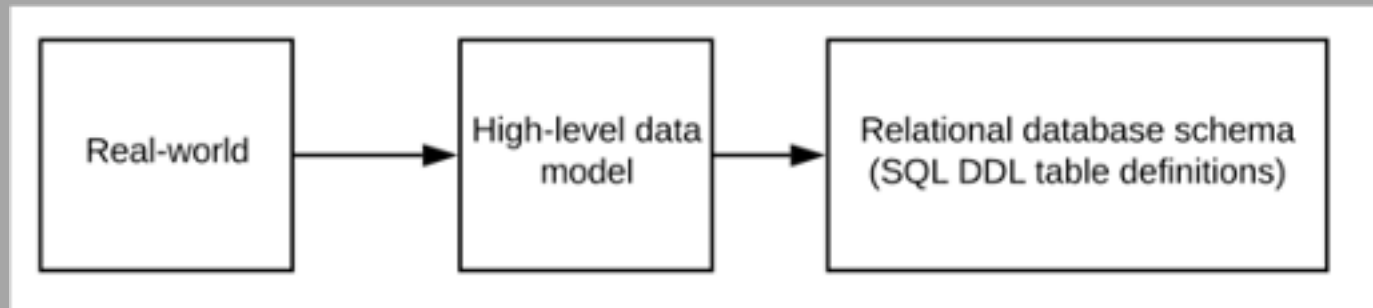


The Entity Relationship Model 1

- Reading:
The “High level database models” chapter of the textbook.

Purpose of the ER Model

- Database design often starts with a high level data model.



- The ER model is a **high-level data model** that describes data objects and their relationships. It was proposed by Peter Chen in 1976 in his seminal paper:

Chen, Peter Pin-Shan. "The entity-relationship model—toward a unified view of data." *ACM Transactions on Database Systems (TODS)* 1(1) (1976): 9-36.

The ER model: entity set

- An entity is an object with properties that have information, or data. For example a student with ID s1234 named John Citizen.
- A class of objects with the same properties is an **entity set**, or **entity type**.
- So strictly speaking an entity is an entity instance or entity occurrence. An entity set is sometimes simply called an entity.

Entity: attribute and key

- Attribute: a property of an entity set.
 - Student number, first name, last name of the Student entity set.
- (Candidate) key: The minimal set of attributes that can uniquely identify each entity.
- Primary key: The key that is selected to identify each entity.

Entity ...

- An entity can be an object with physical (or “real”) existence or objects with a conceptual (or “abstract”) existence.
 - But “object” is not defined. Different database designers have different views.
- Each below is an entity?
 - A student
 - A course
 - A transaction (at a supermarket)

The ER model: relationship

- Relationships describe associations between two or more entities.
 - For example, students ENROL-IN courses.
- A relationship is the association between entities of the participating entity sets.

The degree of relationships

- The number of participating entity sets is the **degree of a relationship**.
 - Binary: Student enrols in a course.
 - Ternary: Agent registers a client at a property.
 - Quarternary: Solicitor signs a contract for a client with a bank.
- Binary relationships are common in the real world.

Relationship constraints: multiplicity

Constraints specify real-world restrictions on relationships.

- A relationship has multiplicity, or cardinality: for an entity from the other end, at maximum how many entities of this end can relate to.
- Cardinality for can be one or many .
- There are one-one, one-many or many-many binary relationships.
- Cardinality constraint can generalise to relationships of higher degrees.

Relationship constraints: referential integrity

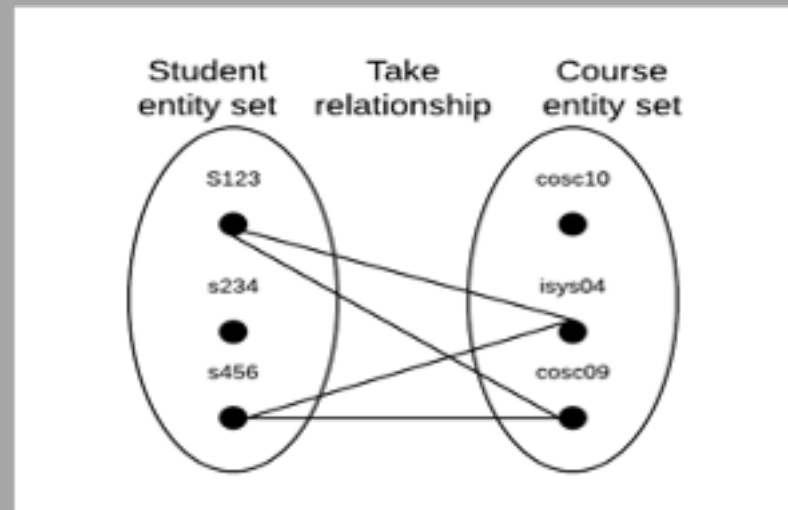
- The referential integrity constraint for an M:1 relationship from E to F states that for each entity of E there must exist a “referenced” entity of F via this relationship.
 - For an entity of E, there is maximal one and minimal one, or exactly one, referenced entity of F.
- Generally, given a relationship from E to F, for an entity of E, the minimal number of related entities of F is ONE (The default is zero).

