UML Class Diagrams

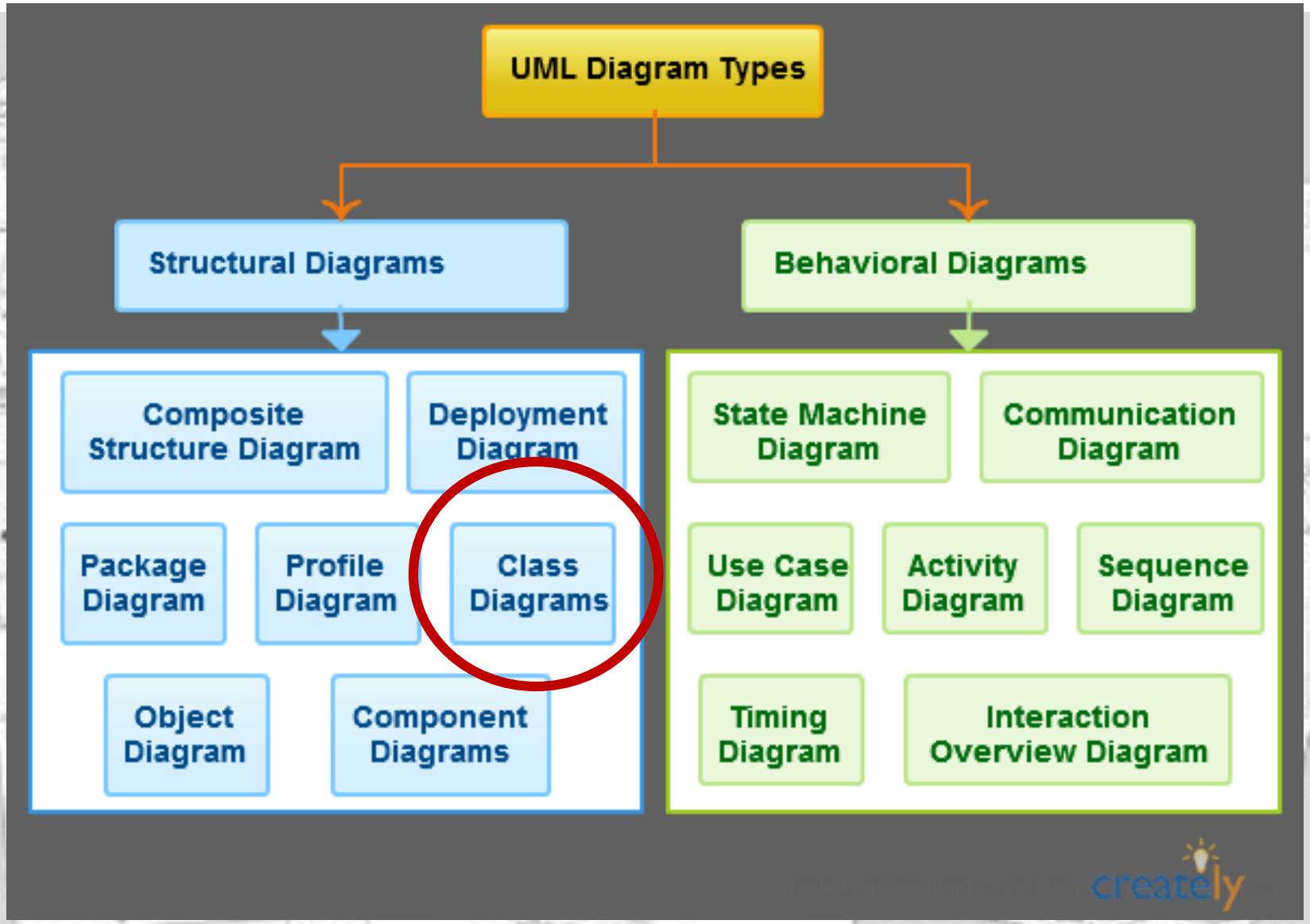
DS 501 Database Systems

Prof. Chandrashekar

Outline

- Using UML for database design
- UML notations

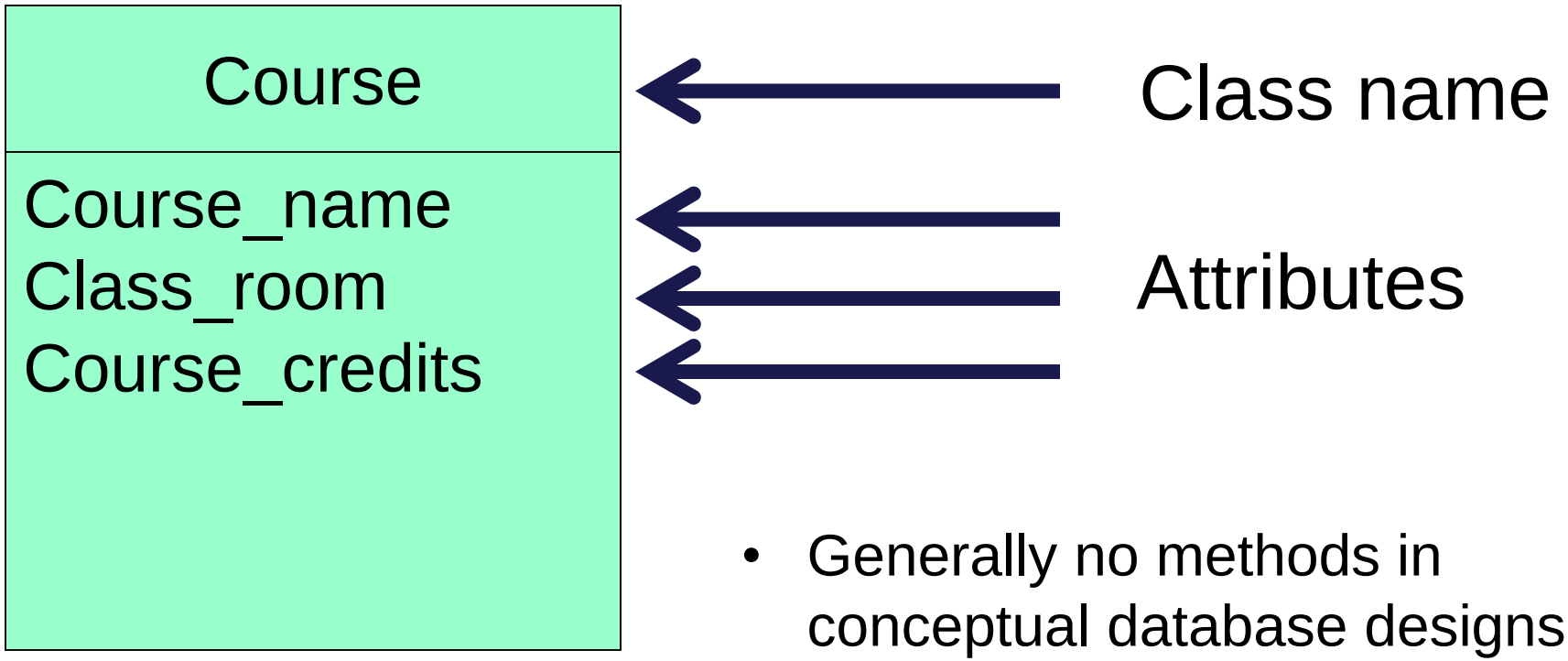
