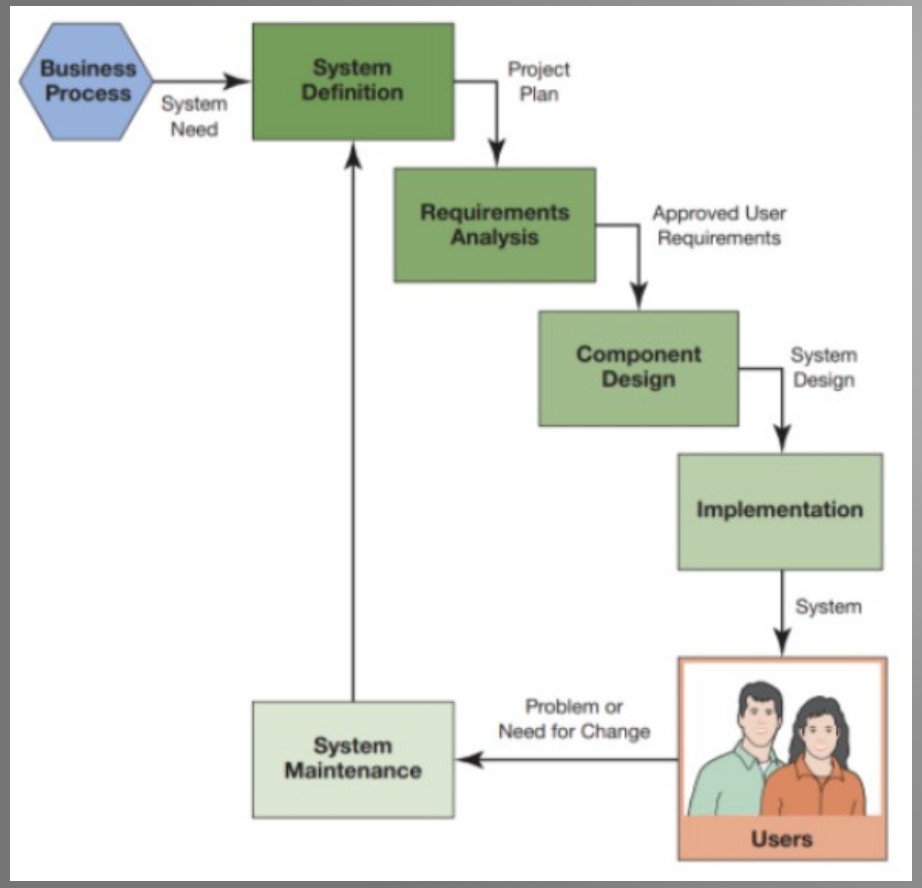
Entity-Relation Diagrams

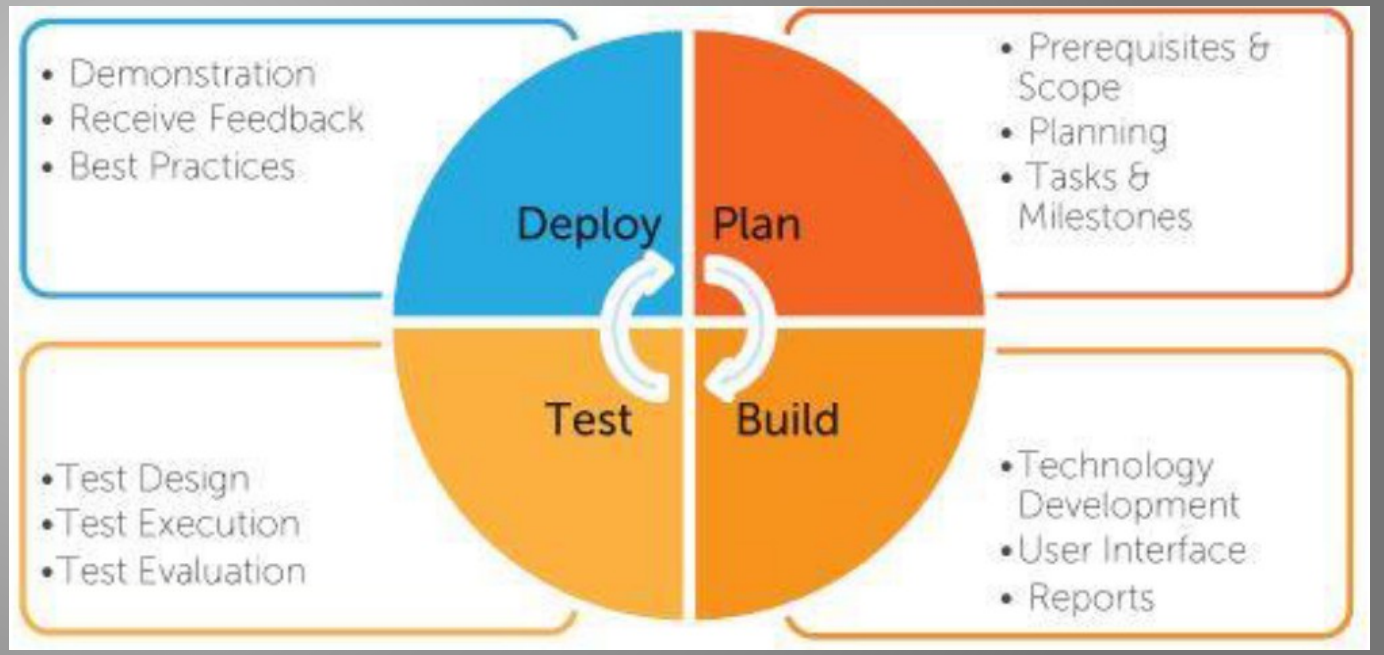
Database Design

UNIVERSITY OF MONTANA