Example: many-many relationship - Student Take Course

Given the Take relationship,

- Each entity of the Student entity set is related to minimal zero and maximal many entities of the Course entity set.
- Each entity of the Course entity set is related to minimal zero and maximal many entities of the Student entity set.

So Take is a many-many relationship.

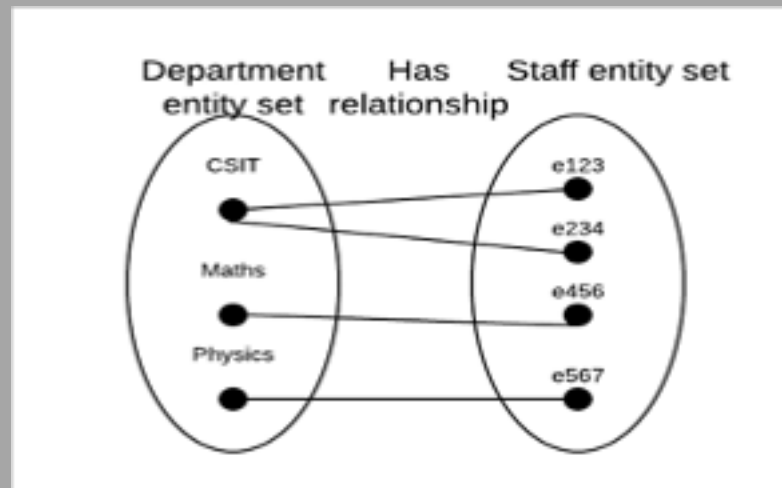


Example: one-many relationship - Department Has Staff

Given the Has relationship,

- Each entity of the Department entity set is related to minimal one and maximal many entities of the Staff entity set.
- Each entity of the Staff entity set is related to a minimal one and maximal many entities of the Department entity set.

So Has is a one-many relationship, and it has referential integrity constraint from Staff to Department -- a department entity related to each given staff entity must exist.

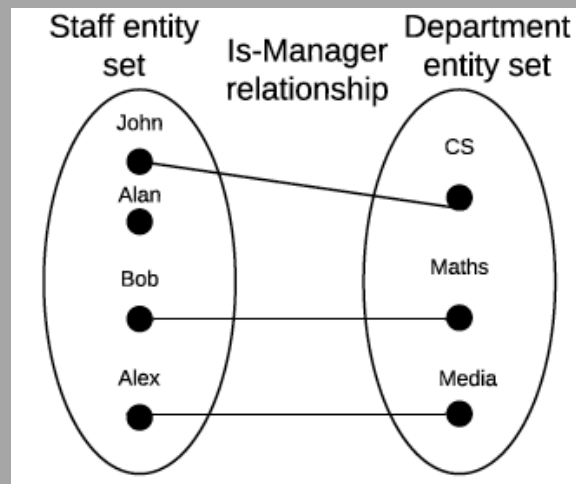


Example: one-one relationship - Staff Is-Manager Department

Given the Is-Manager relationship,

- Each entity of the Staff entity set is related to minimal zero and maximal one entity of the Department entity set.
- Each entity of the Department entity set is related to minimal one and maximal one, or *exactly one*, entity of the Staff entity set.