UML Components



Elements of Class Diagram

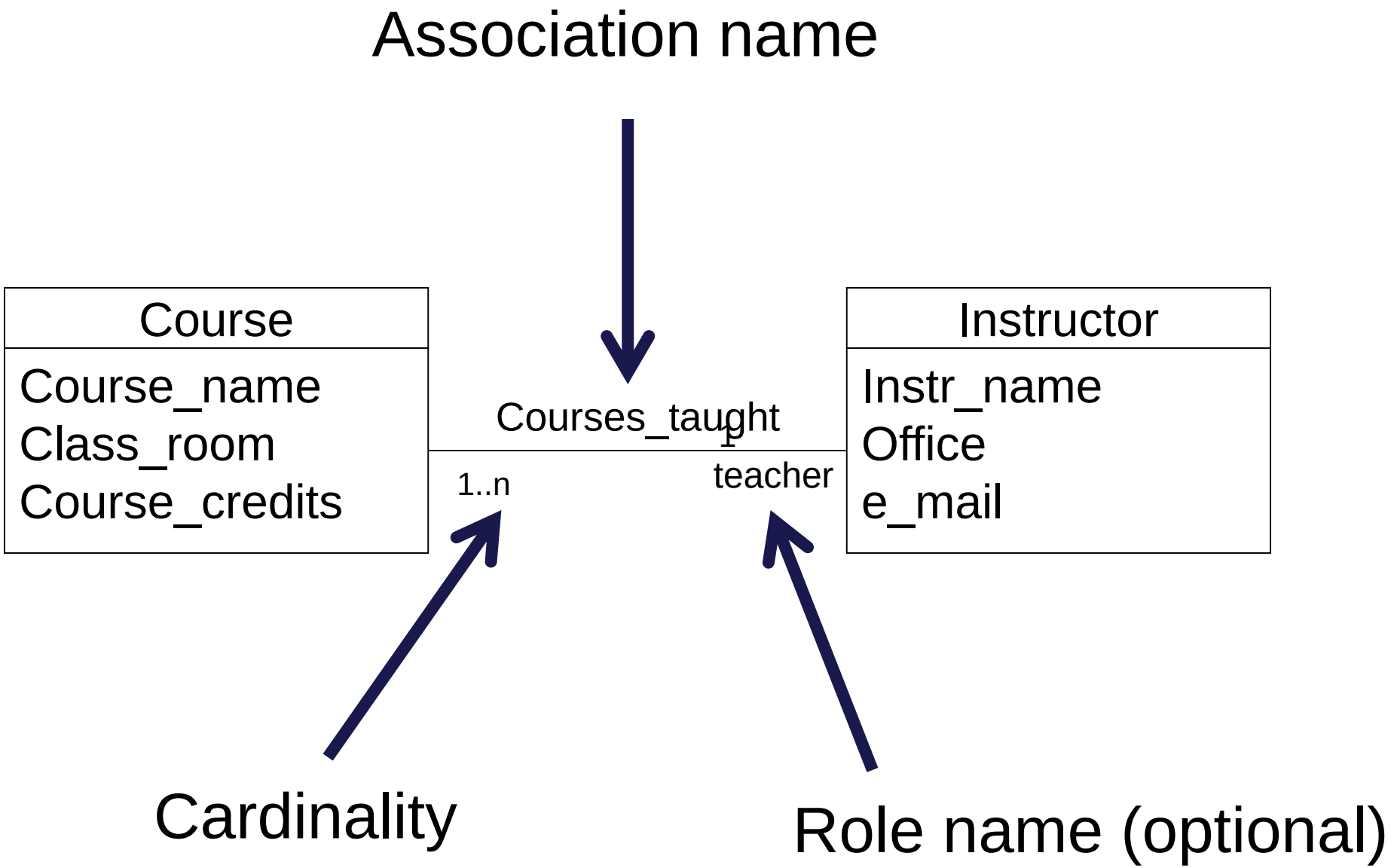
- Class
- Attribute
- Relationships
  - Association
  - Aggregation
  - Composition
