

Key Constraints

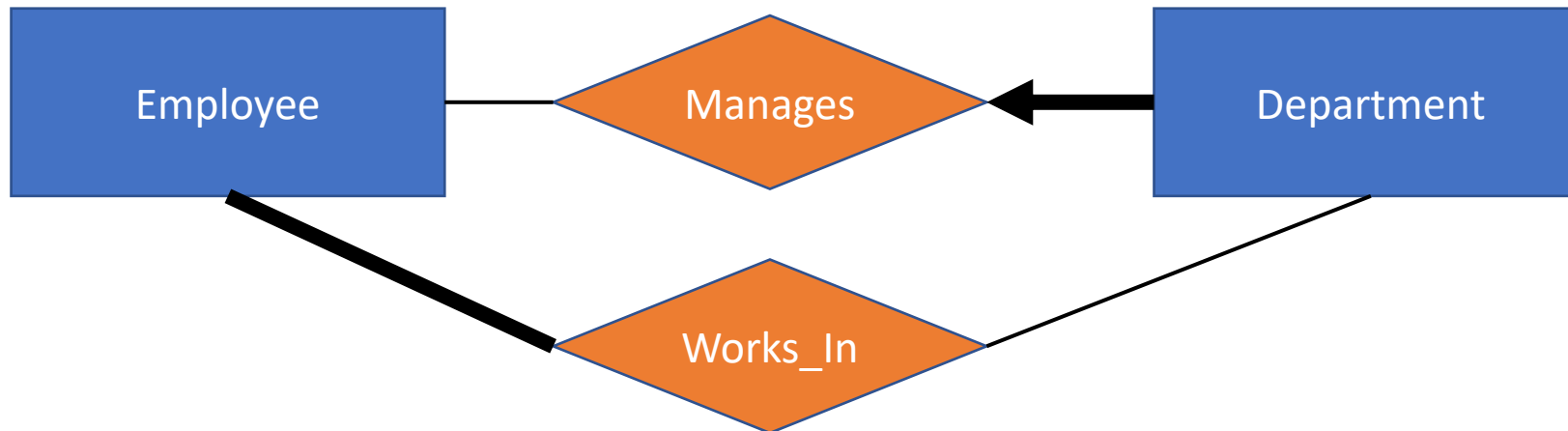
- Restriction between an entity and a relationship indicating each entity can participate in at most one of that relationship
- Indicated by an arrow from the entity to the relationship



- Given a department, we can uniquely determine the Employee that manages the department

Participation Constraints

- Restriction between an entity and a relationship indicating that each entity must participate in that relationship
- Indicated by a bold line



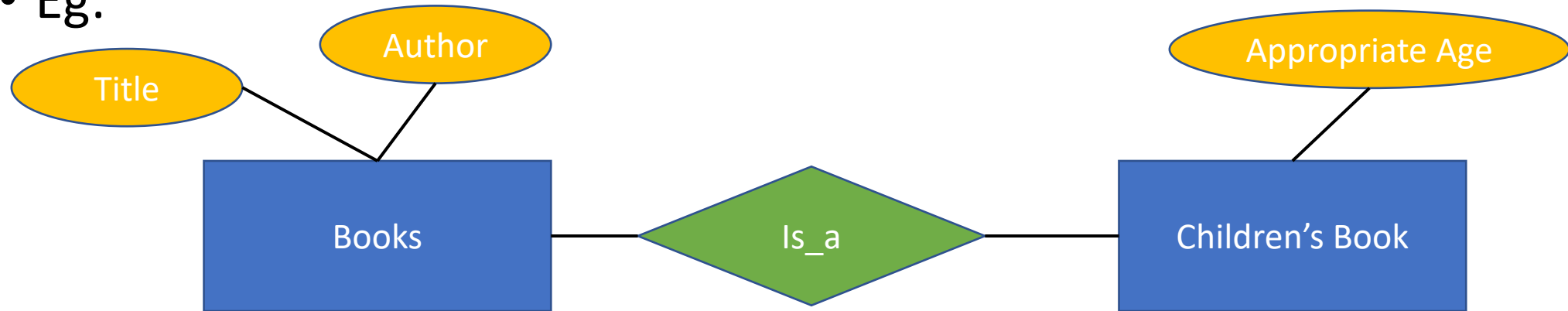
- Participation is said to be “total” (vs “partial”)