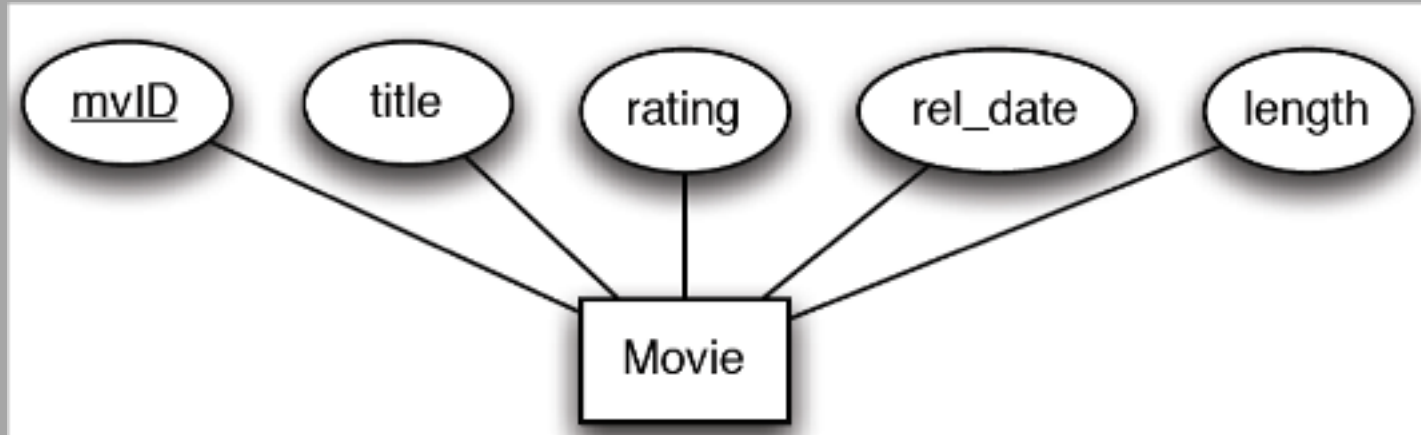
So Is-Manager is a one-one relationship, and it has referential integrity constraint from Department to Staff – a Staff entity must exist for each given Department entity.



ERD: diagram for the ER model

- The concepts of ER modeling are expressed as a diagram called the **entity-relationship diagram (ERD)**. There are many ERD notations:
 - Chen notation (basis for the textbook notation)
 - Crow's foot notation

ERD: Entity Set and Entity



The **entity set** is represented using a rectangle. Attributes are ovals attached to the rectangle. Primary key attributes are underlined.

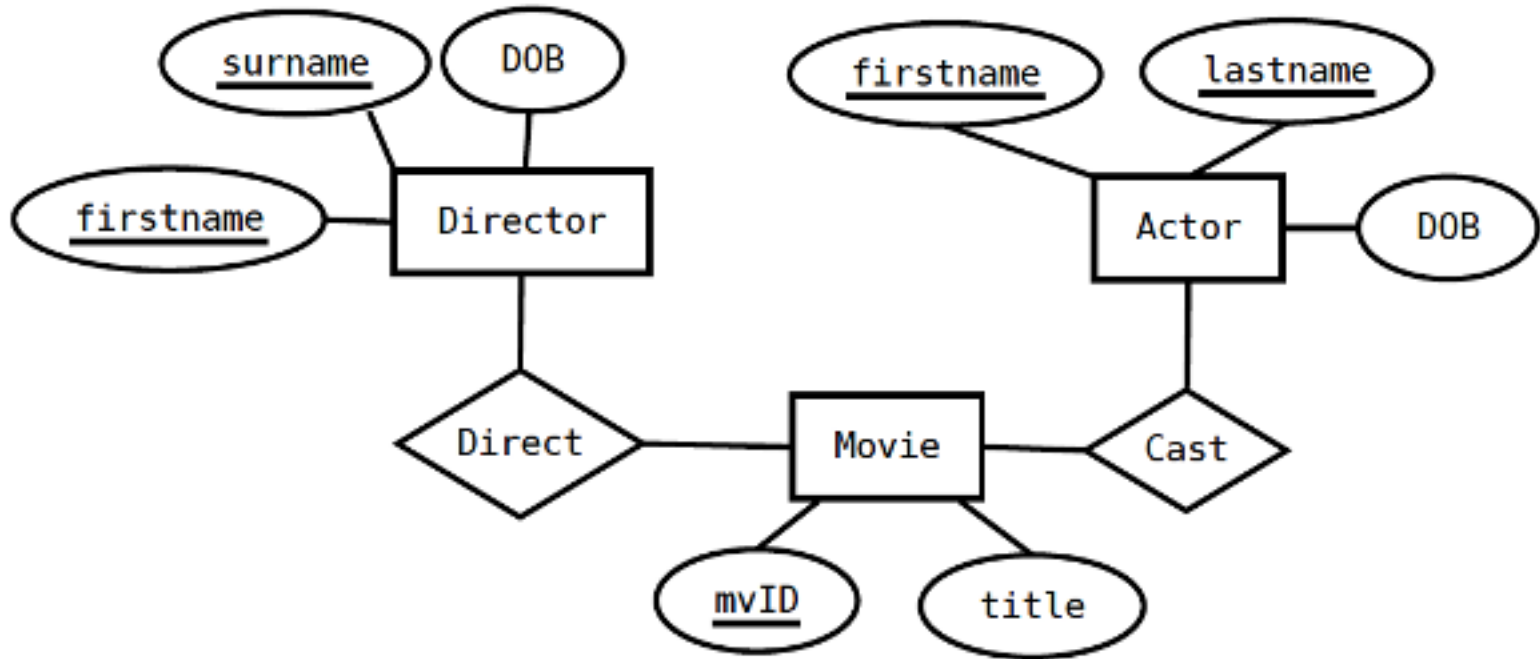
ERD: Relationship

- A **relationship** connects two or more entity sets.
 - A binary relationship connects two entity sets.
- A relationship is represented by a diamond, with lines to each of the entity sets involved.



