Representing Inheritance Associations in a Relational Database

Inheritance provides a convenient mechanism for modelling types of things. All attributes common to every type of the item are stored in a superclass. Attributes specific to each subtype are stored in subtype classes.

Relational databases do not natively support inheritance association and so these must be implemented using available relational techniques.

**Method 1): Implement separate tables for each subclass**.

This is done by adding the inherited attributes from the superclass to each subclass e.g.

Furniture (Superclass)

Cost

Manufacturer

Finish Type

/ \

/ \

/ \

Table Chair (Subclasses)

Cost 🡨🡪 Cost

Manufacturer 🡨🡪 Manufacturer

Finish Type 🡨🡪 Finish Type

Size Style

Colour

Advantages:

- No wasted storage. Only required columns are stored in each table

- Each subtype is distinct and can be accessed directly

Disadvantages:

- The basic inheritance association concept is lost

- Adding new subtypes involves creating extra tables

- Can't process Furniture data as a single set

**Method 2): Implement the superclass as a table**

Furniture

Code No

Finish Type (Common Fields)

Cost

Manufacturer

Size (Table specific)

Shape

Style (Chair specific)

Colour

ItemType (Subtype ID)

The entire hierarchy is implemented as a single Furniture table. The table will have NULL fields depending on what it is representing. E.G. a Furniture entry representing a Chair will not have a Size or Shape value. Instead the table will say 'Null' or 'n/a' in its place. Each entry in the table will have a column for each field in Furniture, but only relevant fields will have values.

Advantages:

- Inheritance association is preserved

- Subtypes are easily identified (SELECT DISTINCT ItemType FROM Furniture)

- New subtypes are easily added

Disadvantages:

- Wasted storage space (very small amounts generally)

TO isolate details of specific subtypes, new VIEWS can be created e.g.

CREATE NEW CHAIRS AS

SELECT CodeNo, FinishType, Cost, ManuFacturer, Style,

FROM Furniture

WHERE ItemType = "Chair";

This creates a view using only the details of a single subtype