Software Development Life-Cycle

- Commonly either waterfall, or rapid prototyping

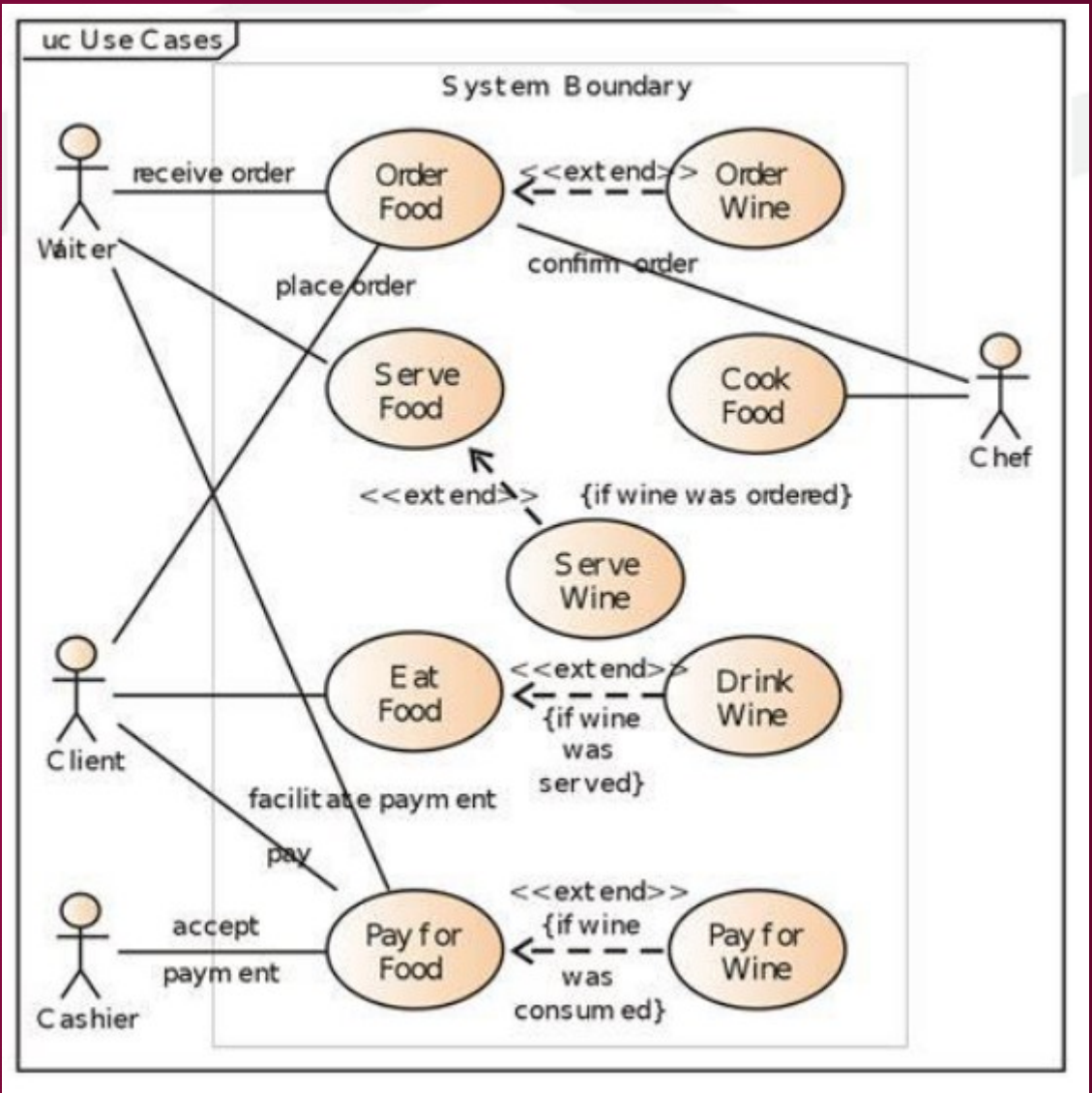




UNIVERSITY OF MONTANA