More General Cardinality Constraints

- No constraints (notation: simple edge; $0..*$ in cardinality notation)
- At least one relationship for each instance (notation: thick edge; $1..*$ in cardinality notation)
- At most one relationship for each instance (notation: edge with arrow; $0..1$ in cardinality notation)
- Exactly one relationship for each instance (notation: thick edge with arrow; $1..1$ in cardinality notation)

Class Hierarchies

- Classify entities into subclasses
- Subclasses are special cases that have properties not relevant to the parent entity
- Eg:



Class Hierarchies

- Tree: no multiple inheritance
- Different than in OOP, entities have representatives in all classes to which they belong
- If an entity is represented in a subclass, then it is also represented in the superclass (and recursively up the tree)
- Eg: Every entity in Children's Book is also represented in Book

Class Hierarchies

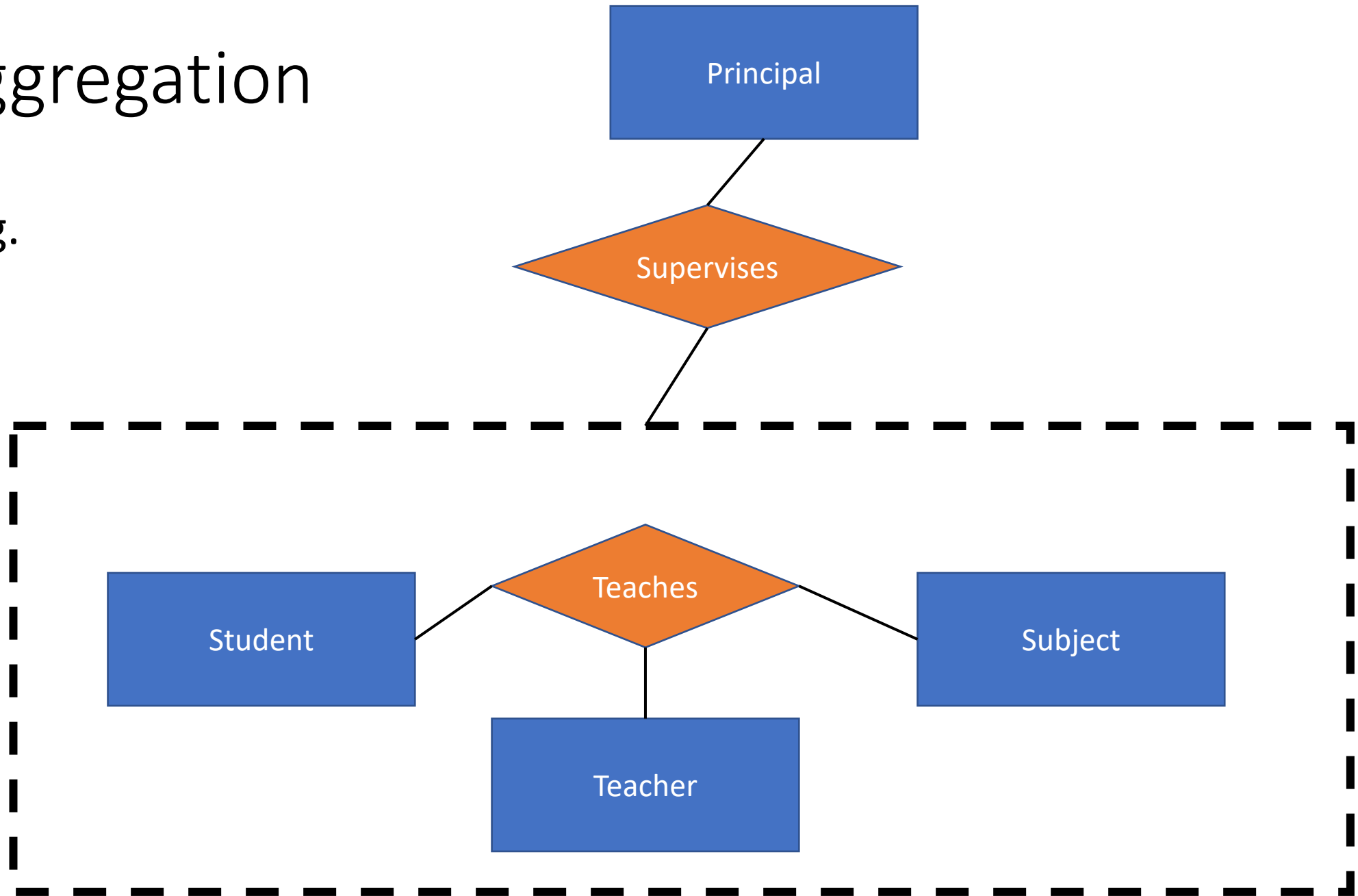
- When to use:
 - Avoid repetition (eg. all books have author and title)
 - Special attributes (eg. appropriate age)
 - Organize large models (inherit changes to schema)
- Constraints:
 - Overlap constraint: Can a book be both a children's book and an adult book? (let's say no -> disjoint) A children's book and a sports book? (yes -> overlapping)
 - Covering constraints: Does every book need to be either a children's book or adult book? (complete vs partial)

