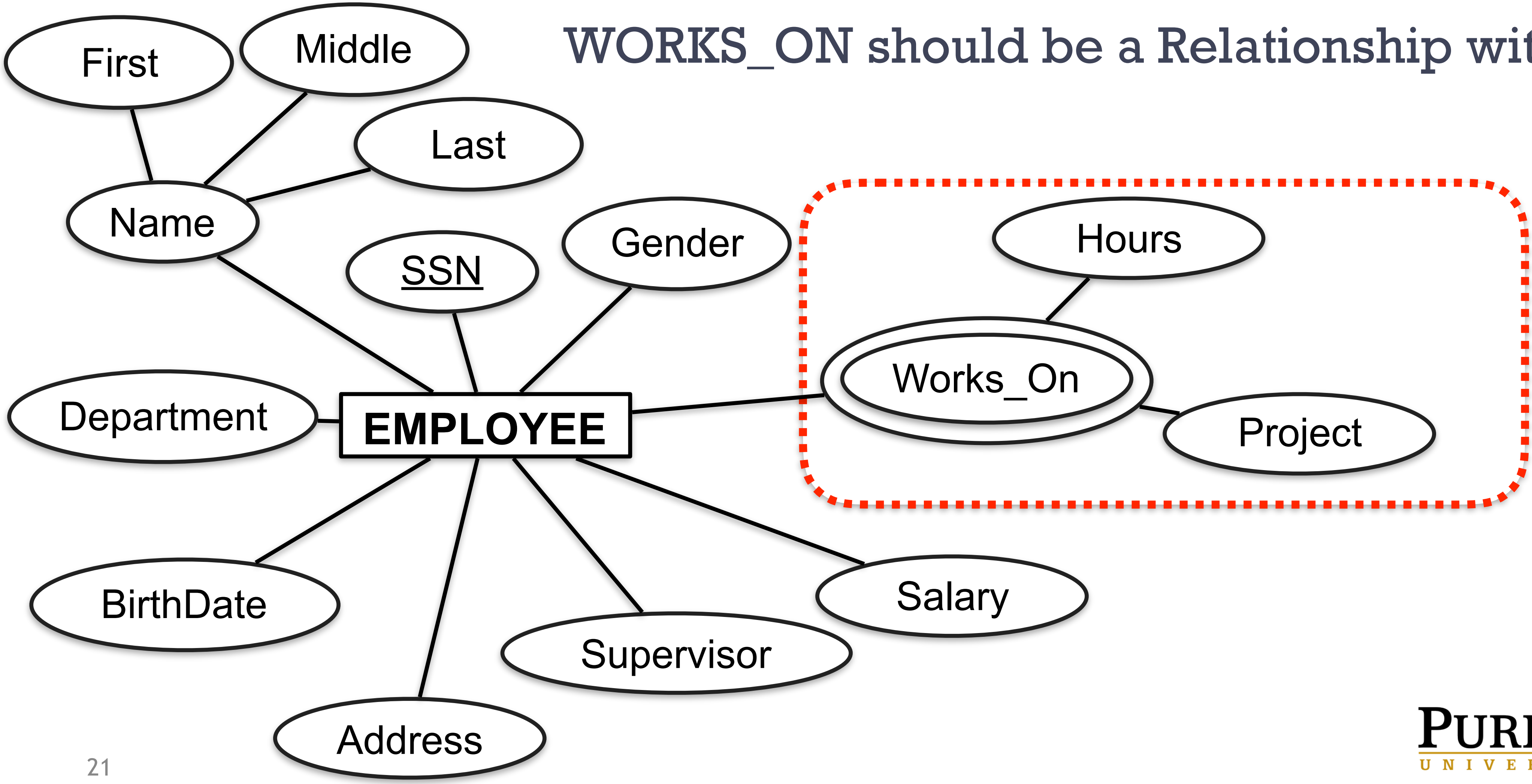


*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

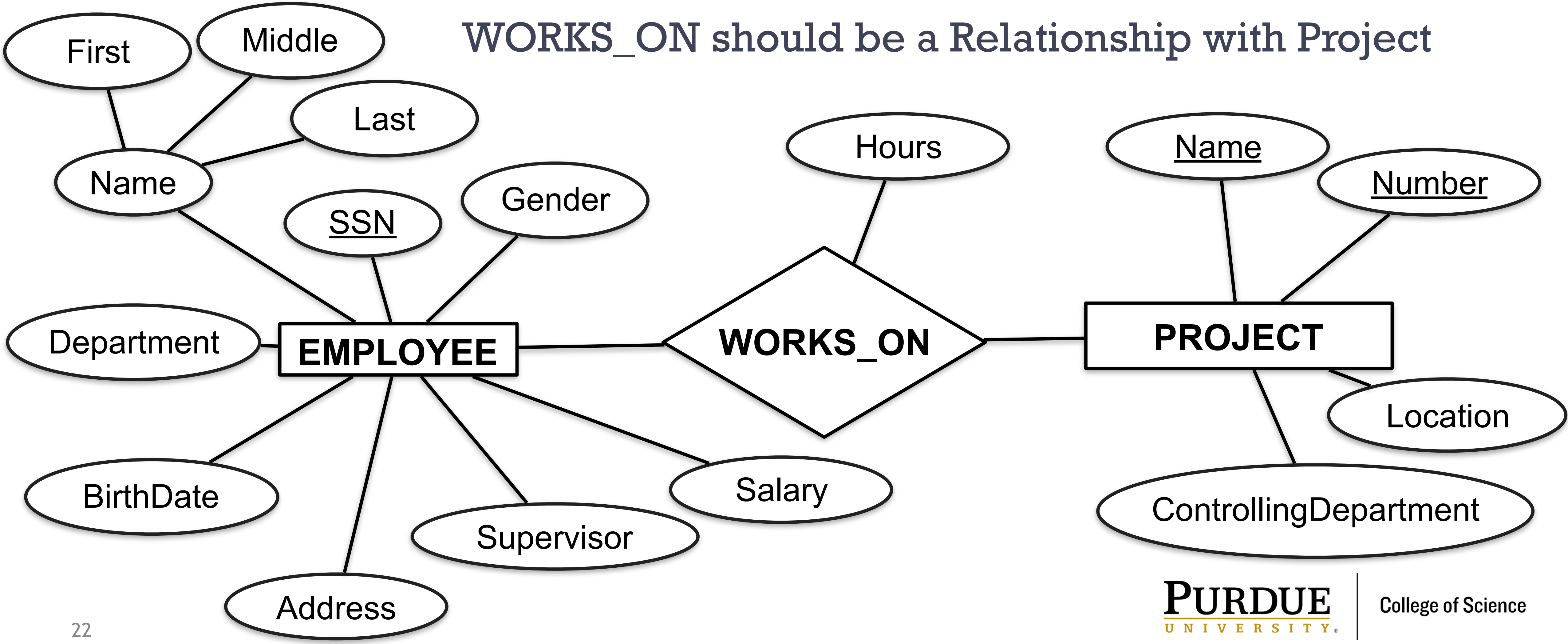
Refining Our Model