Example: Relationships

- Entity set Director has a **composite key** of two attributes: firstname and last name.
- Directors direct movies. Actors cast in movies.



Be Precise: The Multiplicity of Relationships

- One-One relationship: arrow entering both sides.



- Many-One relationship: arrow *entering* “one” side.

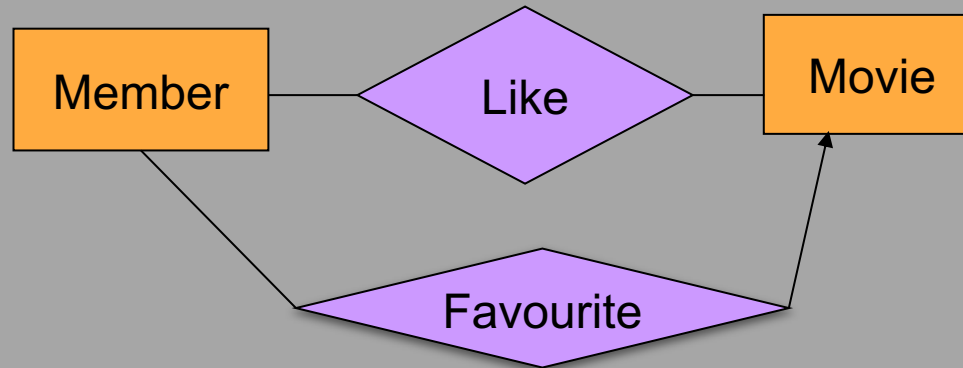


- Many-Many relationship.



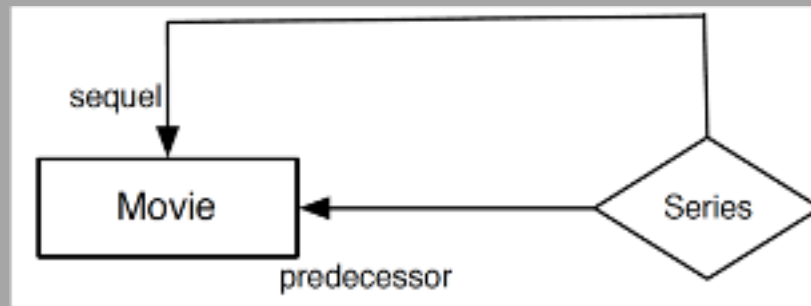
Multiplicity of relationships: examples

- Two different relationships connect the same entity sets.
- “Like” is a many-to-many relationship: a member can like many movies and a movie can be liked by many members.
- “Favourite” is a many-to-one relationship: a member can have one favourite movie but a movie can be the favourite of many members.



Reflexive relationship and roles

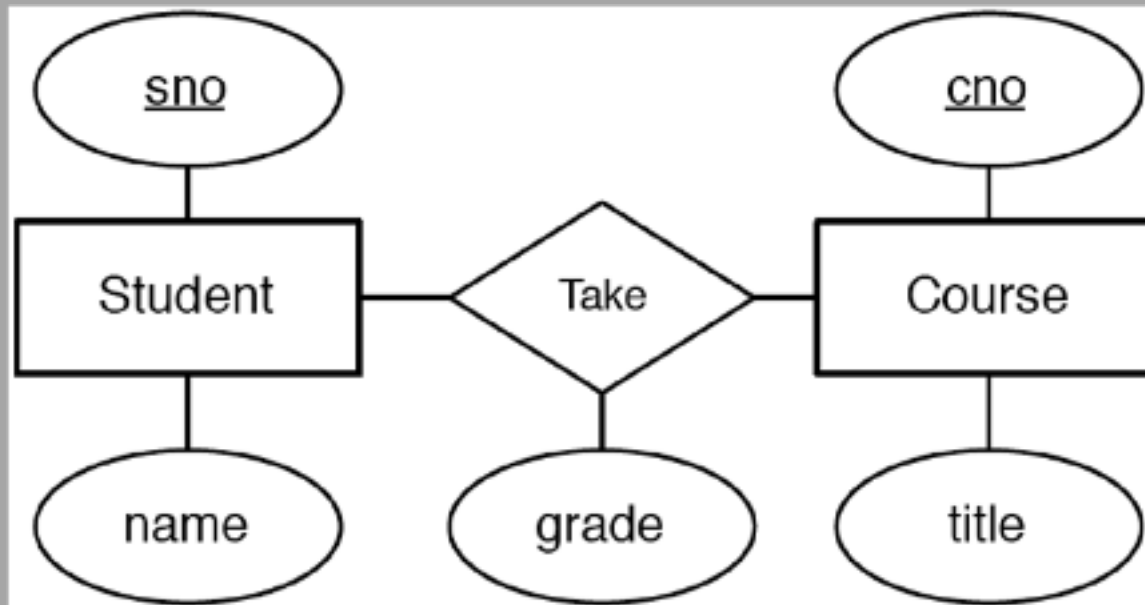
- Sometimes an entity set appears more than once in a relationship.
- Label the edges between the relationship and the entity set with names called **roles**.