  - Inheritance

Class and attributes



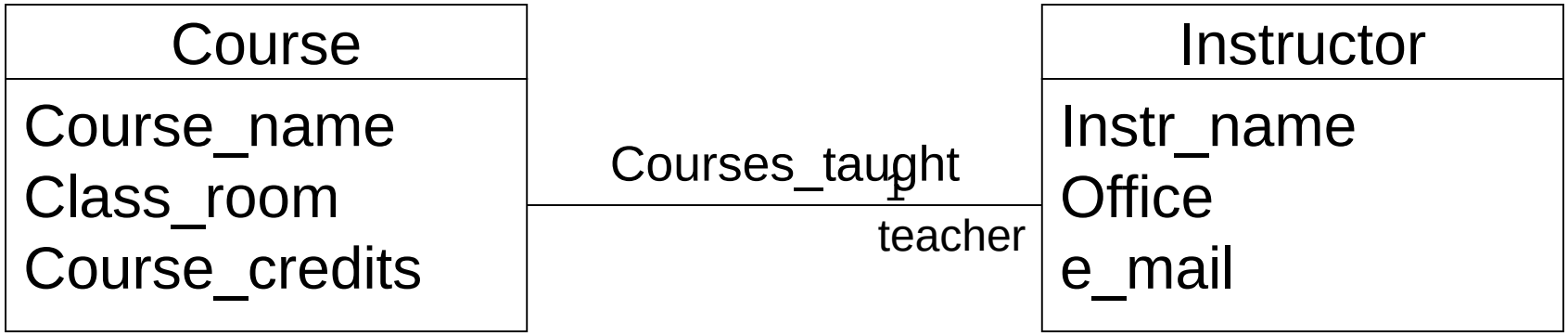
- Generally no methods in conceptual database designs
- Data types may be added as part of refinement