Aggregation

- Allows us to treat a relationship set as an entity set for purposes of participating in other relationships
- Used when we need to model relationship with a relationship set (as opposed to an entity set)
- Notation: dashed box around relationship

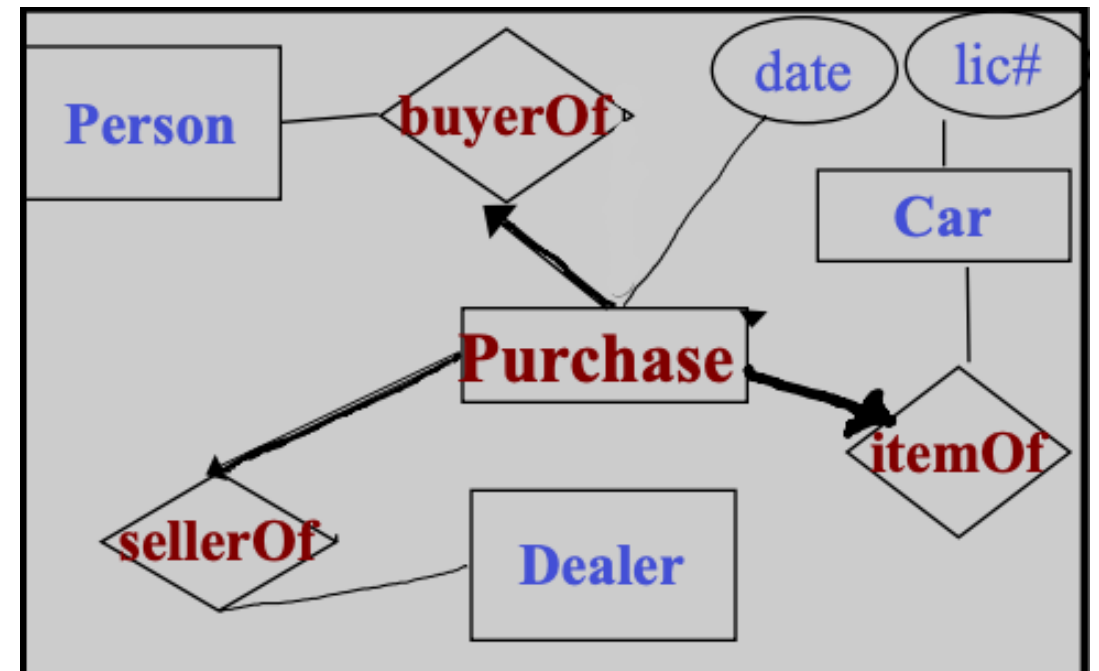
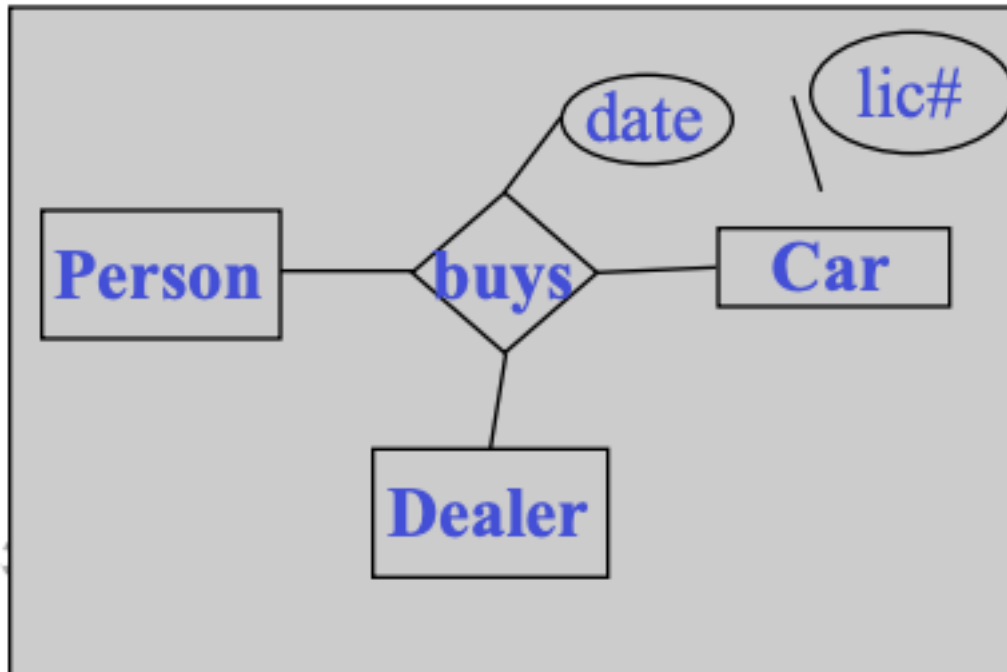
Aggregation

- Eg.

