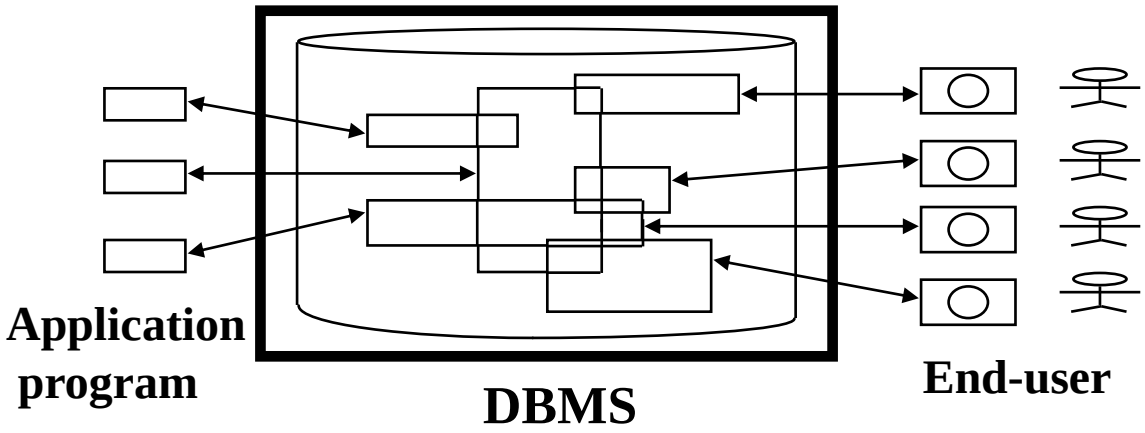
UML to Relations

Dr. Shaheen Khatoon



High-Level Database Design Model

- User-friendly (graphical) specification language
- Translated into model of DBMS

Unified Modeling Language (UML)

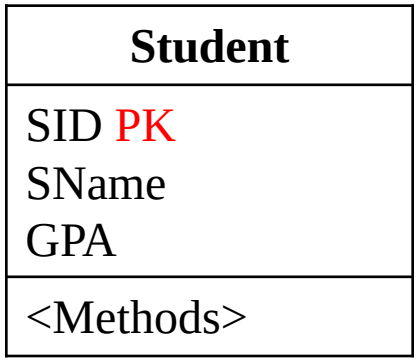
Data modeling subset

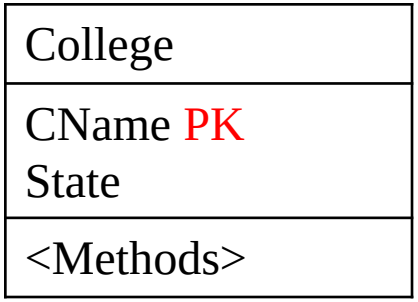
- 5 concepts
  (1) Classes
  (2) Associations
  (3) Association Classes
  (4) Subclasses
  (5) Composition & Aggregation
- Designs can be translated to relations automatically