Association



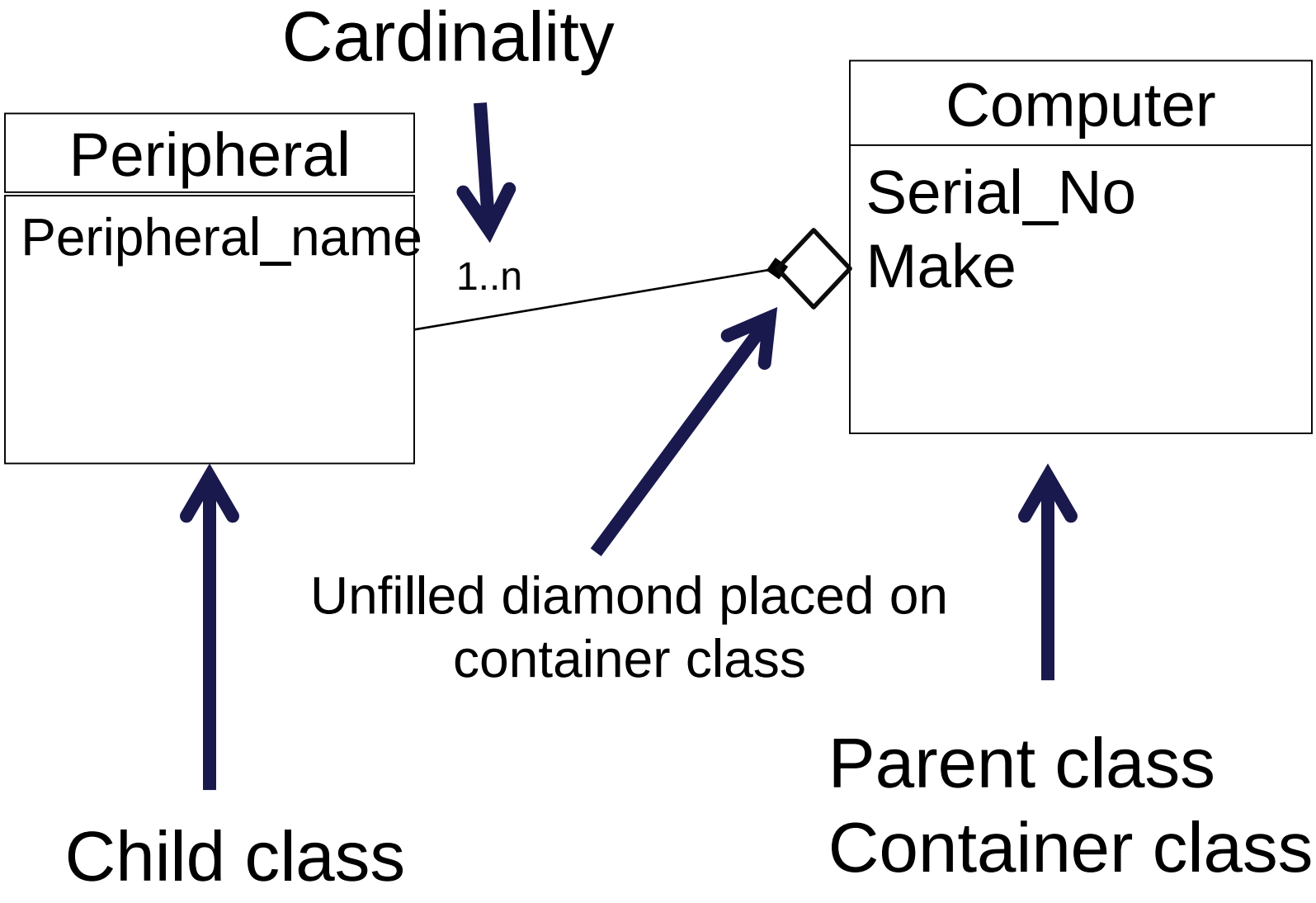
Association cardinality

- One:One
- One:Many
- Many:Many