Predecessor	Sequel
Romancing the Stone	The Jewel of the Nile
Batman Begins	The Dark Knight
Lethal Weapon	Lethal Weapon 2
Spider Man	Spider Man 2
Spider Man 2	Spider Man 3
Toy Story	Toy Story 2
Ice Age	Ice Age 2
Ice Age 2	Ice Age 3

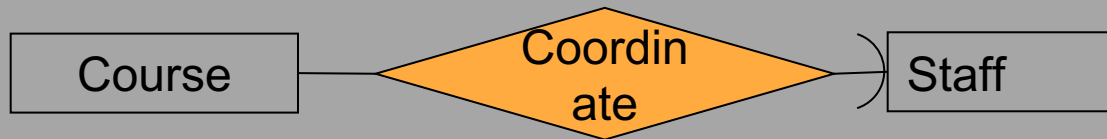
Attributes on Relationships

- Sometimes it is useful to attach an attribute to a relationship.
- Think of this attribute as a property of tuples in the relationship.



Relationships: referential integrity

The referential integrity constraint for an M:1 relationship from E to F states that for an E entity that must exist an “referenced” F entity via this relationship.



ERD: summary

- The ERD aims at a high level view of enterprise data objects in the real world.
- Entities and relationships are explicitly represented.
- The ERD is a high-level data model at the conceptual level independent of the database system implementation.

The UML data model

- Unified Modeling Language (UML) is an object-oriented modeling language for software design.
- UML has many modeling diagrams.
 - Class, sequence, use case, deployment
- For data modeling, UML is popular for modelling application data.

The UML data model

- UML offers the same capabilities as the ER model, except the multiway relationship.
 - UML allows only binary relationships.
 - Multiway relationships have to be converted into multiple binary relationships (to discuss later).
- UML has **Class** for the entity type, and **Association** for the binary relationship.

The UML Data Modelling: five concepts

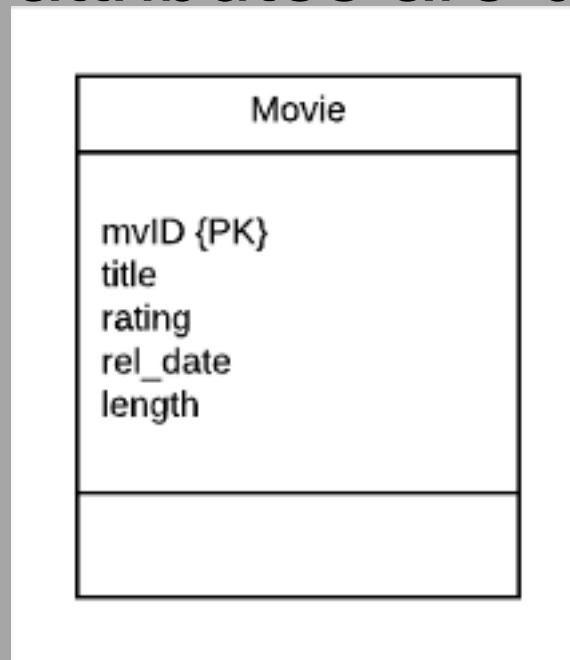
- Class
- Association
- Association class
- Composition and aggregation
- Subclass

UML vs ER model

UML	ER Model
Class	Entity set
Association Class	Binary relationship
Association Class	Attributes on a relationship
Subclass	ISA relationship
Aggregation	Many-one relationship
Composition	Many-one relationship with referential integrity