WORKS_ON should be a Relationship with Project



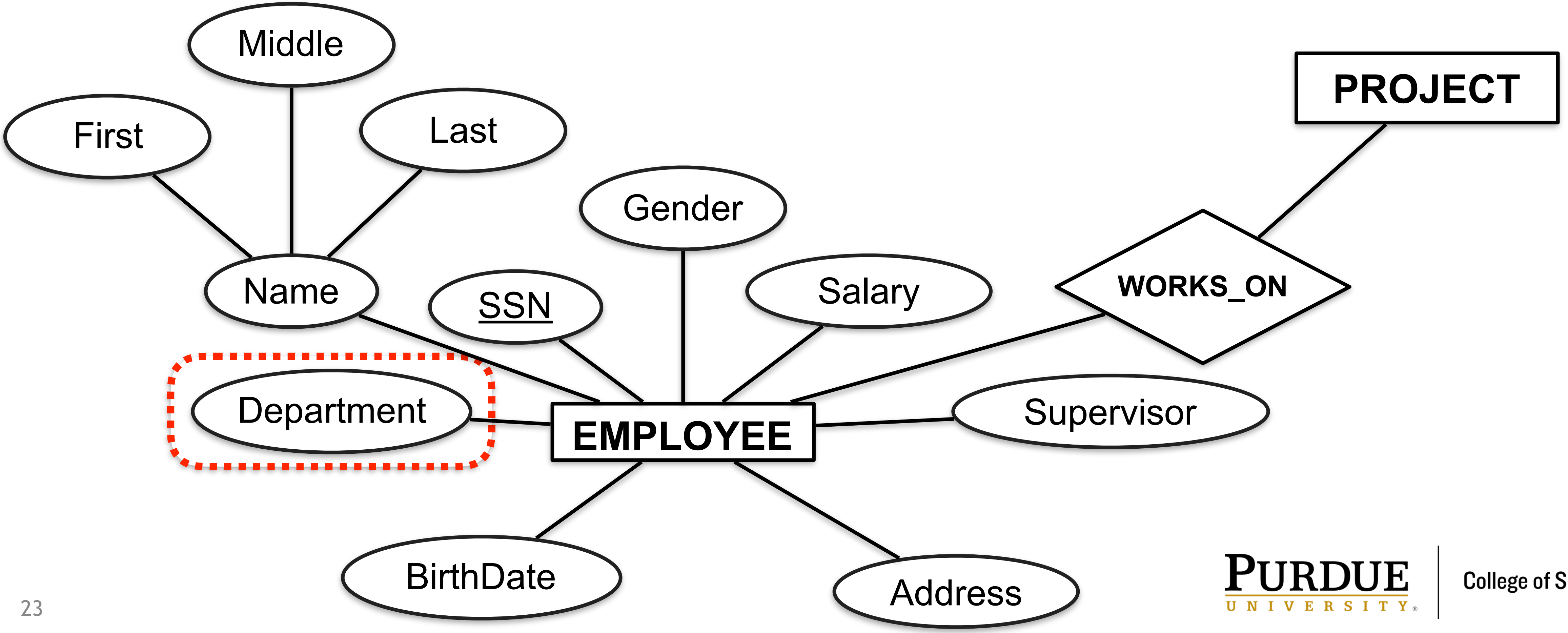
Refining Our Model

WORKS_ON should be a Relationship with Project



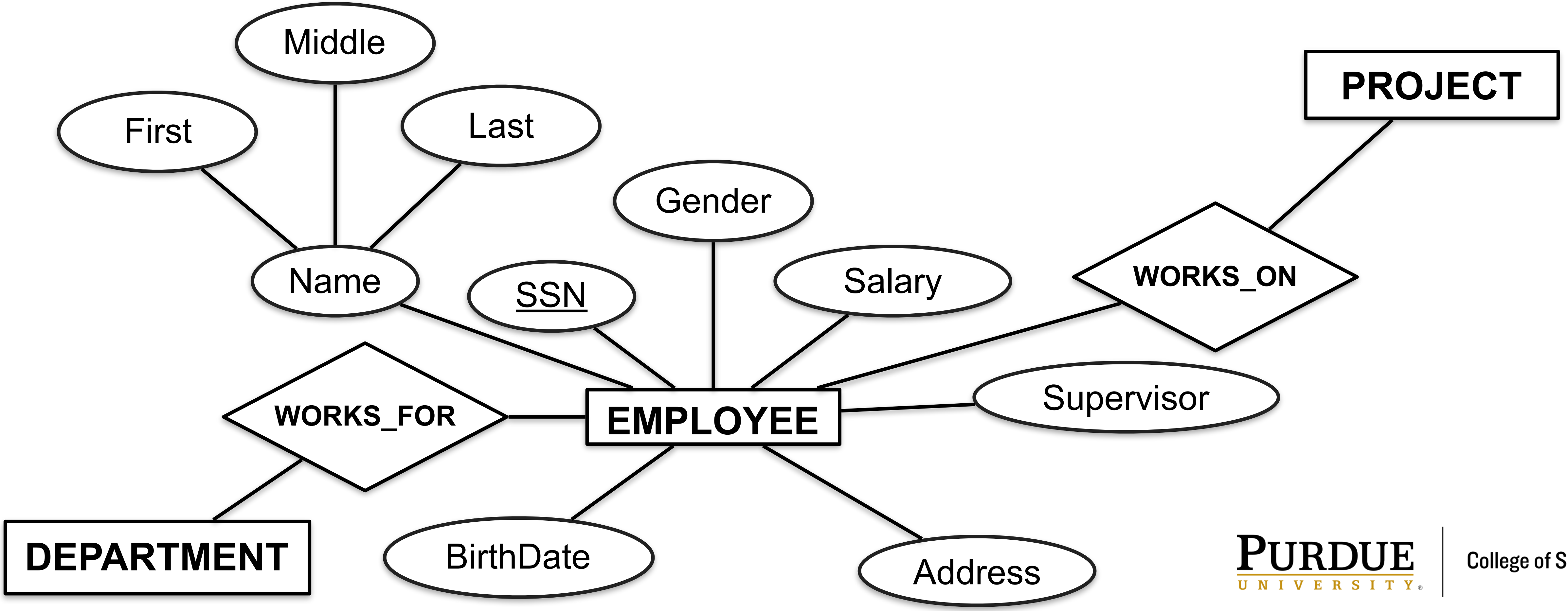
