# Lecture 11: ER Diagrams

*CS1106/CS6503– Introduction to Relational Databases*

Dr Kieran T. Herley

2019-2020

School of Computer Science & Information Technology
University College Cork

## Summary

*The database design process and the role of ER modelling in same. Basic constituents of ER diagrams: entities, relationships and attributes.*

# Database Models

- While designing database it's handy to have notation to "sketch" schemas
- Two common diagrammatic notations:
    - UML diagrams (Universal Modelling Language)
    - ER diagrams (Entity Relationship) → cs1106 choice
- Captures the information requirements of our database: what data it contains and how these interrelate

# Database Development Process

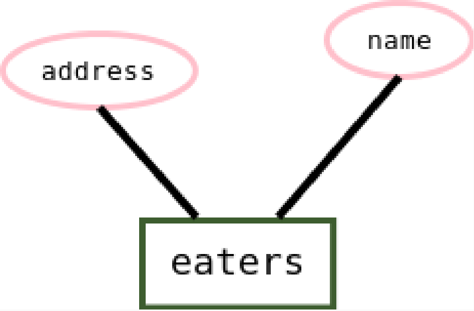
1. Requirements analysis: figure out what is required of our database
2. Design:
   - Develop an ER diagram to sketch a design that meets these requirements
   - "Translate" ER diagram into detailed DB schema
3. Implementation: Generate SQL code for design

# Entities

- An *entity* is a “thing” e.g. a person
- An *entity set* is a collection of similar entities e.g. a collection of persons
- An *attribute* is some property of the entities in an entity set e.g. the date of birth of each person

# Entities in ER Diagrams

- Will represent entity sets using rectangles, attributes as ovals



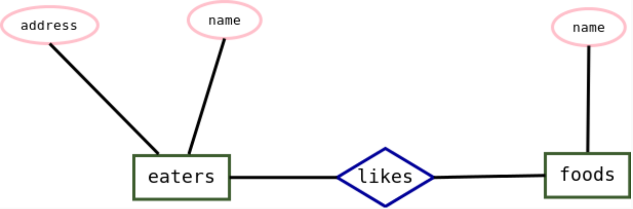
- Entity set `eaters` has attributes name and address; each entity with this set has a value for each attribute

# Relationships

- Relationship captures a connection between two or more entity sets
- Example:
  - Entity sets: students, modules
  - Relationships:
    - is_taking: Alice is taking cs1106
    - loves: Bob loves cs1109

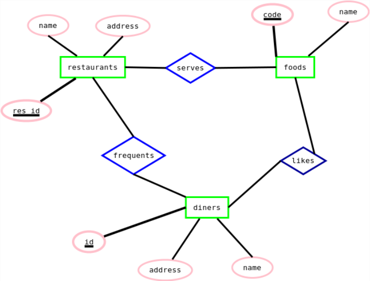
# Relationships in ER Diagrams

- We use diamonds to depict relationships



- Connect diamond to entity sets involved in relationship

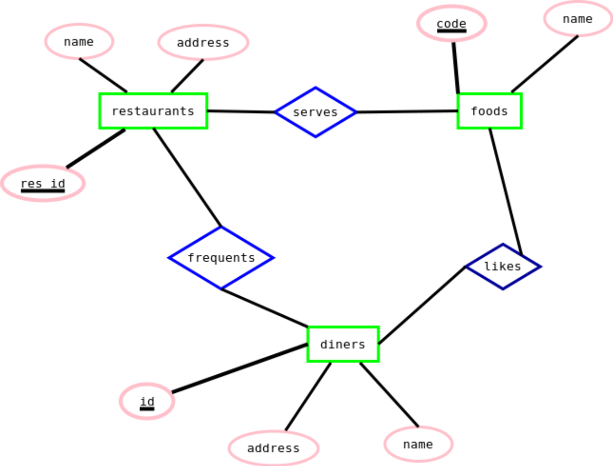
# A More Complex Example



Key:

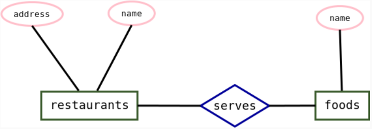
- Restaurants serve some foods
- Eaters like some foods
- Eaters frequent some restaurants