Use Case Diagram

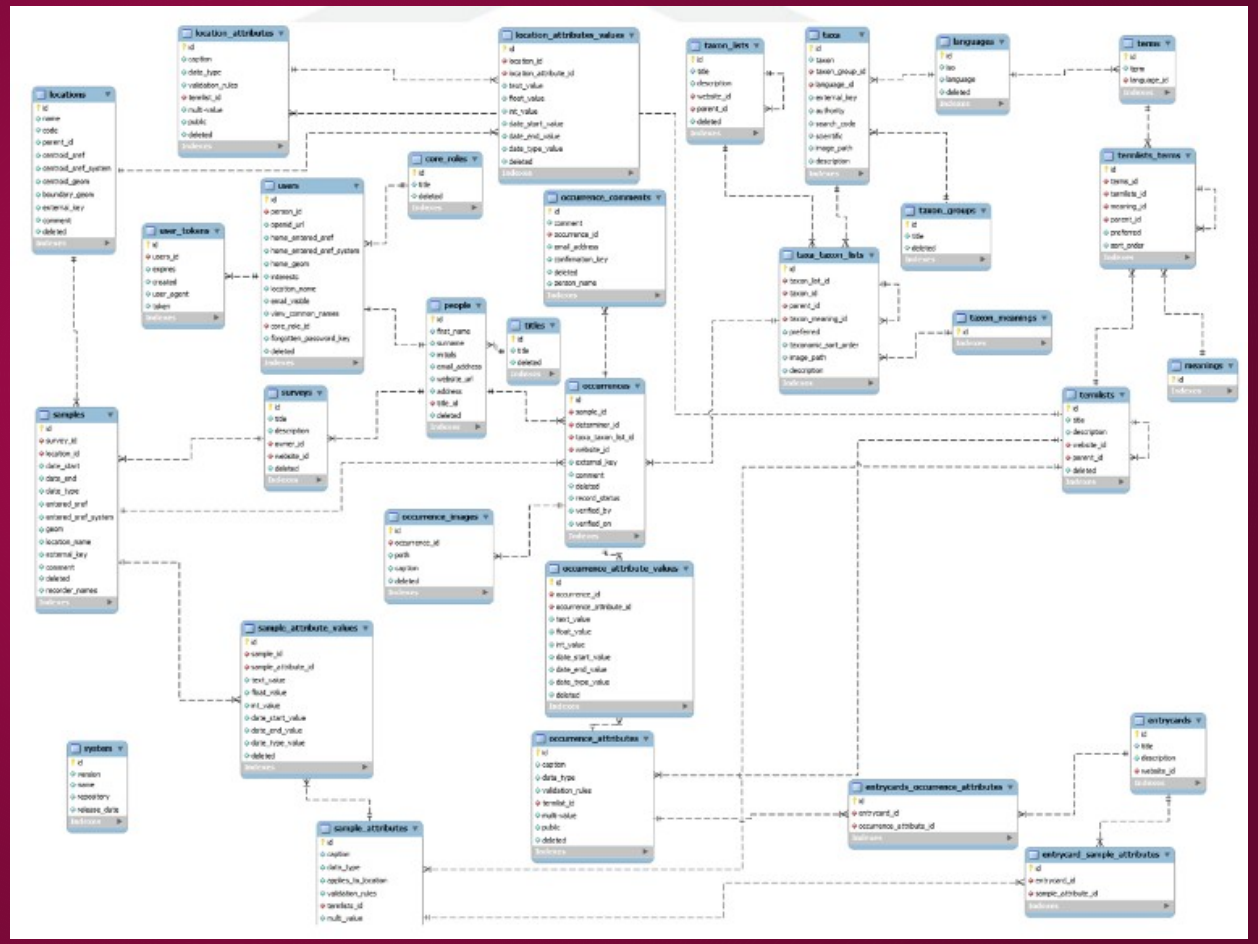
- Actors
  - Active
  - Passive
- Use Case
  - Extend
  - Import
  - Generalization