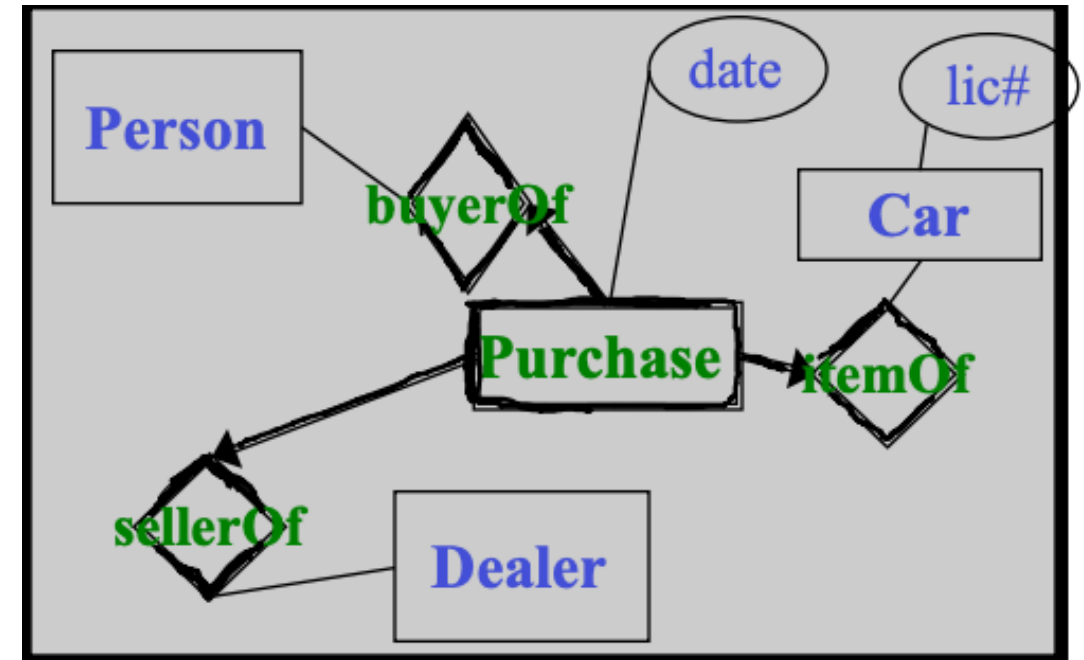
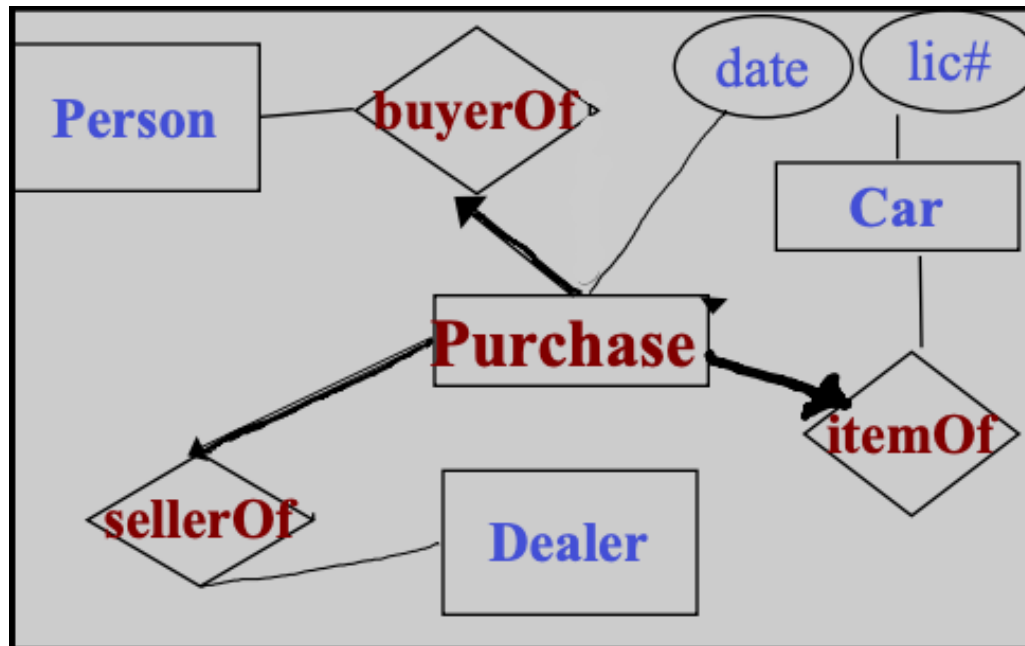


Reification

- Take a relationship R and think of it as a real entity, linked by simple functional binary relationships to the participants of R