Refining Our Model

Department should be a Relationship with DEPARTMENT



Refining Our Model

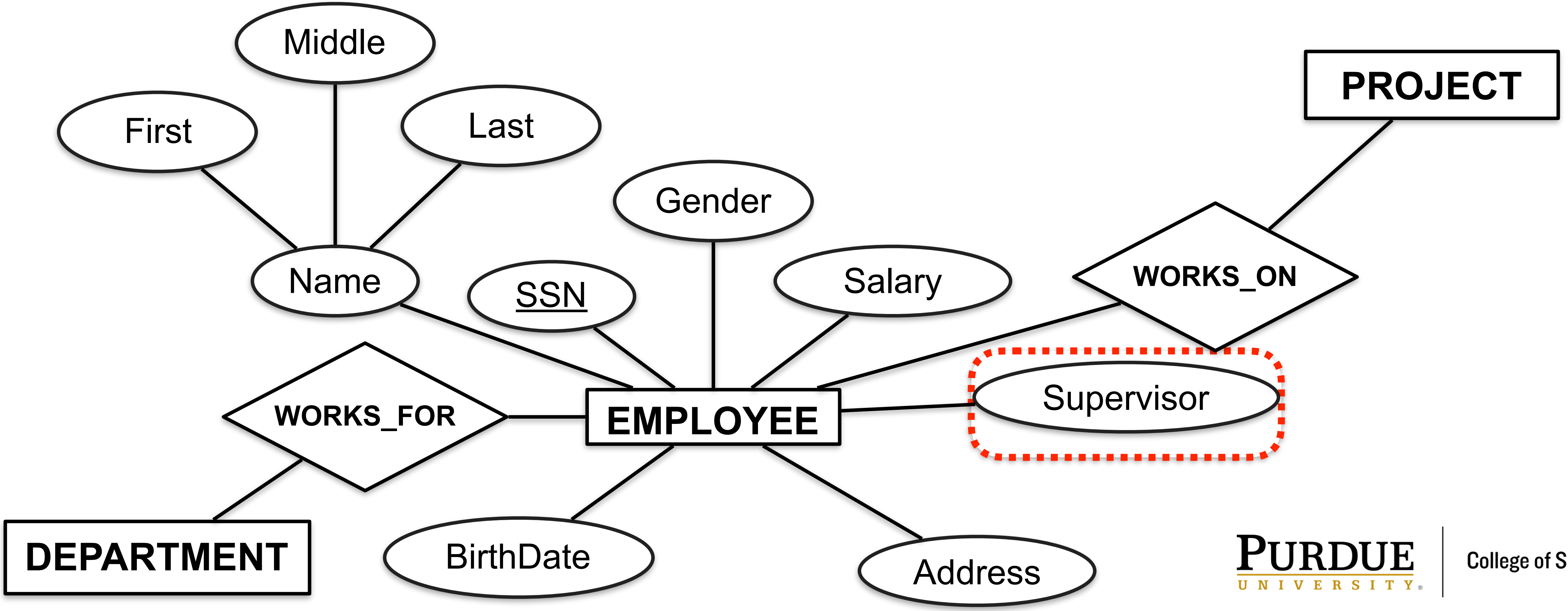
Department should be a Relationship with DEPARTMENT



