UML

UML to Relations

Jennifer Widom

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(<u>sID</u>, sName, GPA)

College(<u>cName</u>, state, 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(sID, cName)

Keys for Association Relations

Depends on multiplicity

Applied (sID, cName)

C1(K1, O1)

C2(K2, O2)

A(K1, K2)

Association Relation Always Needed?

Depends on multiplicity

C1(K1, O1)
C2(K2, O2) ← C2(K2, O2, K1)
~~A(K1, K2)~~

Null

Student(SID, SName, GPA, CName)
College(CName, state, enr)
~~Applied(SID, CName)~~

Null

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

Require a key for every "regular" class.

Determining keys
"Folding"

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

Self-Associations

Student(sID, sName, GPA)

Sibling(sID1, sID2)

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.

Heavily overlapping
⇒ design 3
Disjoint, complete
⇒ design 2

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

Keys for "regular" classes

Student(SID, sName)

ForeignS(SID, country)

DomS(SID, state, SS#)

APStudent(SID)

APCourse(course#, title)

Took(SID, course#, 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)

Null

UML: High-Level Database Design Model

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