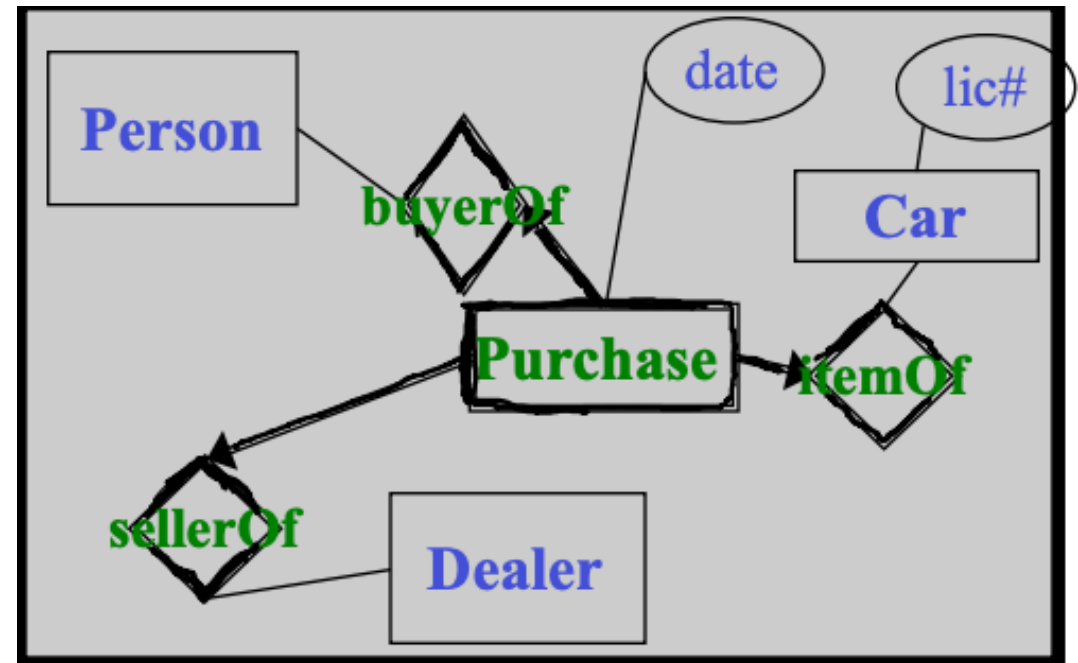
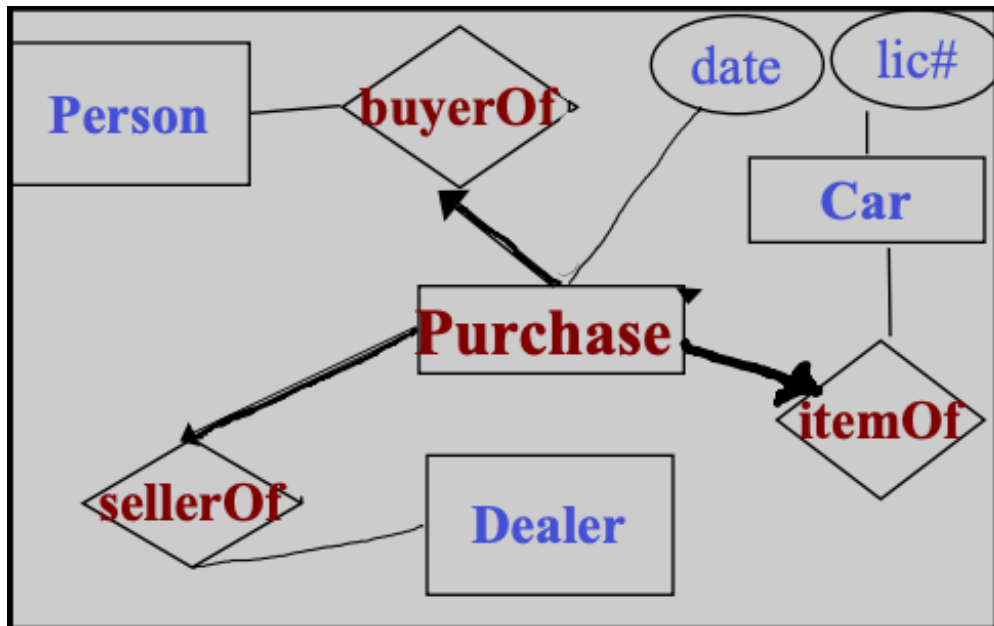


