

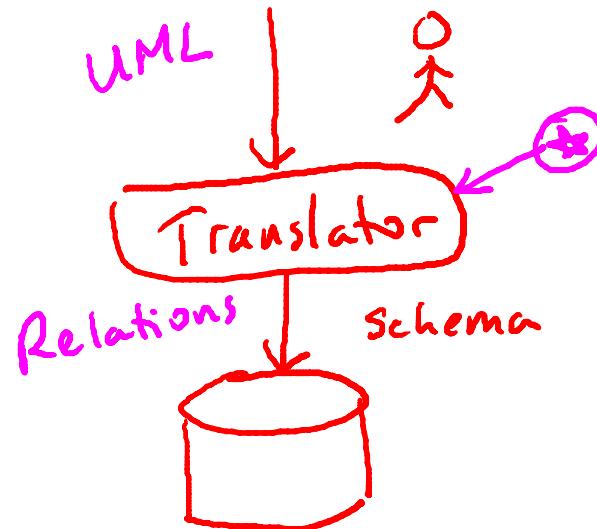
UML

UML to
Relations



High-Level Database Design Model

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



Unified Modeling Language (UML)

Data modeling subset

■ 5 concepts

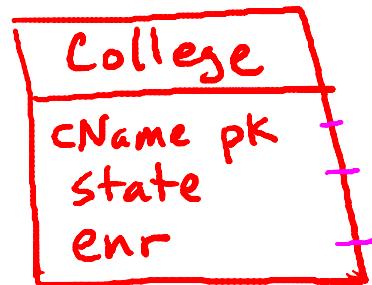
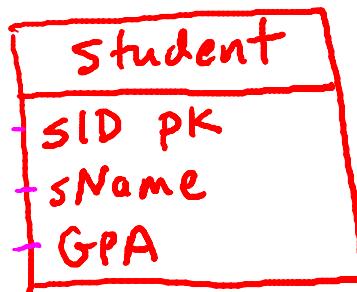
- ✓ (1) Classes
- ✓ (2) Associations
- ✓ (3) Association Classes
- ✓ (4) Subclasses
- ✓ (5) Composition & Aggregation

■ 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



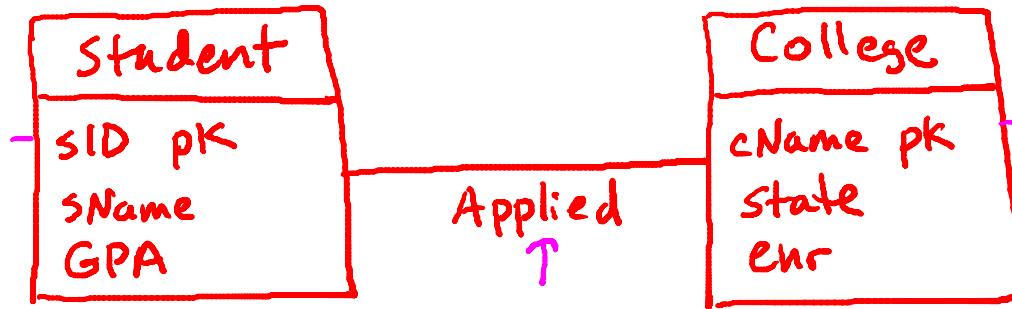
↳ $\text{Student}(\underline{sID}, \text{sName}, \text{GPA})$
 $\text{College}(\underline{\text{cName}}, \text{state}, \text{enr})$

UML Data Modeling: 5 concepts

- ✓(1) Classes
- (2) Associations
- (3) Association Classes
- (4) Subclasses
- (5) Composition & Aggregation