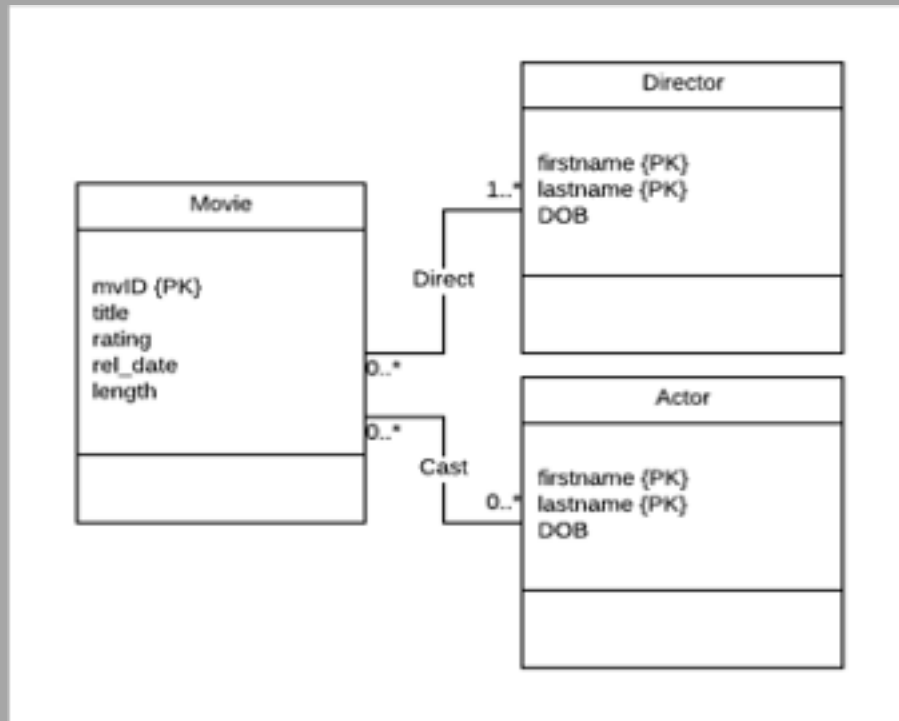
UML: Class

- A UML class is similar to an ER entity set. It has three sections: class name, attributes and methods.
- Primary key attributes are denoted by {PK}.



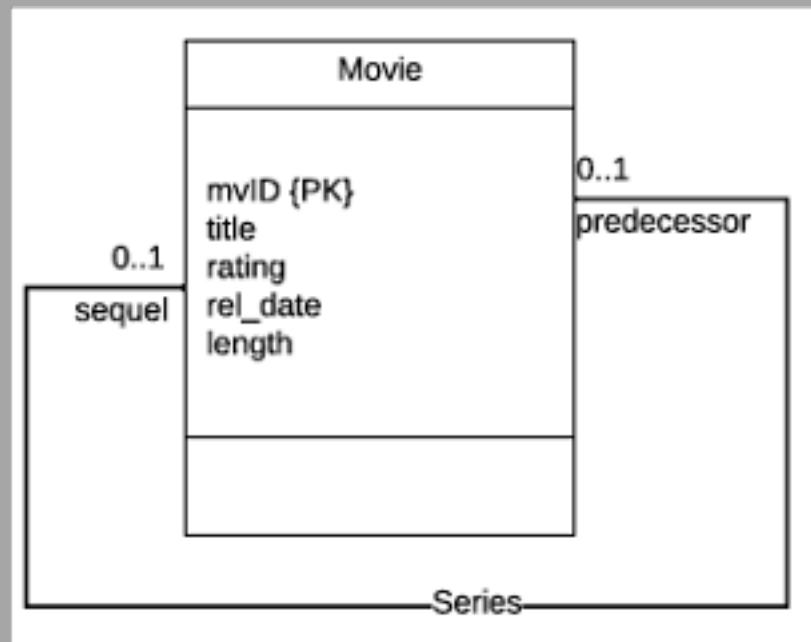
UML: Association

- Association between classes represents a binary relationship.
- Multiplicity and referential integrity constraints are represented as minimum..maximum, which can be 0, 1, or * (many).
- Short form notations: 1 for 1..1 and * for 0..*.