Reification



Reification

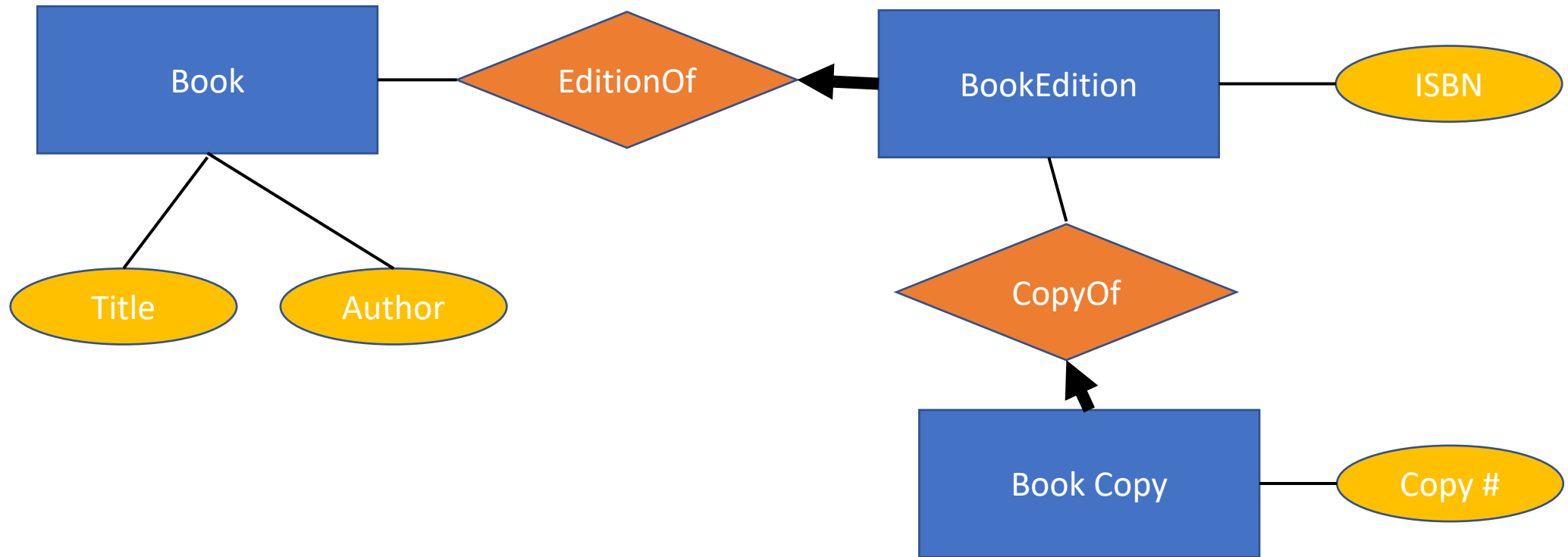
- Problem: No unique way to identify a purchase (ie. no key)
- Solution: make purchase a weak entity that depends on person, car, dealer (or alternatively, make a purchase id)



Manifestation

- Case of one entity being more abstract than another entity
- Eg:
 - Book -> Book Edition -> Book Copy
 - Lion King -> 2019 “Edition” -> Particular showing of it this morning
 - CS 336 -> Fall 2019 offering -> Class this afternoon
- Different than inheritance. Book Copy “is not a” Book Edition which “is not a” Book
- Some properties are still shared (eg. author, title)

Manifestation



- Professors have an SSN, a name, an age, a rank, and a research specialty.
- Projects have a project number, a sponsor name (e.g., NSF), a starting date, an ending date, and a budget.
- Graduate students have an SSN, a name, an age, and a degree program
- Each project is managed by one professor
- Each project is worked on by one or more professors
- Professors can manage and/or work on multiple projects.
- Each project is worked on by one or more graduate students
- When graduate students work on a project, a professor must supervise their work on the project.
- Graduate students can work on multiple projects, in which case they will have a (potentially different) supervisor for each one.
- Departments have a department number, a department name, and a main office.
- Departments have a professor (known as the chairman) who runs the department.
- Professors work in one or more departments, and for each department that they work in, a time percentage is associated with their job.
- Graduate students have one major department in which they are working on their degree.
- Each graduate student has another, more senior graduate student (known as a student advisor) who advises him or her on what courses to take.