UML to Relations: Associations

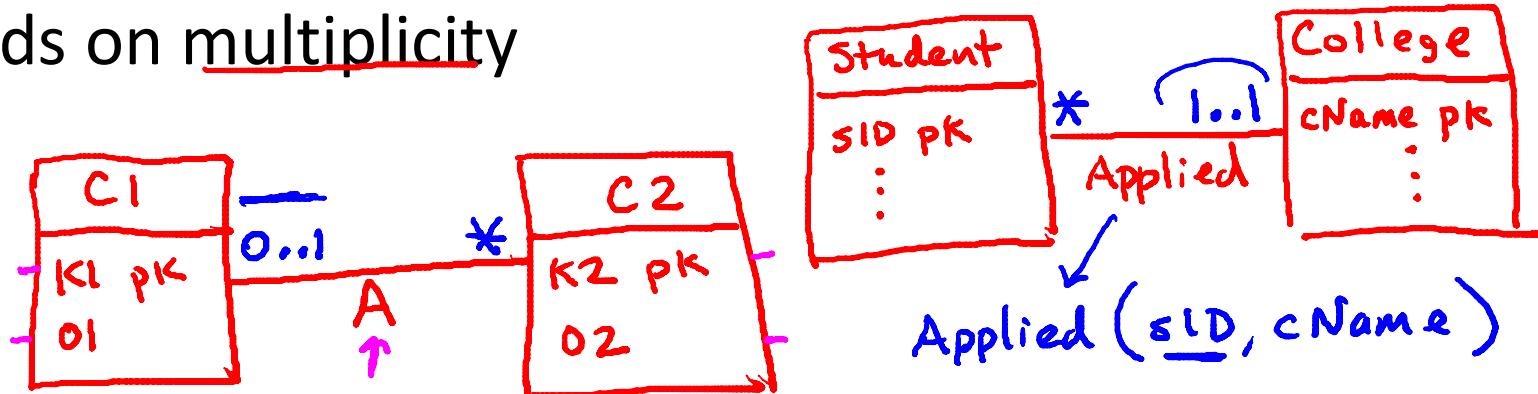
Relation with key from each side



- Student(...)
- College(...)
- Applied (s_{ID} , c_{Name}) \leftarrow

Keys for Association Relations

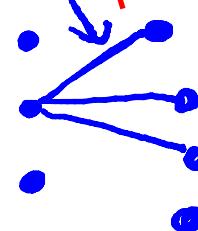
Depends on multiplicity



$C1(K1, O1)$

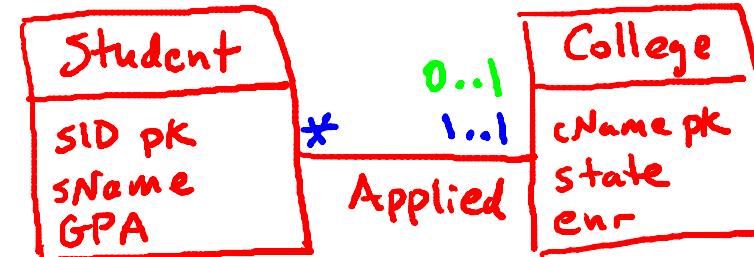
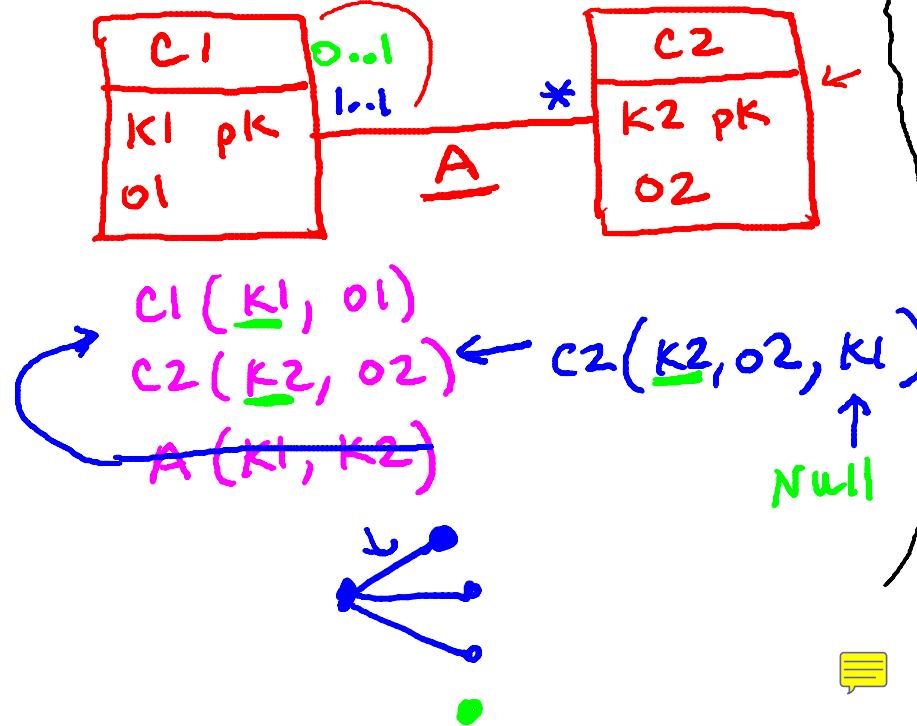
$C2(K2, O2)$

$A(K1, \underline{K2})$



Association Relation Always Needed?