Purdue University | College of Science

Refining Our Model

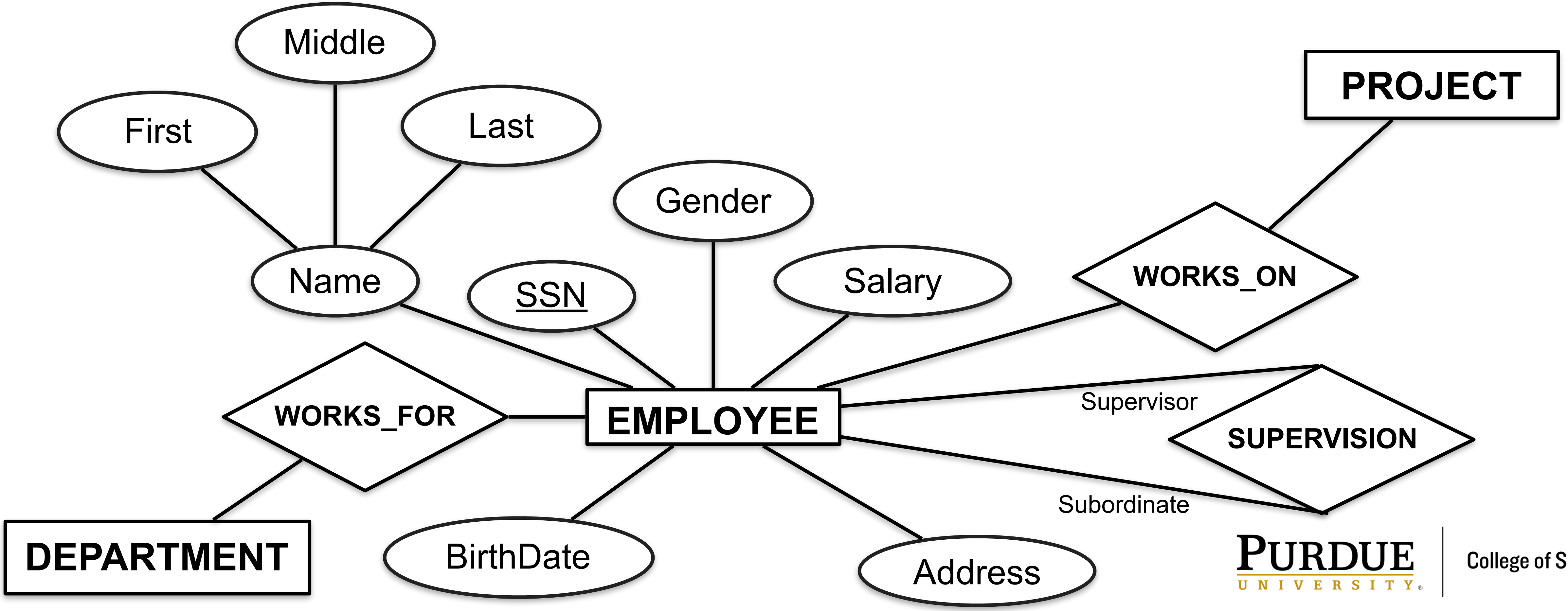
Supervisor should be a relationship with EMPLOYEE



PURDUE UNIVERSITY | College of Science

Refining Our Model

Supervisor should be a relationship with EMPLOYEE



PURDUE UNIVERSITY | College of Science

Relationship Constraints

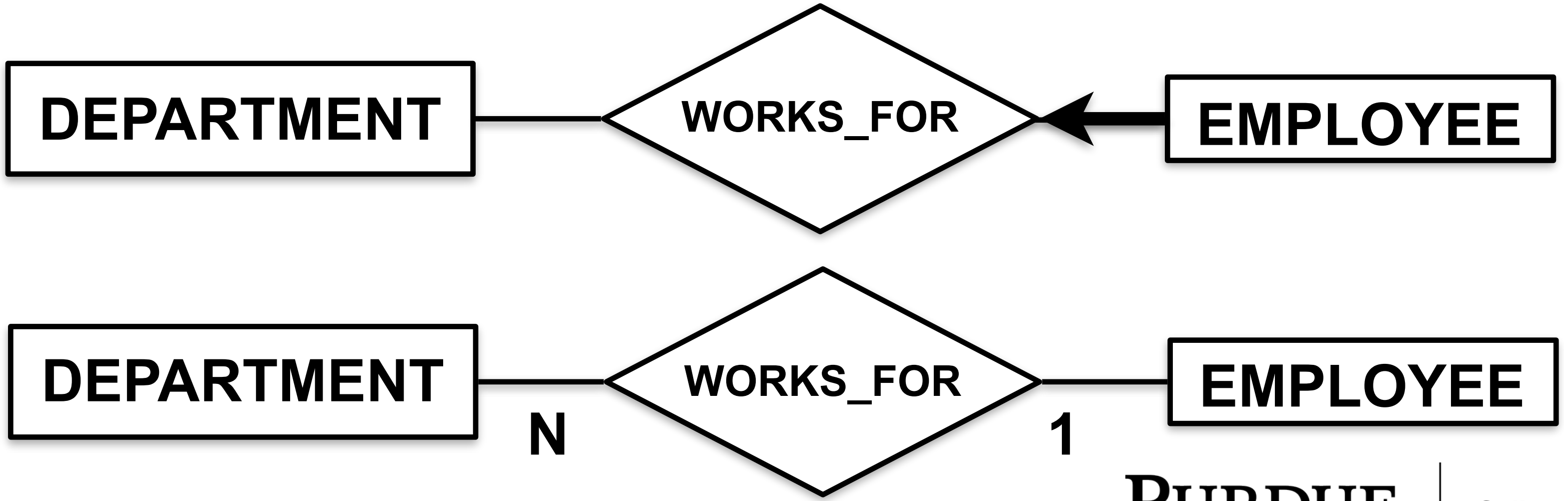
Cardinality and Participation

- For each relationship, it is possible to add constraints for the participating entities
- Driven by application needs
  - In how many relationships can an entity participate?
    - Only once or multiple?
    - WORKS_FOR versus WORKS_ON
  - Must every entity participate in some relationship?
    - WORKS_FOR versus CONTROLS