https://www.youtube.com/watch?v=zid-MVo7M-E

ER Diagrams

- Entities
- Relationships
- Attributes



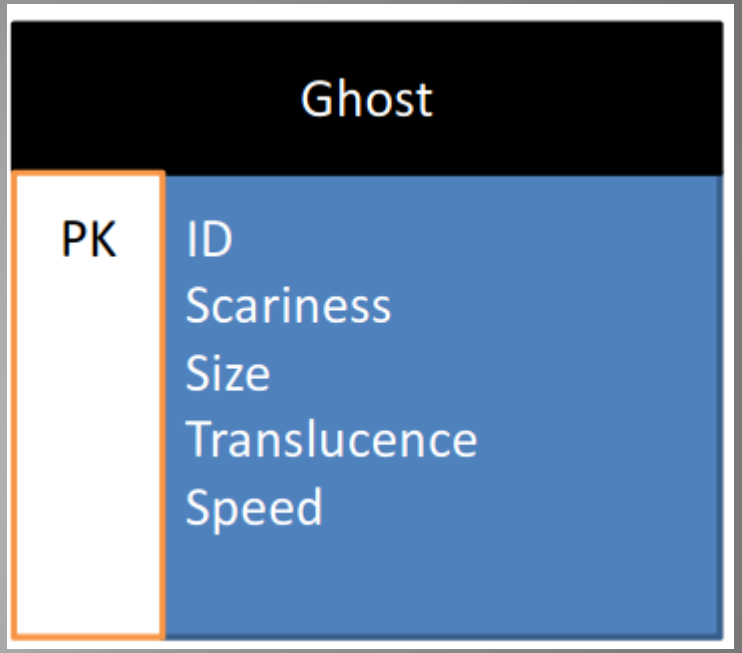
https://www.youtube.com/watch?v=QpdhBUYk7Kk

Entity Type

- A set of things
  - Ex: All employees in a company
- Should be singular (Employee, not Employees)
- “Bob Jones, the employee” is an instance of this entity type