  Provided every "regular" class has a key

UML to Relations: **Classes**

Every class becomes a relation; pk → primary key



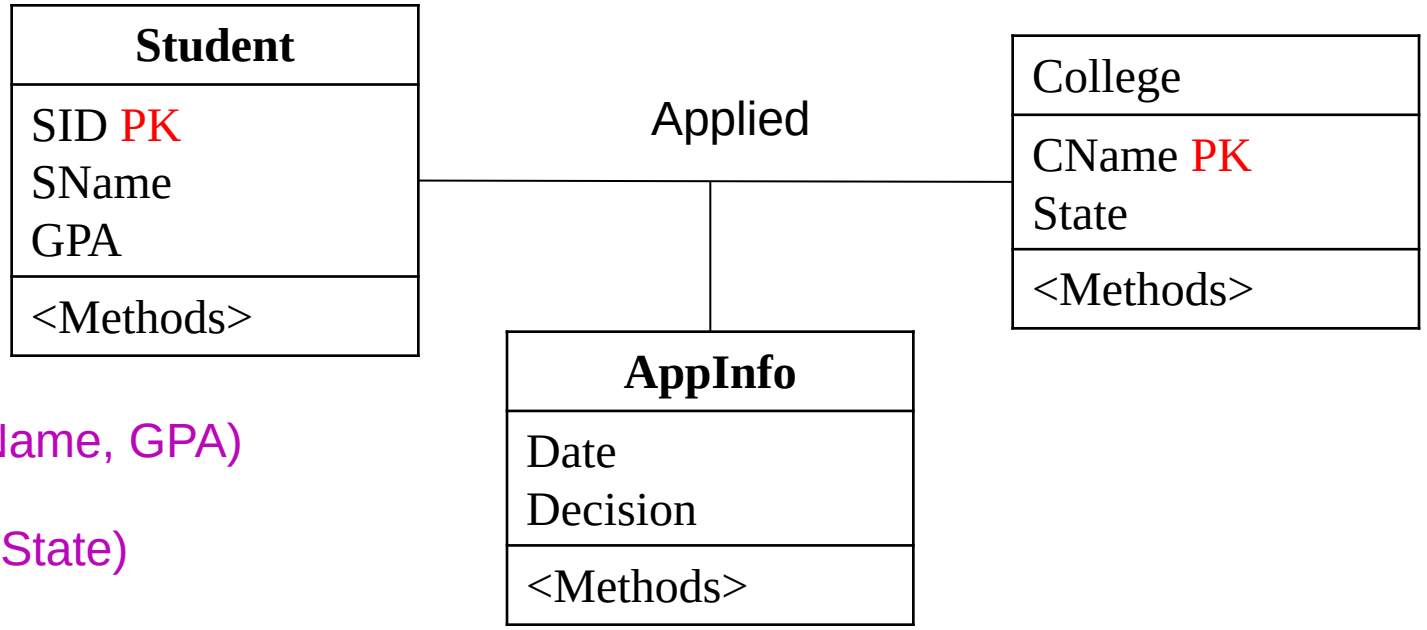


Student (SID, SName, GPA)

College(CName, State)

UML to Relations: **Associations**

Relation with key from each side



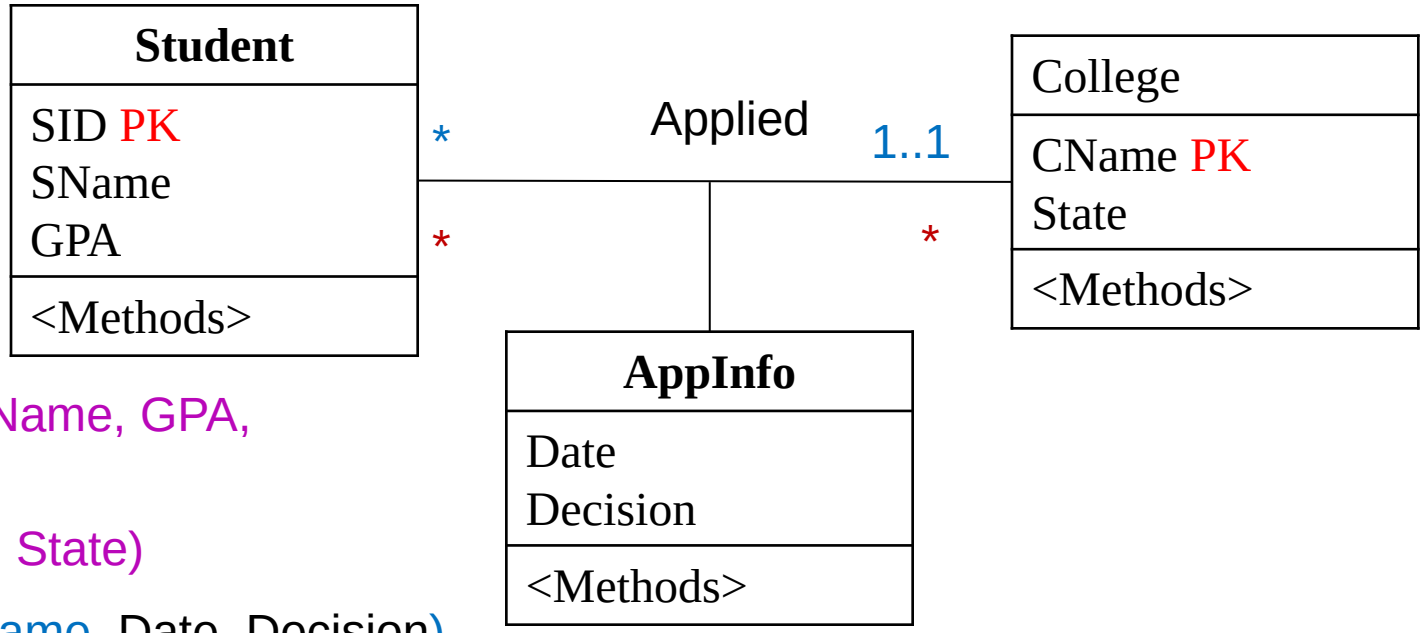
Student (SID, SName, GPA)