What is the interpretation of cardinalities?

Aggregation (HAS-A)

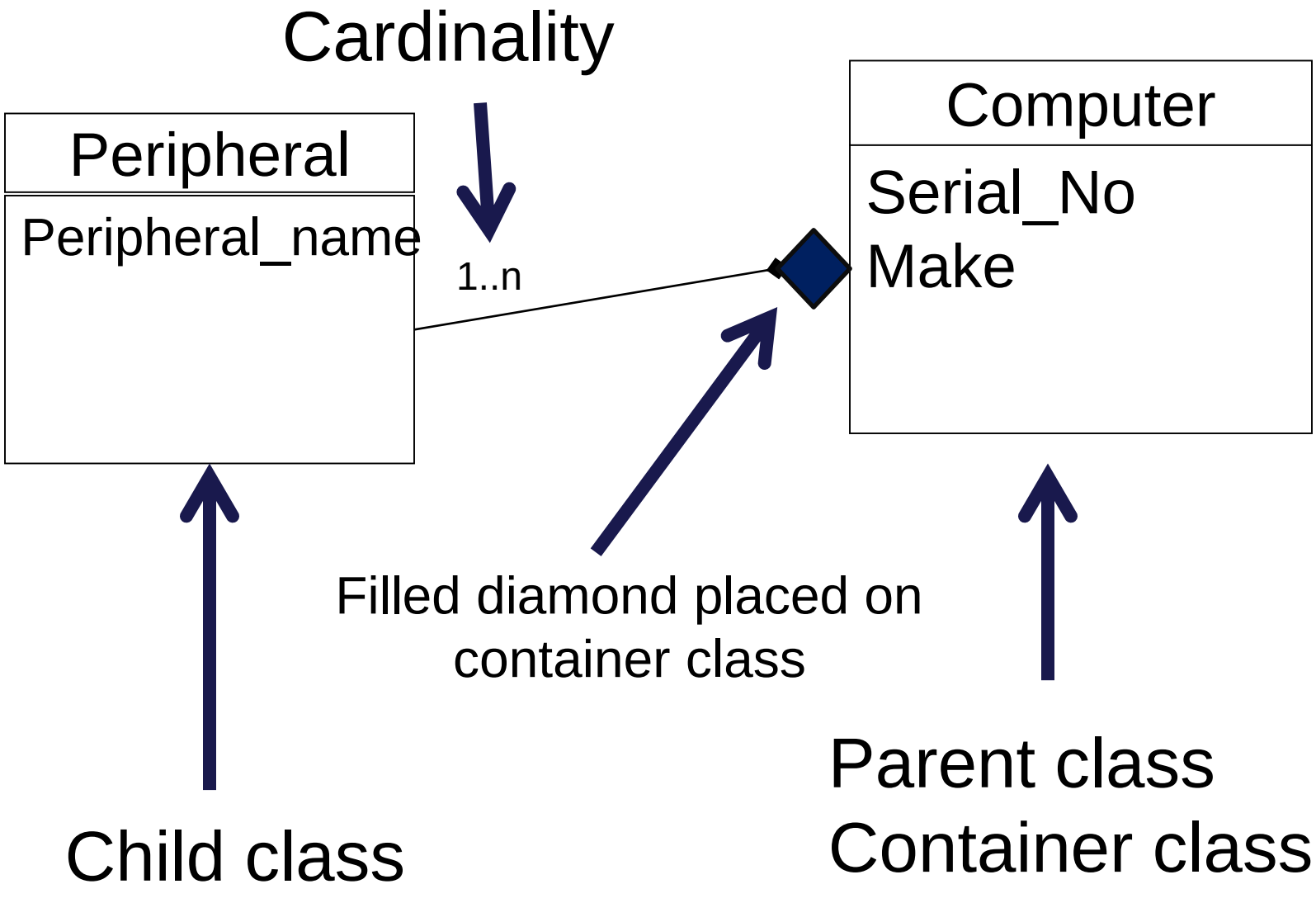


Computer has-a peripheral

Aggregation (HAS-A)