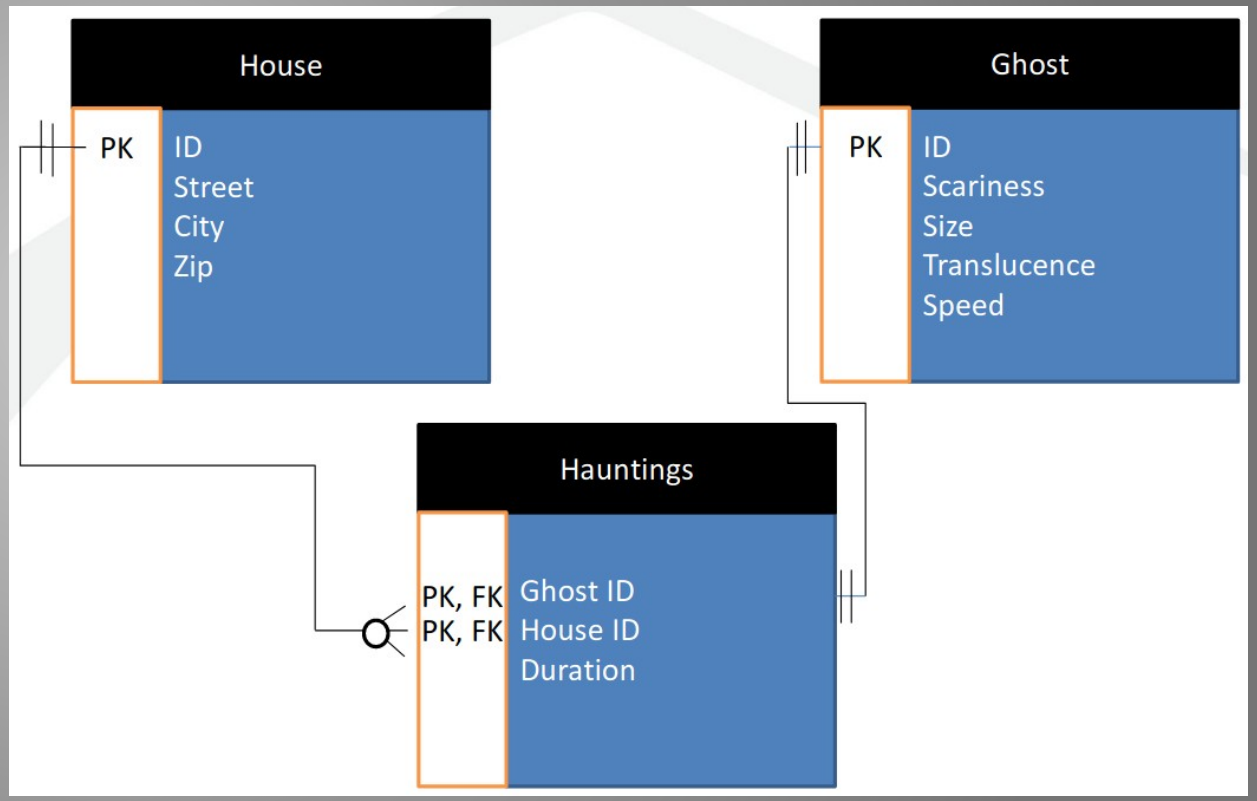
Attribute types

- A set of values
  - Ex: All employee salaries
- Bob has a salary of $40,000 is an instance of an attribute



Relationship type

- A statement about entity types
  - Example: Employee-Manages-Department
- “Bob manages Sales” is an instance of this relationship

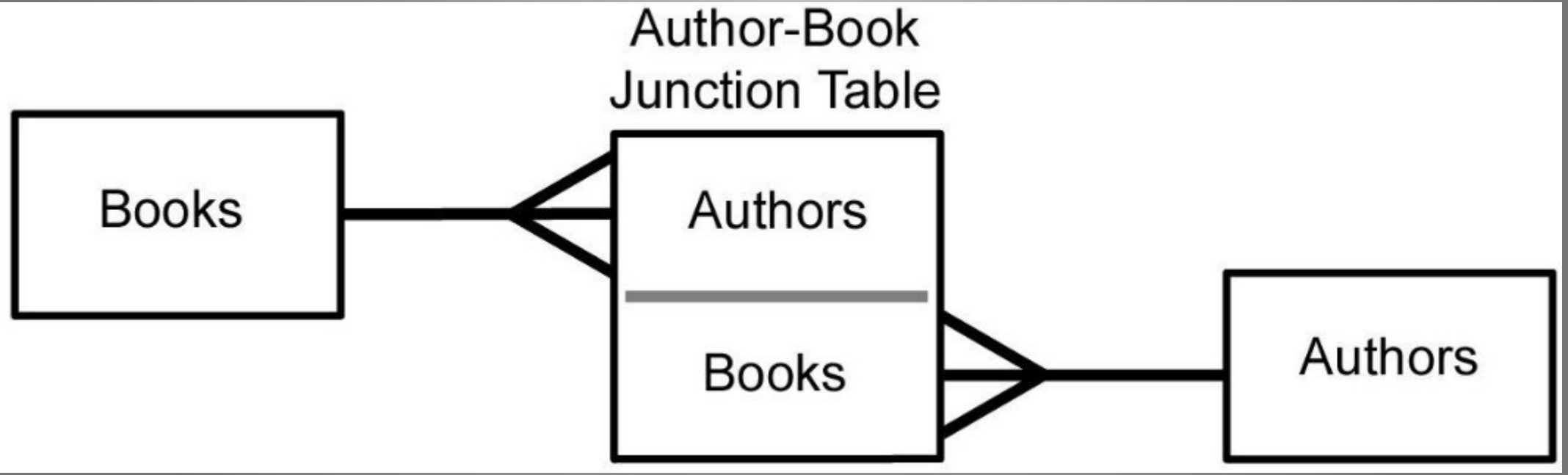


Many-to-many relationship

- How do we represent a many-to-many relationship in a table?