College(CName, State)

AppInfo(SID, CName, Date, Decision)

Keys for Association Relations

Depends on multiplicity



Student (SID, SName, GPA, CNAme)

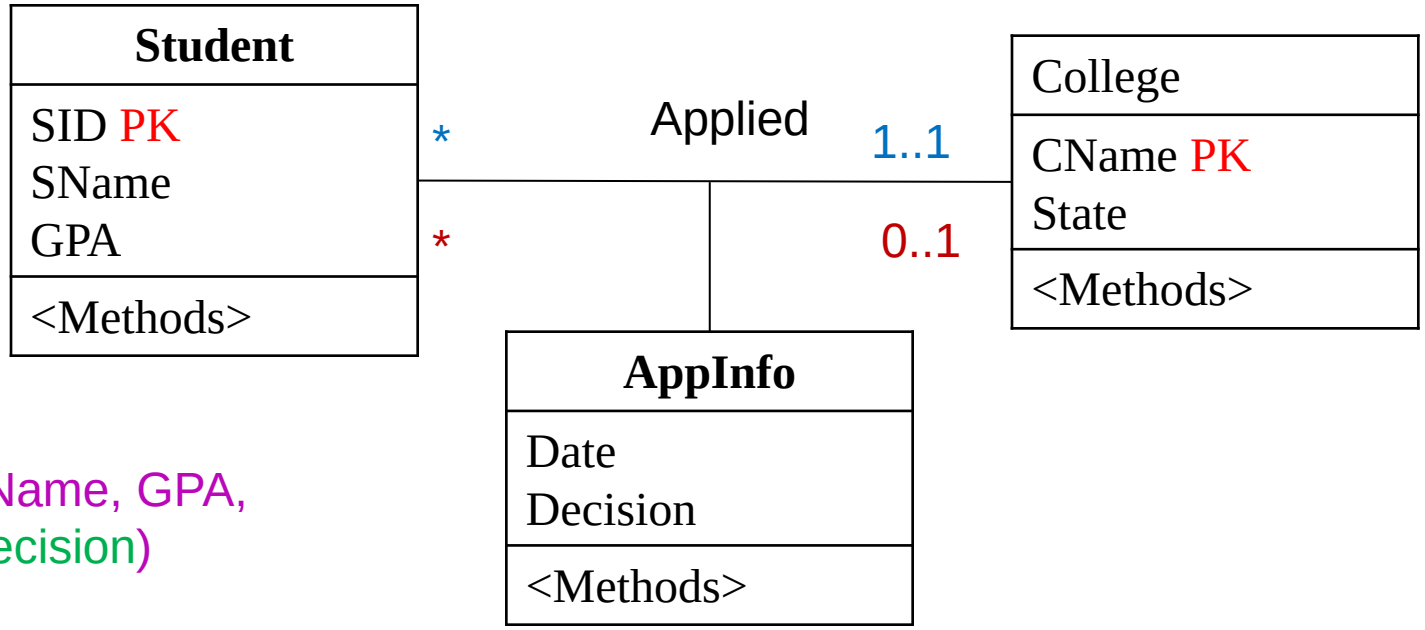
College(CName, State)