- Aggregation is similar to association
- The association name is fixed to "HAS-A"
- Cardinality cannot be "many:many"
- Child can exist independently until attached to a parent

Composition (IS-PART-OF)

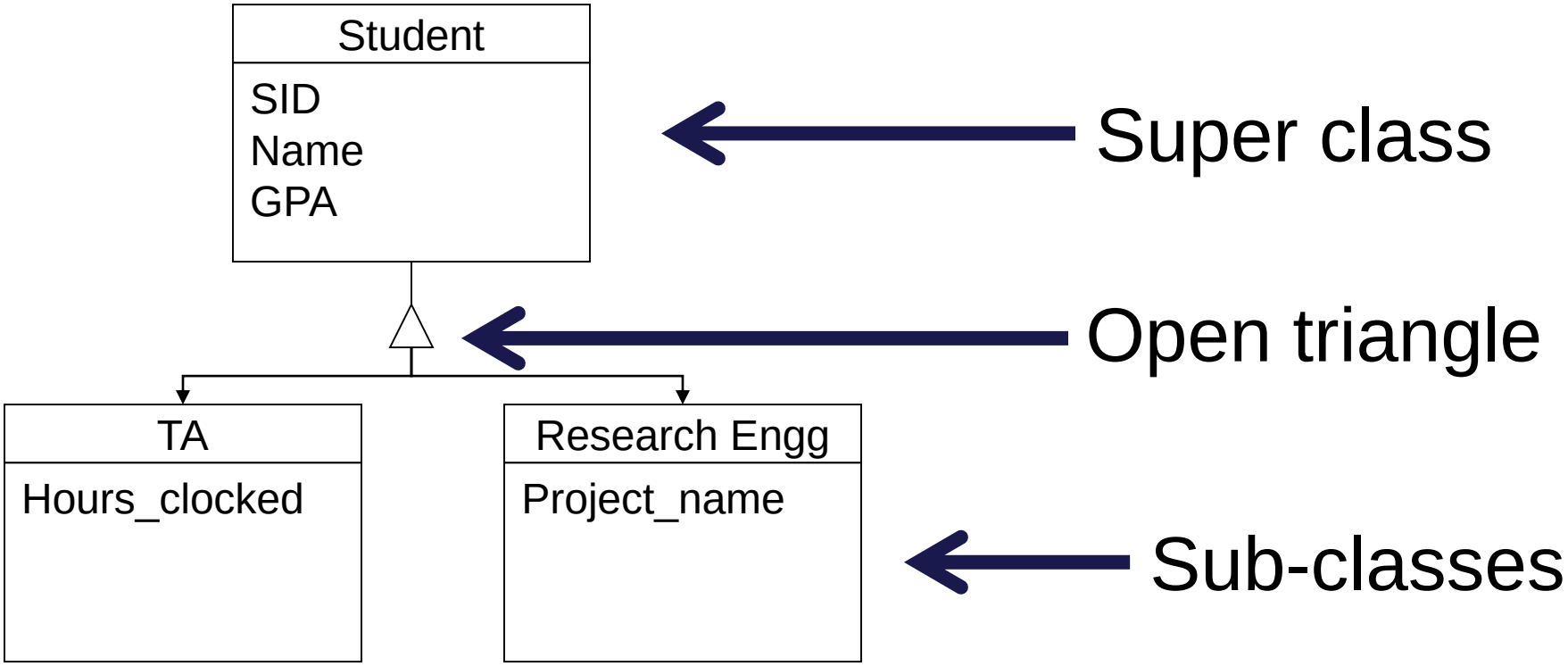


Peripheral is-part-of computer

Composition (IS-PART-OF)