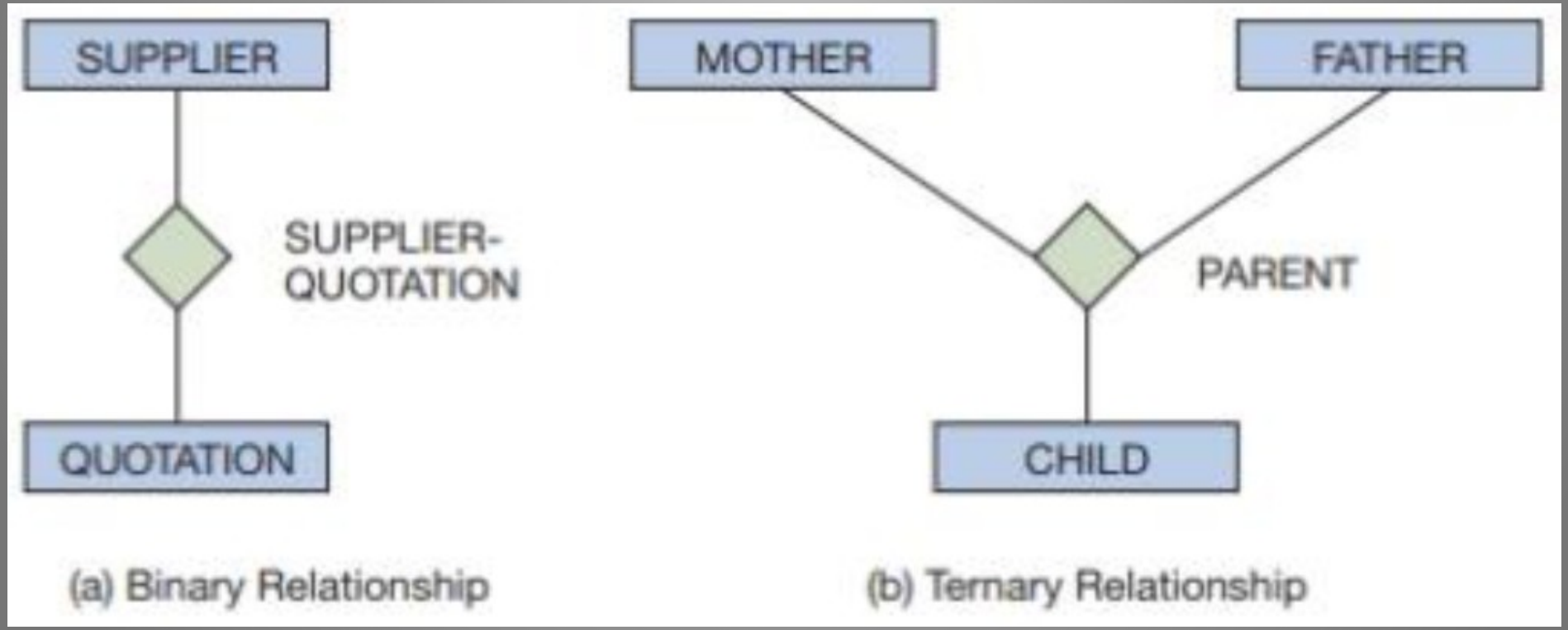
Many-to-many relationship

- How do we represent a many-to-many relationship in a table? Add a bridge (or junction) table