AppInfo(SID, CName, Date, Decision)

AppInfo(SID, CName, Date, Decision)

Association Relation Always Needed?

Depends on multiplicity



Student (SID, SName, GPA,
CName, Date, Decision)

College(CName, State)

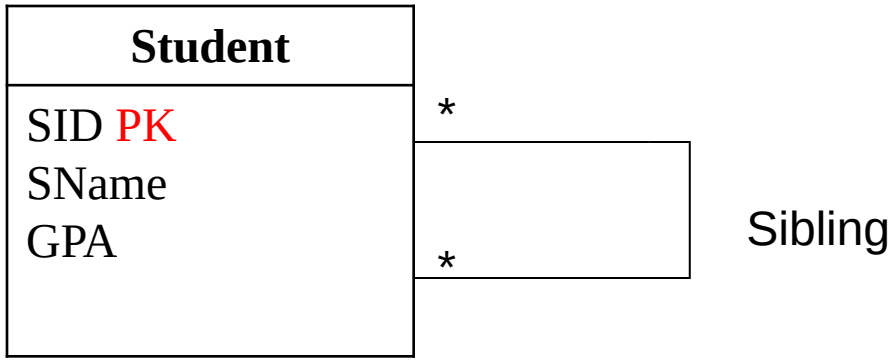
- Suppose we had 0..2 on the right-hand side, so students can apply to up to 2 colleges? Is there still a way to "fold in" the association relation in this case, or must we have a separate Applied relation?

Explanation

We might create relation
Student(sID,sName,GPA,cName1,cName2),
assuming null values are allowed.

We can similarly fold in any 0..N or M..N multiplicity, although the bigger N is, the less sense this translation makes.

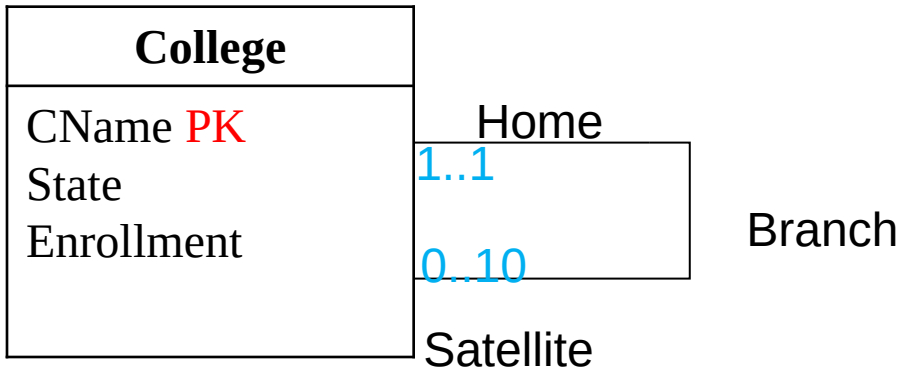
Self-Associations



Student (SID, SName, GPA)

Sibling (SID1, SID)

Self-Associations



College (CName, State, Enrollment

Branch (Home, Satellite)