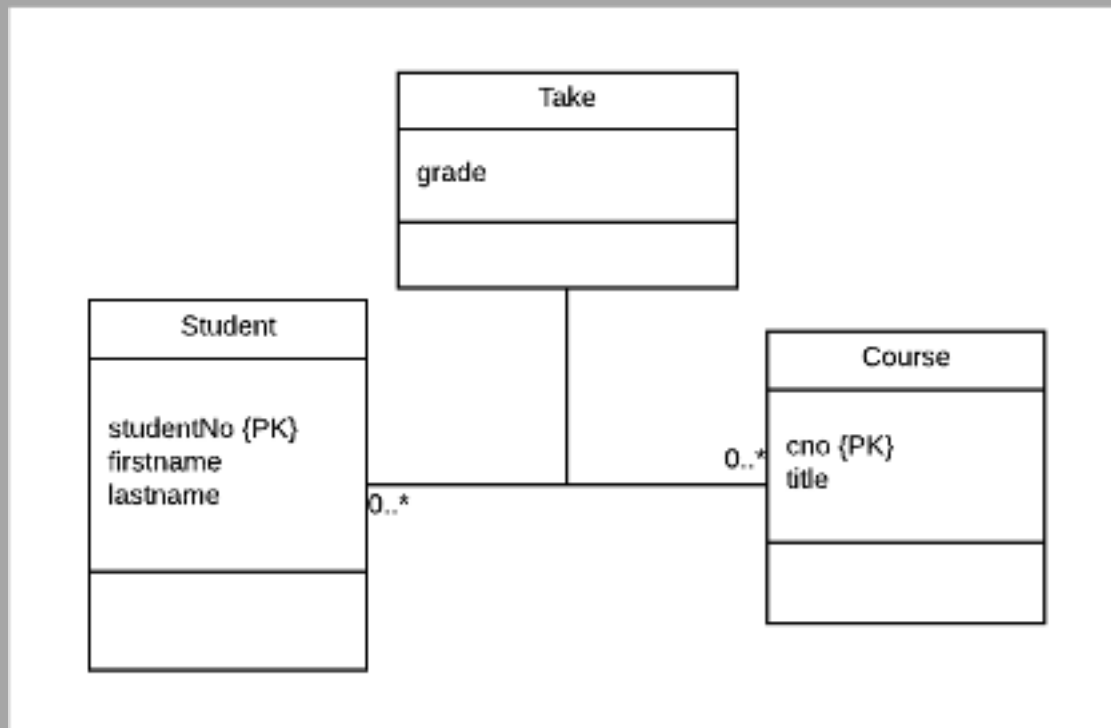
Self-association

Self-association for the same class is similar to the reflexive relationship for the same entity set.



Association class

The association class with attributes is like a relationship with attributes. An association class has no PK attributes.



Aggregation and Composition: elaboration on the M:1 and 1:1 relationships

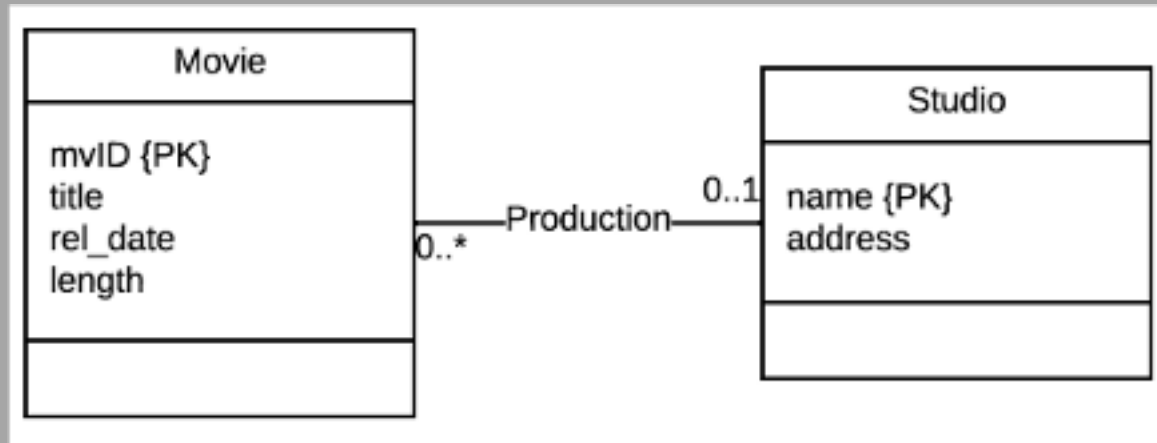
The M:1 (including 1:1) relationships can be elaborated using special notations to stipulate on how the model should be mapped to relations in the relational database schema (discussed later).

- The *Aggregation* is a M:1 association with a diamond on the “one” end, indicating that the class has zero or one multiplicity.
- A *Composition* is a M:1 association with a solid diamond on the “one” end, indicating that the class has an exact one multiplicity.