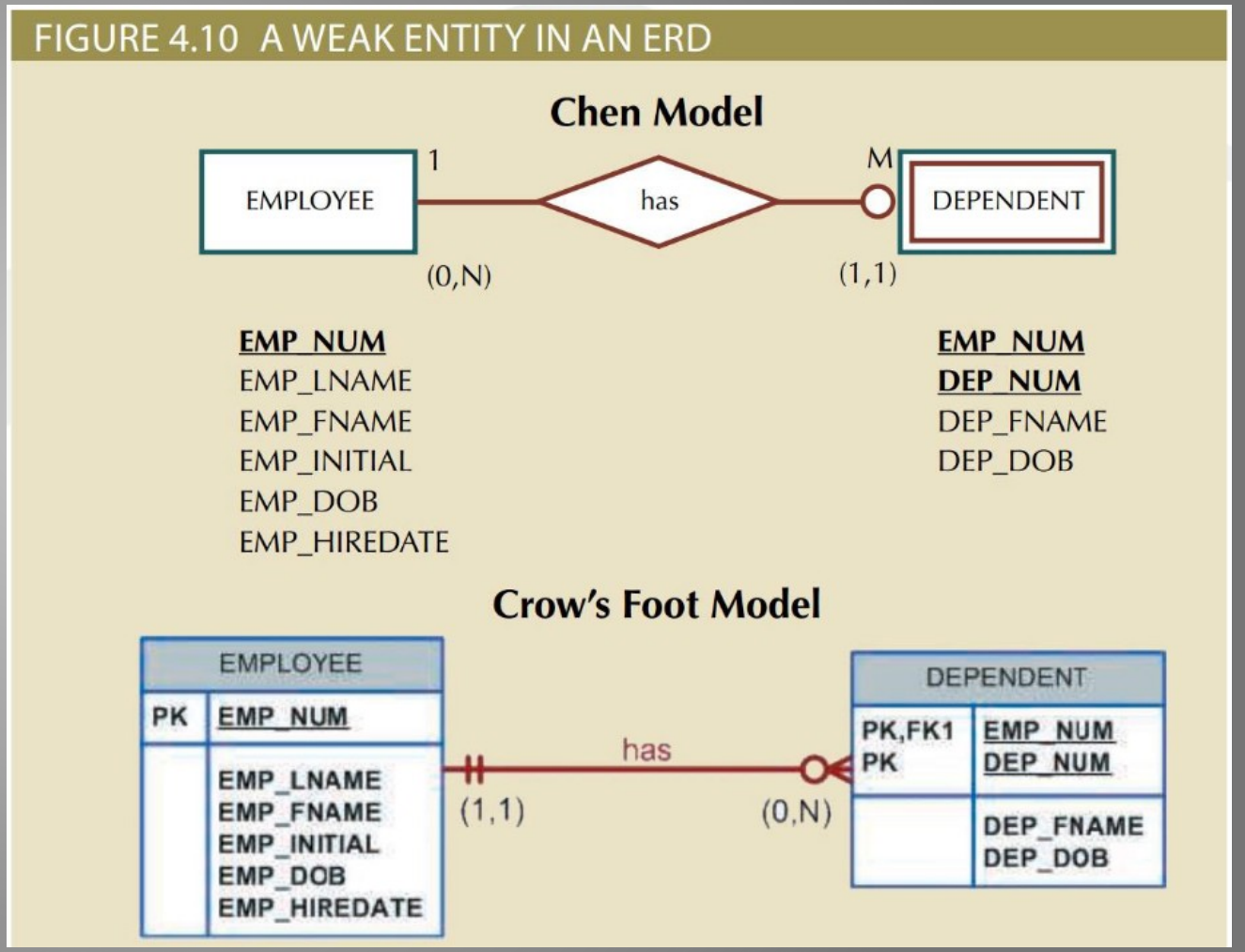
Other ER diagrams

- There are other styles used, but the same ideas are expressed



Other ER diagrams

- There are other styles used, but the same ideas are expressed



Phases of Database Design

- Analysis
- Logical Design
- Physical Design

Analysis

- Develop an ER model
  - Discover entities, relationships, and attributes
  - Determine cardinality
  - Distinguish independent and dependent entities
  - Create supertype and subtype entities

Logical Design

- Convert the ER model into tables, columns, and keys
  - Implement entities
  - Implement relationships
  - Implement attributes
  - Normalize tables

Physical Design

- Add indexes
- Specify how tables are organized

Discovery (Starting Analysis)

- Interview the database users and managers
  - Entities are usually nouns
    - Not all nouns are entities
  - Relationships are usually verbs
    - Describe connection between entities
  - Attributes are usually nouns
    - Denote specific data (names, dates, etc)