Depends on multiplicity



Annotations:

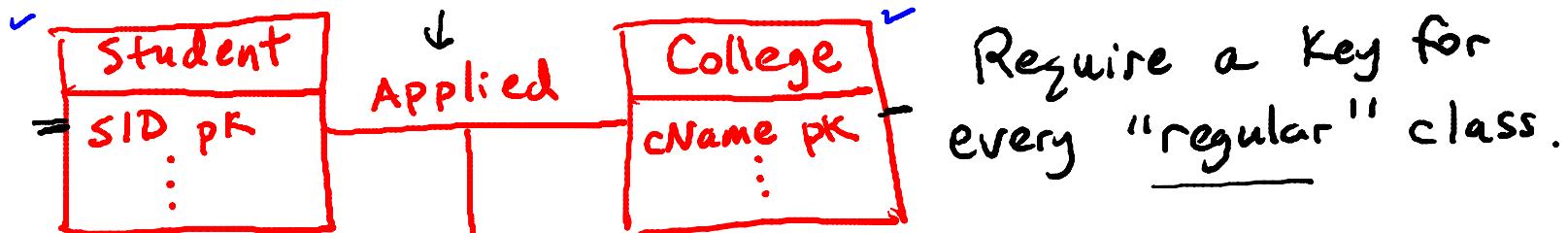
- $\text{Student}(s_{ID}, s_{Name}, GPA) \rightarrow c_{Name}$
- $\text{College}(c_{Name}, state, enr) \xrightarrow{\text{Null}}$
- ~~$\text{Applied}(s_{ID}, c_{Name})$~~

UML Data Modeling: 5 concepts

- ✓(1) Classes
- ✓(2) Associations
- (3) Association Classes
- (4) Subclasses
- (5) Composition & Aggregation

UML to Relations: Association Classes

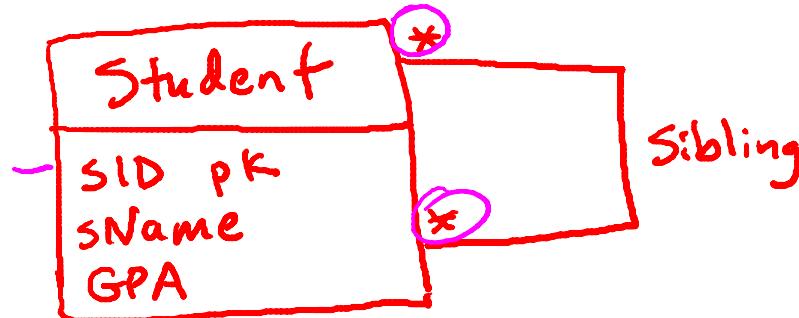
Add attributes to relation for association



Determining keys
"Folding"

- ✓ Student(sID, ...)
- ✓ College(cName, ...)
- ✓ Applied(sID, cName, date, decision)

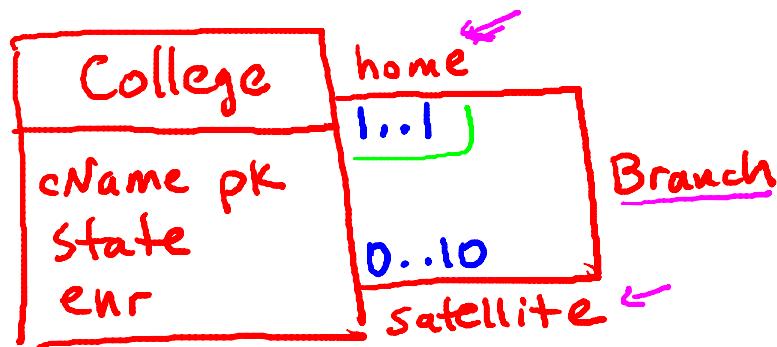
Self-Associations



$\text{Student}(\underline{\text{SID}}, \text{SName}, \text{GPA})$

$\text{Sibling}(\underline{\text{SID1}}, \underline{\text{SID2}}) \leftarrow$

Self-Associations



College (cName, state, enr)

Branch (home, satellite)
(cName's) ↑

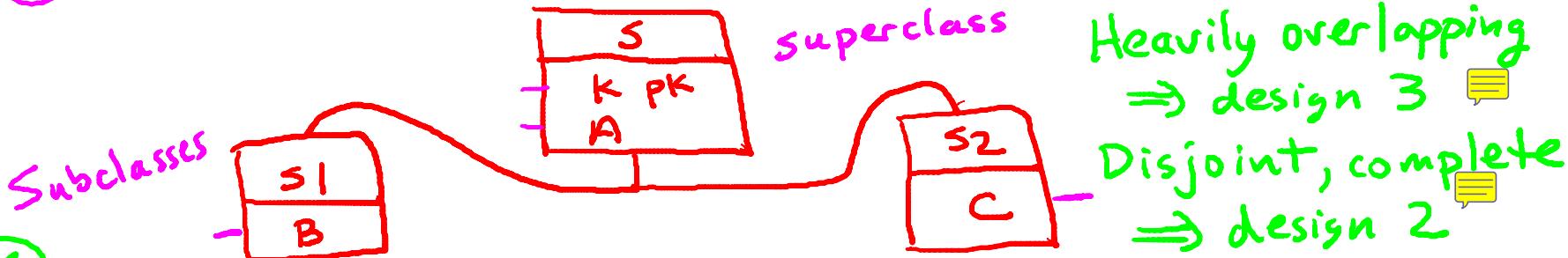
UML Data Modeling: 5 concepts

- ✓(1) Classes
- ✓(2) Associations
- ✓(3) Association Classes
- (4) Subclasses
- (5) Composition & Aggregation