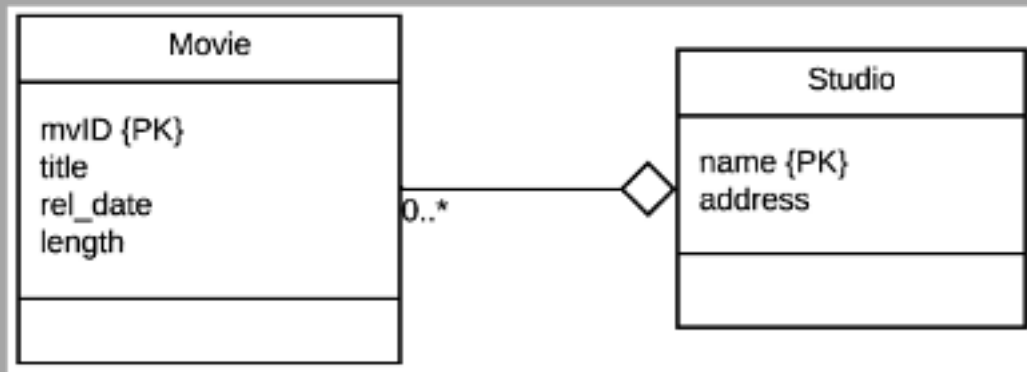
Both Aggregation and Composition associations do not need to have a name, as it will never be mapped into a separate relation in the relational database schema (discussed later).

Aggregation: Example

The production M:1 relationship models that a movie has one production studio and it is possible that such information is not available.

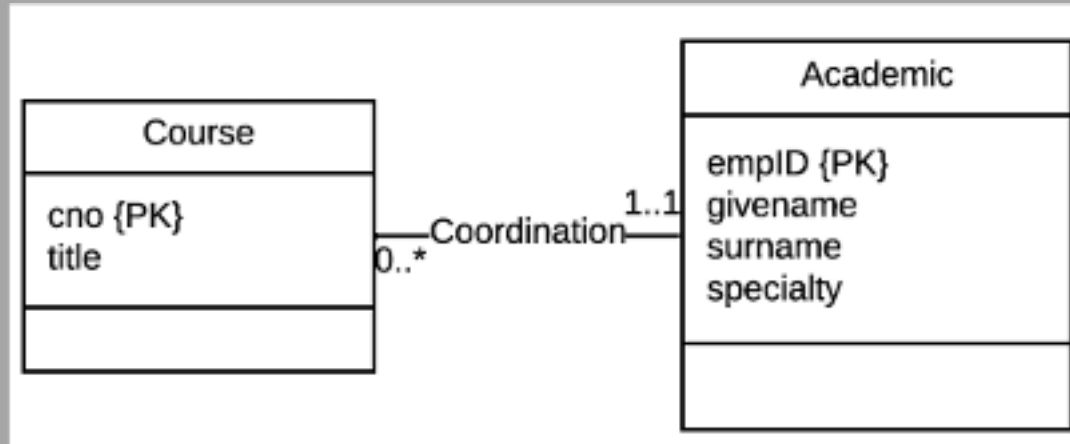


The Production relationship is elaborated as an aggregation relationship. The open diamond indicates 0..1. The name “Production” is not needed.

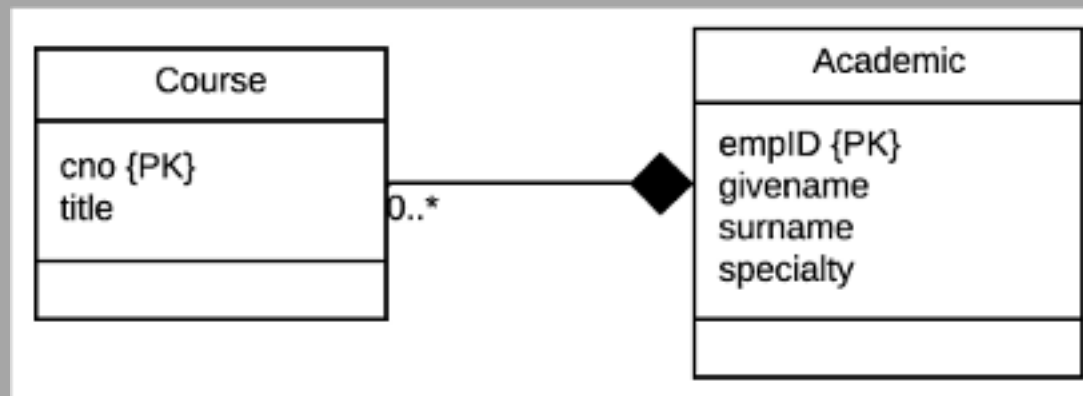


Composition: Example

The Coordination relationship models that a course has exactly one coordinator.



The Coordination relationship is elaborated as a composition. The solid diamond



Database design and diagramming tools

- There are many database design tools that may automatically generate SQL DDL scripts to define the relational database schema.
 - fabForce DBDesigner 4 (open source), Oracle's Designer, IBM's Rational Rose, Computer Associates' ERwin and ER/Studio Data Architect.
- There are diagramming tools for data modelling, software design and other purposes.
 - UML, Visio, omnigraffle
- UML is widely used and will be used for tute exercises and assignments.
 - Lucidchart.com (free EDU account) for UML diagramming.