Relationship Constraints

WORKS_FOR relationship

- Our application may require that every EMPLOYEE should work for exactly one DEPARTMENT
- A DEPARTMENT can have any number of EMPLOYEES working for it



ER Model Concepts

Weak Entities

- An entity that is only of interest due to its relationship with another entity
- 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

ER Model Concepts

Weak Entities

- Example:
  - DEPENDENT is a weak entity type
  - A DEPENDENT entity is identified by the dependent's first name, and the specific EMPLOYEE to whom the dependent is related
  - Name of DEPENDENT is the partial key
  - EMPLOYEE is its identifying entity type via the identifying relationship type DEPENDENT_OF

Weak Entity Example

DEPENDENTS is a Weak Entity related to EMPLOYEE

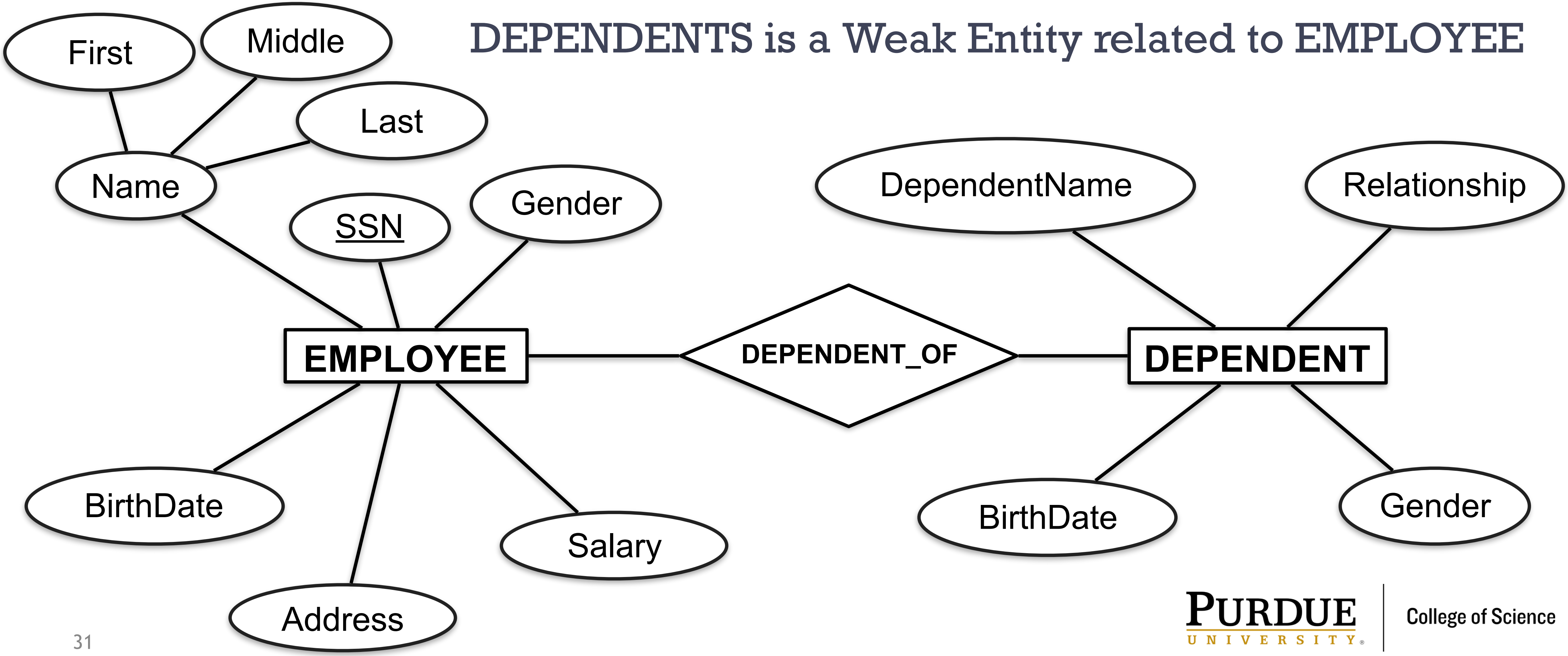


*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

Weak Entity Example

DEPENDENTS is a Weak Entity related to EMPLOYEE

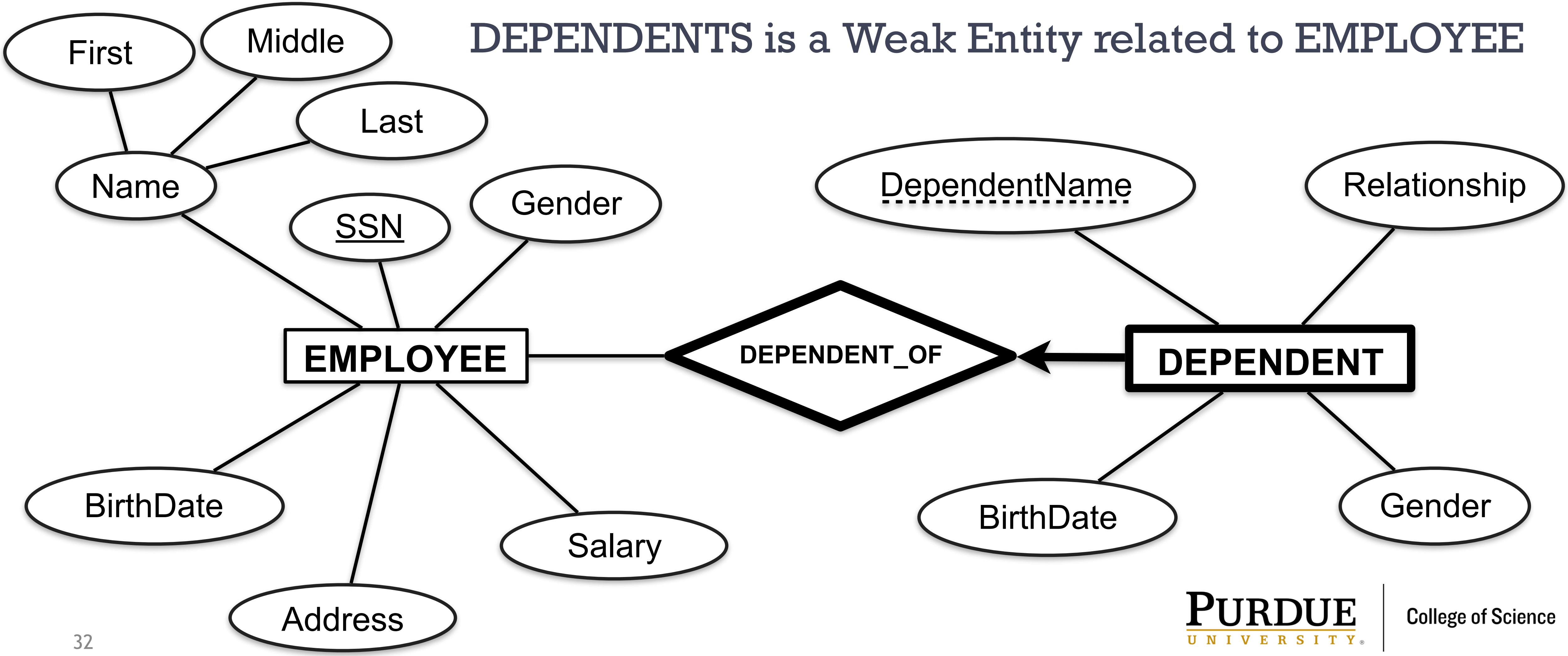
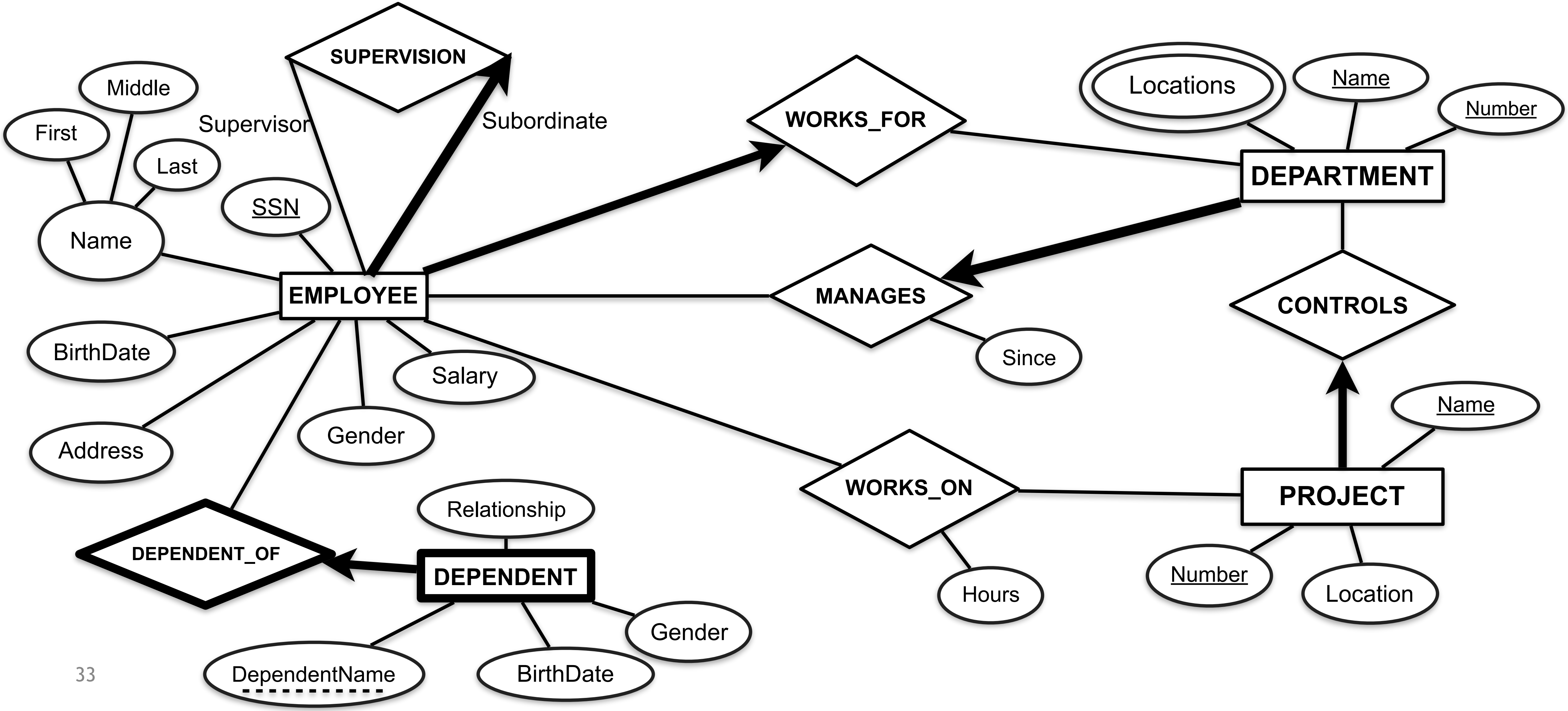


*Figure adapted from “Fundamentals of Database Systems” by Elmasri.