# A More Complex Example cont'd



# Relationship Sets

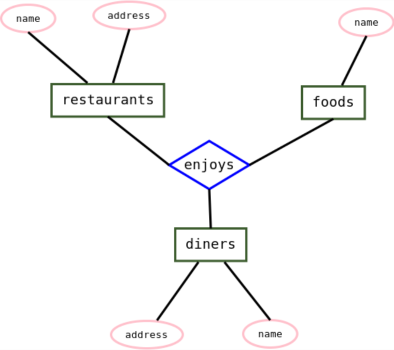
- The "value" of an entity set is the set of entities that currently belong to it e.g. the set of eaters in our database
- The "value" of a relationship is a *relationship set*, the set of tuples with one component for each related entity set
- Example (`serves` relationship)



| Restaurant | Food |
|---|---|
| Luigi's | Spaghetti |
| Luigi's | Lasagne |
| Luigi's | Pizza |
| Gunther's | Bratwurst |
| Gunther's | Sauerkraut |
| Pierre's | Beans on Toast |
| Pierre's | Pizza |

# Binary and Multiway Relationships

- Relationships such as likes, frequents, serves are all *binary* i.e. involve two entity sets
- We sometimes need relationships involving three or more entity sets

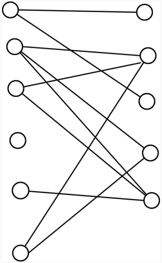


# Relationship Sets cont'd

| Restaurant | Eater | Food |
|---|---|---|
| Luigi's | Alice | Spaghetti |
| Pierre's | Alice | Beans on Toast |
| Luigi's | Bob | Spaghetti |
| Gunther's | Bob | Bratwurst |
| Pierre's | Clare | Pizza |

# Many-Many Relationships

- Consider likes relationships between eaters and foods
- In many-many relationship each entity in either entity set can be connected with many entities from the other entity set
- Example (`likes` relationship):

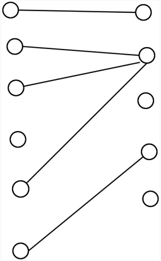


- Each eater may like many different foods
- Each food may be liked by many eaters

# Many-One Relationships

- In a many-one relationship each entity from the first set is connected to at most one entity in the second set
- Note that each entity in the second entity set can be connected to many entities in the first
- Example:

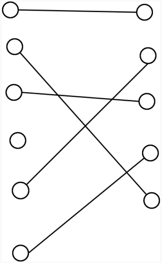
- Imagine a `favourite_foods` relationship between eaters and foods
- Each eater can have (at most) one favourite food
- A food can have any number of eaters form whom it is their favourite: zero, one, two, many



# One-One Relationships

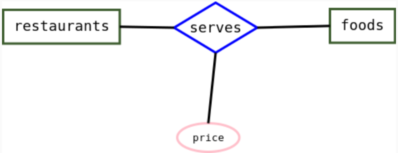
- In a one-one relationship each entity in either set is connected to at most one entity in the other entity set
- Example:

- Imagine `is_head_chef` relationship between chefs and restaurants
- Each restaurant has (at most) one head chef
- Each chef can be head chef in at most one restaurant (we assume)



# Relationships With Attributes

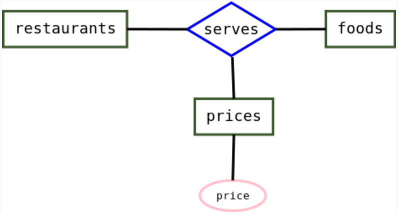
- It can be useful to associate attributes with relationships instead of entity sets



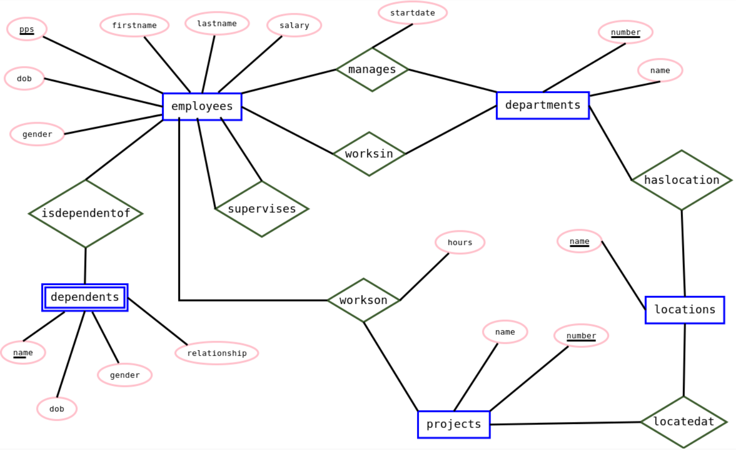
- Intuitively such an attribute is a property of the tuples in the relationship

# Relationships With Attributes cont'd

- Can usually re-work relationship-attributes by spinning off a separate entity set



# An Example


pps
firstname
lastname
salary
startdate
number
name
dob
manages
employees
departments
gender
worksin
haslocation
isdependentof
supervises
hours
name
dependents
workson
locations
name
number
name
relationship
gender
dob
projects
locatedat

# Acknowledgements

This lecture is based on material from the text by Ullman and Widom and their website.