- Aggregation is similar to composition
- The relationship name is fixed to "IS-PART-OF"
- Cardinality cannot be "many:many"
- Child CANNOT exist independently until attached to a parent ("existence dependency")
- Deletion semantics – All child objects get deleted when parent is deleted

Inheritance (IS-A)



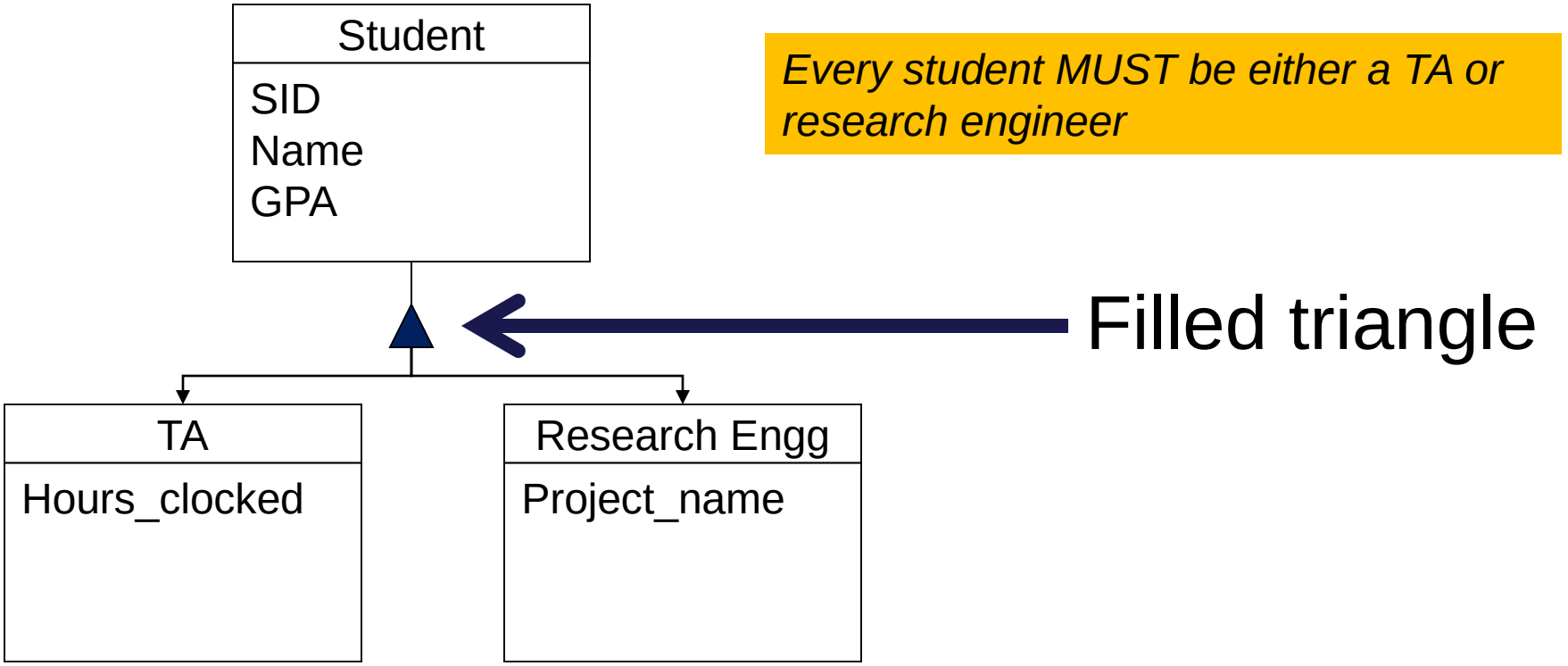
A student can be either a TA or research engineer or neither

Inheritance (IS-A)

- The relationship name is fixed to "IS-A"
- Child object inherits all attributes of parent objects
- Class hierarchy can be arbitrarily deep
- Cardinality
  - cannot be "one:many"
  - cannot be "many:many"
  - Cardinality can only be 1:1 or 1:0
  - So no need to (and must not) mention cardinality

Partitioned subclasses

- Parent object cannot exist stand-alone



Describe this in 100 words!