Sample Database ER diagram



ER diagram Design Choices

Many Subjective Options

- Entity versus Attribute?
- Depends:
  - How to handle multiple or composite attributes?
  - Addresses?
    - Single (attribute) or multiple (entity)?
      - Do we care about the city etc.? Yes — entity.
  - Employee working for the same department at different times?

Summary of ER diagrams

Conceptual Database Design

- Follows requirements gathering from user
- Map to Entities and Relationships, associated attributes, domains, and some constraints:
  - Membership and participation constraints
  - Key constraints
- Lot of subjective decisions based upon user specifications
- Aim is to capture as much of the user specifications as possible

Course Projects Database

Flight Dataset

- The database keeps track of airports. Each airport has an ID and a location.
- The database also keeps track of airline carriers. Each carrier has an ID and a name
- It keeps track of flights:
  origin/destination airport, flight time, time of departure/arrival, delays, flight number, carrier, tail number, cancellation(if any), reason of cancellation (if any), distance covered, delay (if any) and reason of delay.

ER diagram for Flight Database

Flight Dataset

- The `airportID` can be used as the key for AIRPORT table.
- The `carrierID` can be used as the key for CARRIER table.
- For the FLIGHT table: The following options *cannot* be used as primary keys:
  - {`carrier, tail number`}: The same plane can be used for different trips.
  - {`carrier, tail number, flight number`}: The same plane can be used for the same flight number on different days.
  - {`carrier, tail number, flight number, date`}: The same plane can used for the same trip in the same day multiple times.

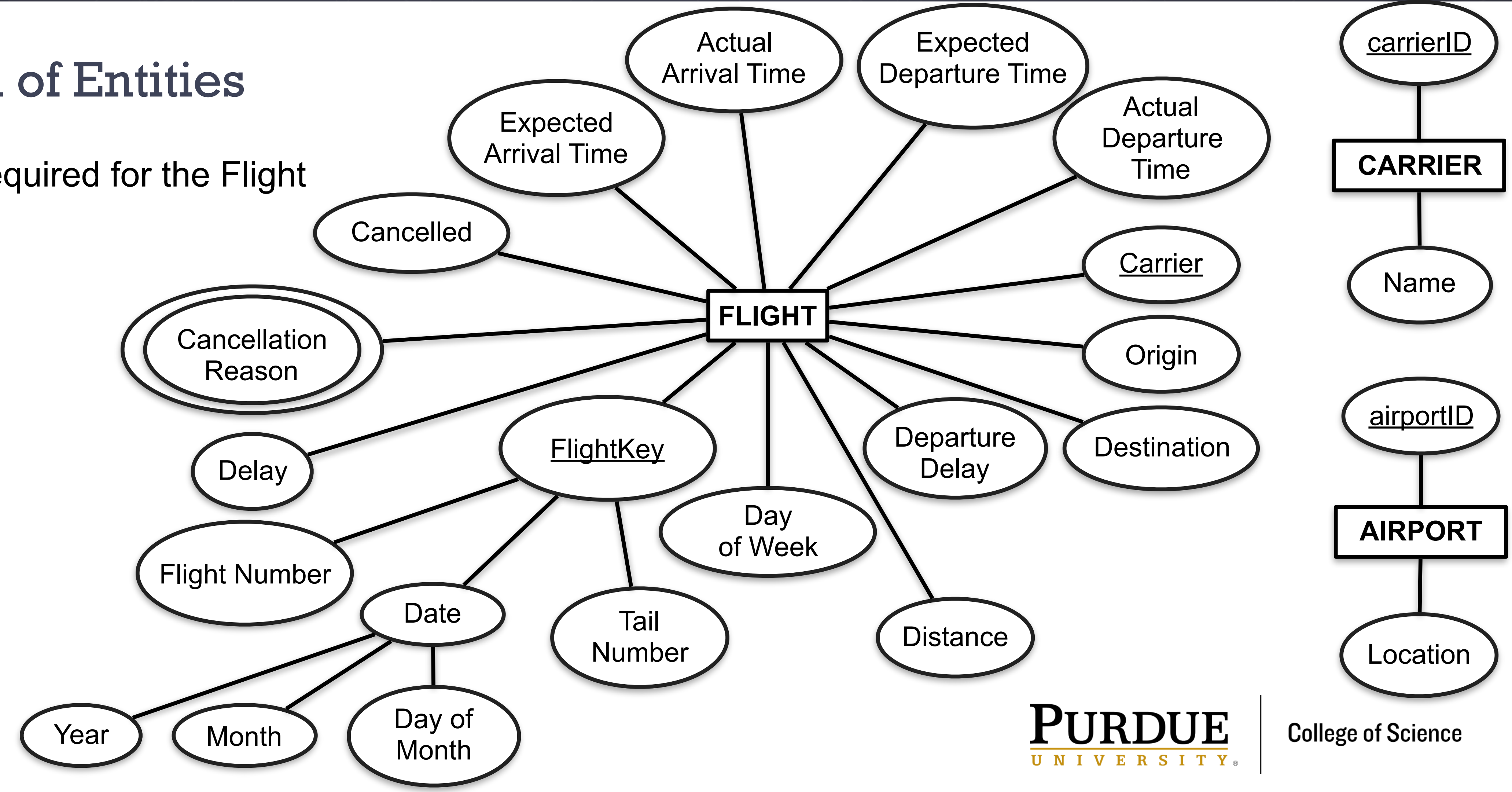
- {`carrier, tail number, flight number, date, expected departure time`} can be used as a key.