CName

UML Data Modeling: 5 concepts

(1) Classes

(2) Associations

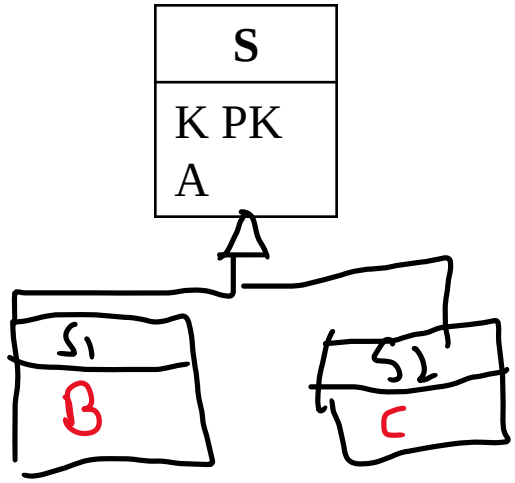
(3) Association Classes

(4) Subclasses

(5) Composition & Aggregation

Subclasses

1) Subclass relations contain superclass key + specialized attrs.
2) Subclass relations contain all attributes
3) One relation containing all superclass + subclass attrs.



1: S(K, A), S1 (K, B) S2(K, C)

2: S(K, A), S1 (K, A, B) S2(K, A, C)

3: (K, A, B, C)

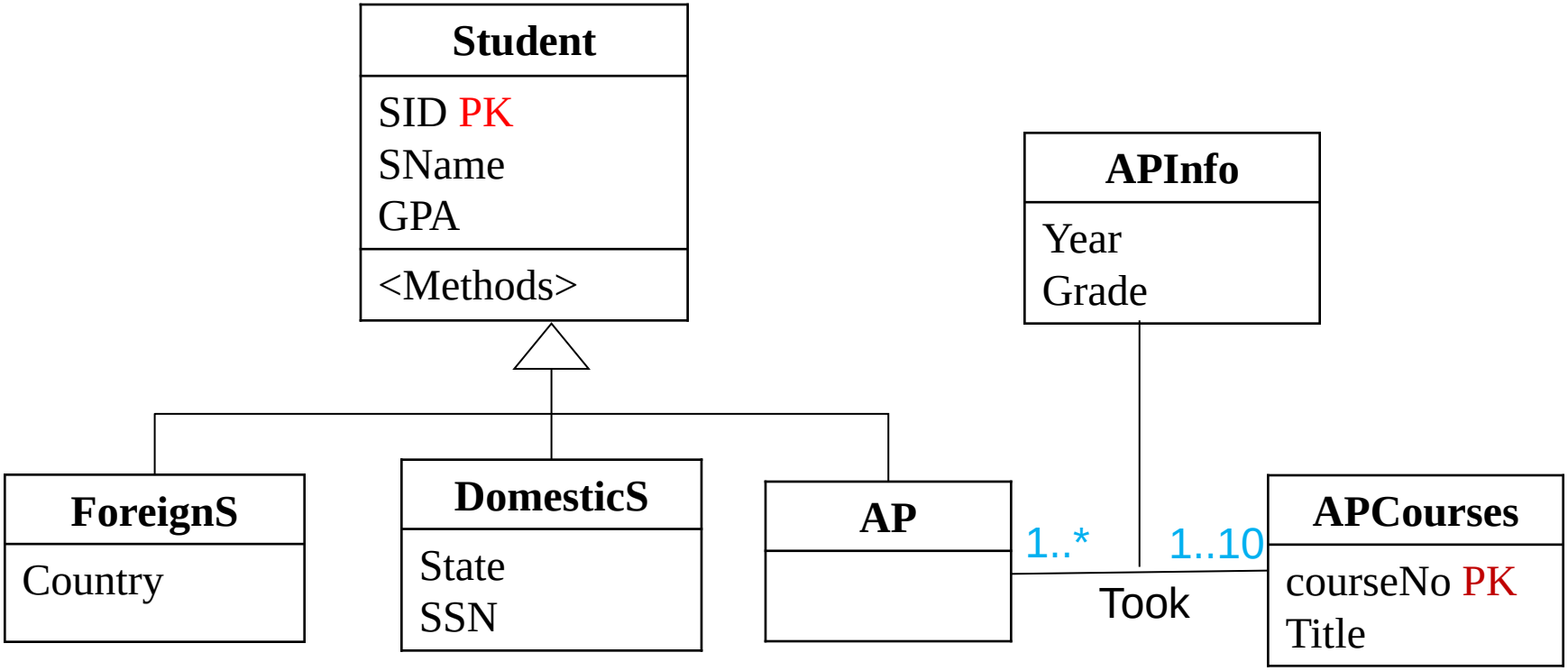
Best translation may depend on properties