Subclasses

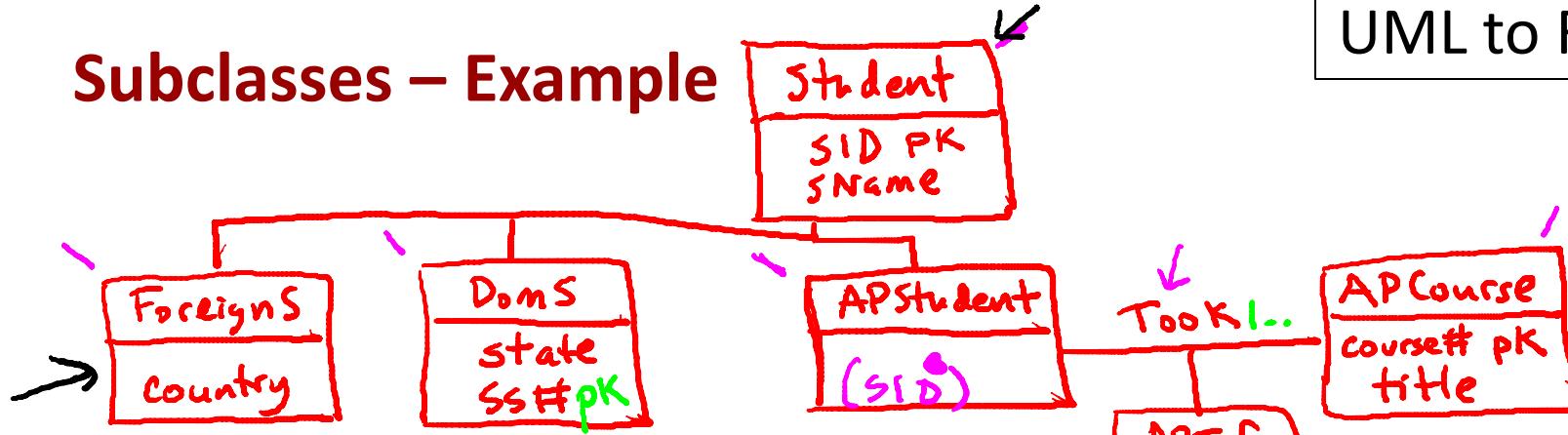
- 1) Subclass relations contain superclass key + specialized attrs. ↗
- 2) Subclass relations contain all attributes
- 3) One relation containing all superclass + subclass attrs.



④ Best translation may depend on properties

- (1) $s(K, A)$ $s1(K, B)$ $s2(K, C)$ ↗
- (2) ~~$s(K, A)$~~ $s1(K, A, B)$ ↙ $s2(K, A, C)$ ↗
- (3) $s(K, A, B, C)$ ↗

Subclasses – Example



Student (sid, sName)

ForeignS (sid, country)

Doms (sid, state, ss#)

APStudent (sid)

APCourse (course#, title)

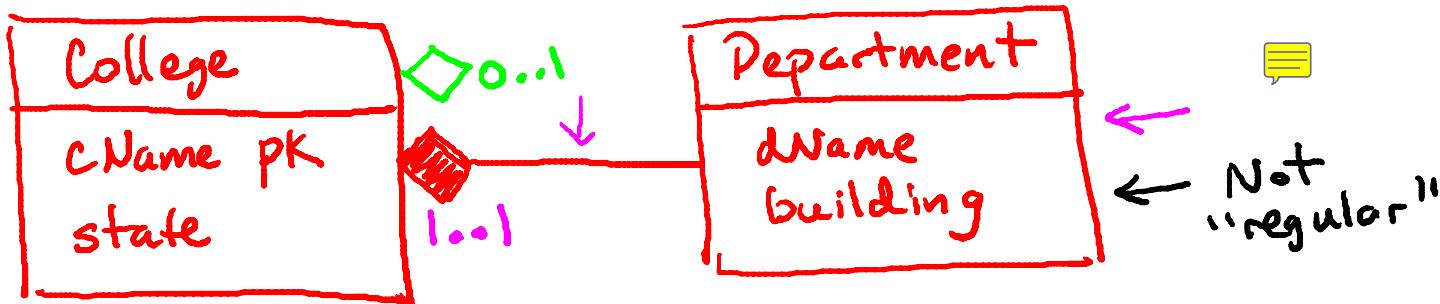
Took (sid, course#, year, grade)

Keys for
"regular"
classes

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)`

↑
Null

UML: High-Level Database Design Model

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