ER diagram for Flight Database

Initial Design of Entities

The initial entities required for the Flight database:

Flight
Carrier
Airport



PURDUE UNIVERSITY® | College of Science

Flight Database

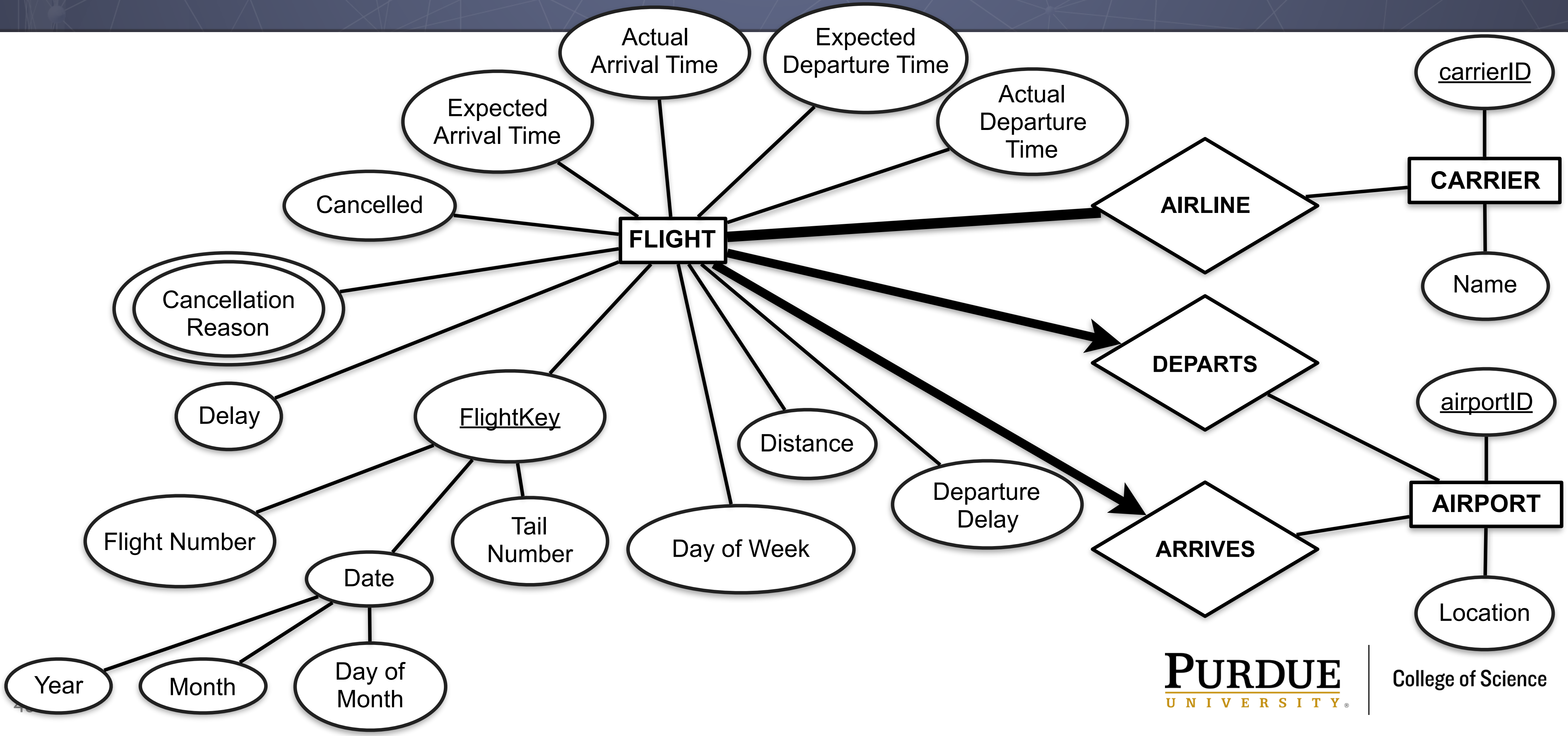
Relationships

- ORIGIN_AIRPORT between (Flight, Airport).
  - It's N:1 relationship. I.e., an airport can have multiple departing flights, but a flight can have only one origin airport.

- DESTINATION_AIRPORT between (Flight, Airport)
  - It's N:1 relationship. I.e., an airport can have multiple arriving flights, but a flight can have only one destination airport.

- FLIGHT_CARRIER between (Flight, Carrier)
  - It's N:1 relationship. I.e., a carrier can have multiple flights, but a flight can be operated by one carrier. May not be true for all airlines: sharecodes

Adding Relationships to ER diagram



Entity-Relationship Model Outcome

Summary

In this lecture, we introduced the Entity-Relationship modeling tool.

Concepts covered:

- Entities, Relationships, Attributes
- Relationship constraints
- Weak Entities
- Pictorial representation of ER Model