Heavily overlap design 3

Disjoint complete: Design 2

S1 (K, A, B) S2(K, A, C)

Subclasses – Example: Using Design 1



Student (SID, SName, GPA)
ForeignS (SID, Country)
DomesticS (SID, State, SSN)
AP(SID)
APCourse(CourseNo, Title)
AppInfo(SID, CourseNo, Year, Grade)

UML Data Modeling: 5 concepts

(1) Classes

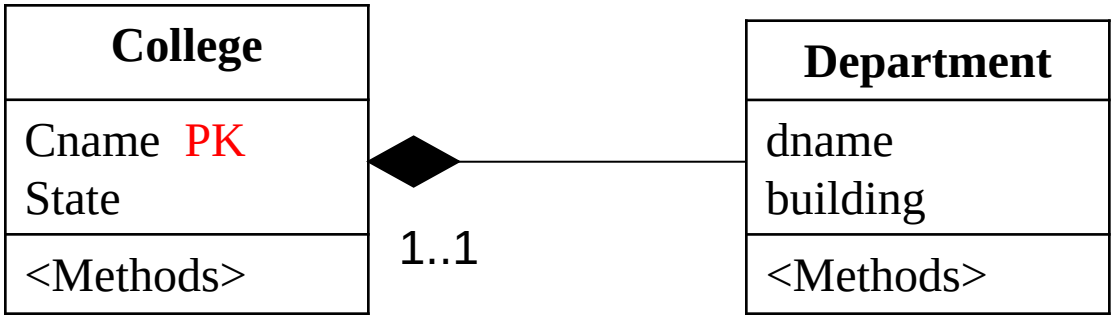
(2) Associations

(3) Association Classes

(4) Subclasses

(5) Composition & Aggregation

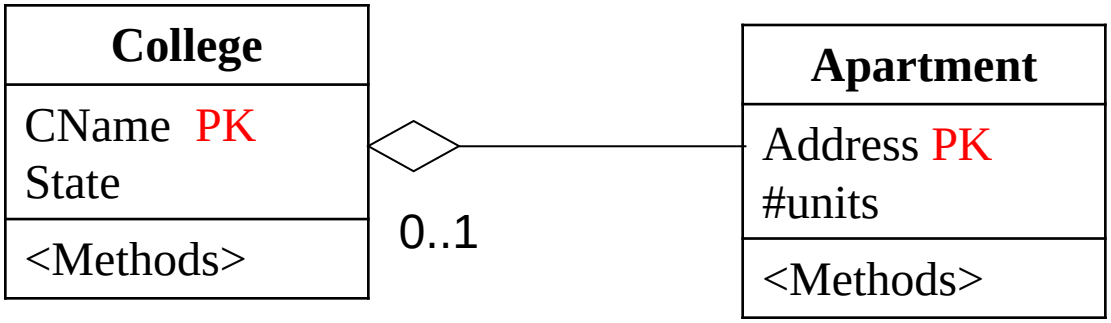
Composition & Aggregation



College(CName, State)

Department(dname, building, CName)