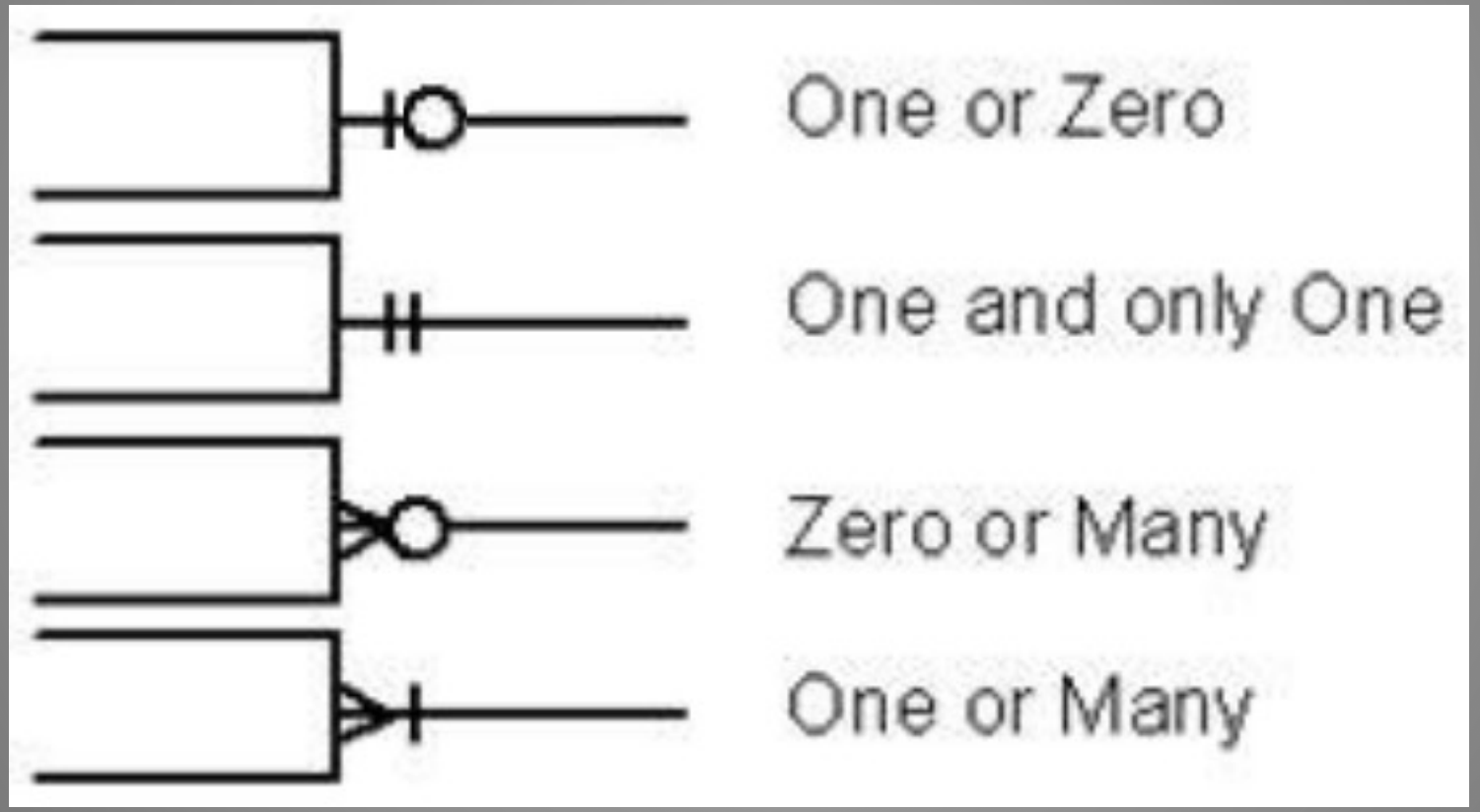
Names

- Entity Names
  - Singular noun
  - Easily understood
  - Consistent (don’t use synonyms)
- Relationship Names
  - Entity-Verb-Entity
  - Verb (if the entities are obvious)
  - Should be active (owns, rather than owned-by)
- Attribute Names
  - EntityQualifierType (EmployeeFirstName, Or VehicleIdentificaionNumber)
  - When obvious, parts can be omitted

Cardinality

- Relationship maximum
  - Usually 1 or M
  - One maximum for each entity in a relationship
- Relationship minimum
  - Usually 0 or 1
  - One minimum for each entity in a relationship

Crows foot notation



Cardinality

- Attribute maximum, can be described by:
  - Singular - each entity has at most one attribute instance
  - Plural - each entity can have multiple attribute instances
  - Unique - No two entities can have the same attribute instance
- Attribute minimum
  - Either required or optional
- Examples:
  - EmployeeNumber 1-1(1) - Unique, max of 1, required
  - PassportNumber 1-M (0) - Unique, max of many, not required
  - FullName M-1 (1) - Not unique, max of 1, required

Independent and dependent entities

- Dependency relations exist if one entity exists only in relation to another instance
  - A passport doesn’t make sense without a person and a country, so it would be dependent on both
  - A music album doesn’t make sense without the songs in that album, so ‘Album’ would depend on ‘Song’
- Independent entities are sometimes called strong
- Dependent entities are sometimes called weak

Supertype and subtype entities

- A subtype is an entity that is usually specialized version of an entity
  - Managers are employees, but not all employees are managers
    - Employee is the supertype
    - Manager is the subtype