Aggregation



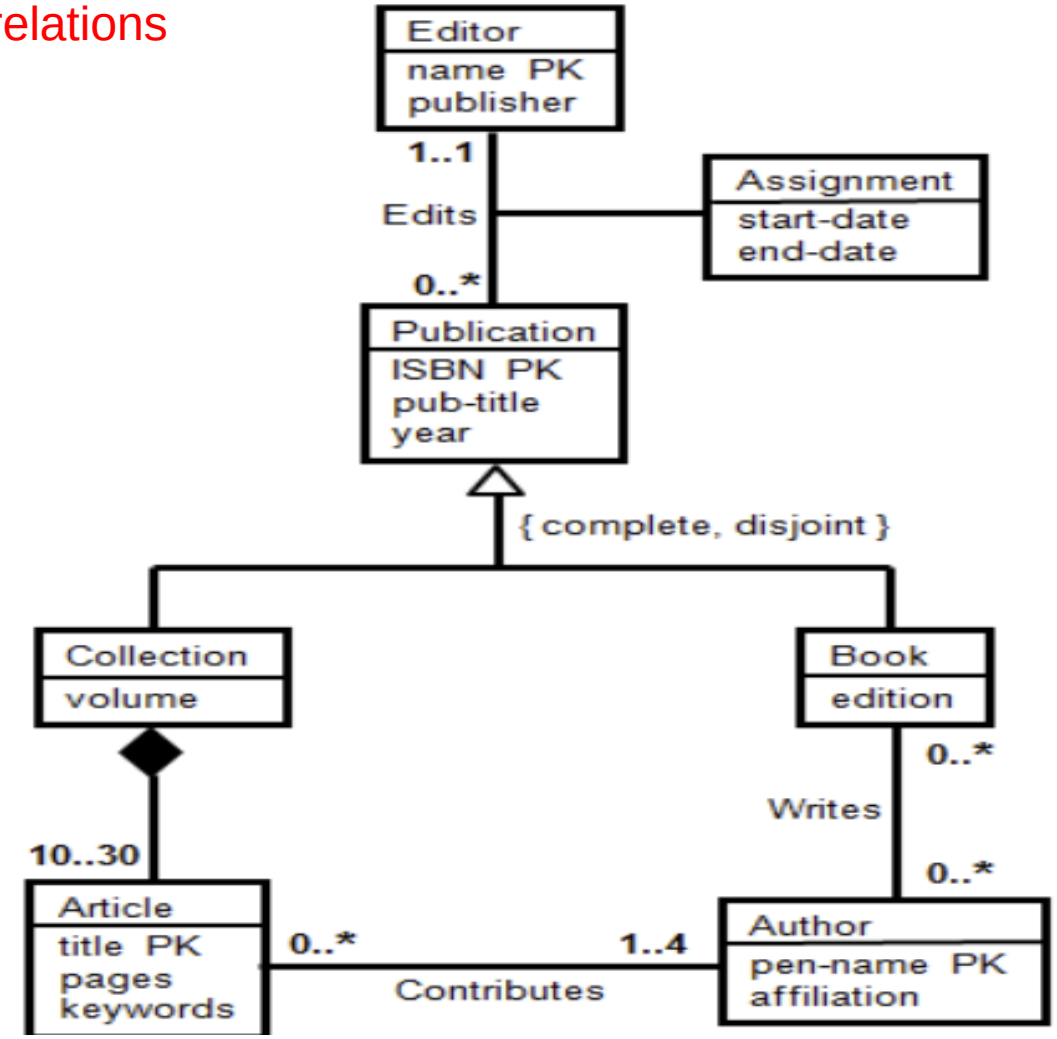
College(CName, State)

Apartment(Address, #units, CName)

Null

Homework

Translate this UML into relations



UML: High-Level Database Design Model

- User-friendly graphical specification language
- Designs translated to relations automatically

Summary

Higher-Level Database Design

- *Unified Modeling Language* (UML)
  Data modeling subset
- Graphical
- 5 concepts
  (1) Classes
  (2)Associations
  (3)Association Classes
  (4)Subclasses
  (5)Composition & Aggregation

❖ Can be translated to relations automatically