

CHUYỂN ĐỔI MÔ HÌNH THỰC THỂ - QUAN HỆ THÀNH MÔ HÌNH HƯỚNG ĐỐI TƯỢNG

HOÀNG QUANG

Abstract. Entities represent classes of real-world objects. As a result, problem of converting Entity-Relationship Model into Object-Oriented Model is taken shape naturally. In order to solve this problem, some conventions are concerned. Classes are known as abstract data types and each class is available to have multi-valued attributes. The translation from Entity-Relationship Model to Object-Oriented Model means to convert entities and relationships into object classes.

Tóm tắt. Một tập thực thể của mô hình thực thể - quan hệ được xem như một lớp các đối tượng trong thế giới thực. Vì vậy việc chuyển đổi mô hình thực thể - mối quan hệ thành mô hình hướng đối tượng là khá thích hợp. Với qui ước ban đầu rằng các lớp bên trong mô hình hướng đối tượng là cho phép sử dụng các thuộc tính đa trị và xem lớp như là kiểu dữ liệu trừu tượng thì việc chuyển đổi này thực chất là quá trình chuyển đổi các tập thực thể và các mối quan hệ bên trong mô hình thực thể - mối quan hệ thành các lớp của mô hình hướng đối tượng.

1. GIỚI THIỆU

Hiện nay mô hình cơ sở dữ liệu hướng đối tượng đã xuất hiện như một giải pháp nhằm giải quyết sự phức tạp trong việc mô hình hóa thế giới thực. Mô hình dữ liệu hướng đối tượng cho phép dùng lại các dữ liệu và các phương pháp thông qua tính kế thừa, cho phép xây dựng các đối tượng phức tạp, và cho phép định danh các đối tượng một cách độc lập. Tuy nhiên, để các hệ quản trị cơ sở dữ liệu hướng đối tượng có thể sử dụng và kế thừa được phần dữ liệu trên các cơ sở dữ liệu quan hệ, vấn đề được đặt ra là cần có một phương pháp để chuyển đổi mô hình dữ liệu quan hệ thành mô hình hướng đối tượng. Việc làm này đồng thời còn hỗ trợ các hệ thống multi-database là các hệ thống cho phép người sử dụng thực hiện các thao tác truy cập đồng thời trên các cơ sở dữ liệu quan hệ và cơ sở dữ liệu hướng đối tượng.

Việc chuyển đổi mô hình dữ liệu quan hệ thành mô hình hướng đối tượng là có thể thực hiện được khi một mô hình dữ liệu quan hệ được thiết kế từ mô hình thực thể - mối quan hệ và có giải pháp cho việc chuyển đổi mô hình thực thể - mối quan hệ thành mô hình hướng đối tượng.

Trong thực tế khá nhiều hệ thống quen thuộc được thiết kế xuất phát từ mô hình thực thể - quan hệ. Các thành phần chính trong mô hình này là các tập thực thể và các mối quan hệ. Mỗi tập thực thể bao gồm các thực thể có liên quan với nhau, mà mỗi thực thể được xác định thông qua một thể hiện của tập các thuộc tính. Mỗi mối quan hệ trong mô hình này lại biểu thị quan hệ giữa các thực thể của các tập thực thể. Mỗi mối quan hệ này cũng có thể chứa các thuộc tính của riêng nó. Mỗi quan hệ giữa các tập thực thể có thể là mối quan hệ is-a (quan hệ kế thừa), mối quan hệ phản xạ (mối quan hệ giữa các thực thể trong cùng một tập thực thể), mối quan hệ nhị nguyên (mối quan hệ 1-1, 1-n, n-n), mối quan hệ đa nguyên (mối quan hệ giữa ba tập thực thể trở lên).

Trong một hệ thống thông tin, mỗi tập thực thể có thể có nhiều mối quan hệ với các tập thực thể khác nhau. Sơ đồ thực thể - quan hệ biểu diễn mỗi tập thực thể bằng một hình chữ nhật,

mỗi mối quan hệ bằng một hình thoi kèm các cung nối đến các tập thực thể liên quan. Trên mỗi cung này ghi nhận bản số (m, n) , chứa số tối thiểu m và số tối đa n có thể của một phần tử của tập thực thể tham gia vào mối quan hệ. Nếu số tối thiểu hay số tối đa này lớn hơn 1 thì kí hiệu bởi n .

Nếu xem một tập thực thể trình bày một lớp các đối tượng trong thế giới thực thì vấn đề chuyển đổi mô hình thực thể - quan hệ thành mô hình hướng đối tượng là có thể thực hiện khá thích hợp.

Mô hình dữ liệu hướng đối tượng sử dụng một số các khái niệm hướng đối tượng: các lớp (kiểu), các đối tượng phức tạp, đồng nhất đối tượng và sự thừa kế là các khái niệm liên quan đến cấu trúc của mô hình.

Lớp được hiểu là tập các thực thể hay các đối tượng có cùng các đặc tính và hành vi giống nhau. Các đặc tính này được mô tả như các thuộc tính bên trong một lớp đối tượng. Sự liên kết giữa các lớp đối tượng được biểu diễn thông qua các thuộc tính quan hệ.

Mặc dù mô hình cơ sở dữ liệu hướng đối tượng hình thành như một giải pháp nhằm giải quyết sự phức tạp trong việc mô hình hóa thế giới thực, nhưng cho đến nay vẫn chưa có một chuẩn cho mô hình dữ liệu này. Won Kim [4] ủng hộ khái niệm thuộc tính đa trị có trong một lớp và xem như là kiểu dữ liệu trừu tượng. Nhóm nghiên cứu ODMG-93 [1995] đã mở rộng khái niệm kiểu dữ liệu của thuộc tính đa trị: ngoài khái niệm Set (tập hợp), các tác giả còn đưa ra các khái niệm Bag (túi), List (danh sách) và Array (mảng). Nhóm ODMG còn ủng hộ việc khai báo cấu trúc các thuộc tính bên trong một lớp đối tượng là gắn liền với việc khai báo các phương thức của mỗi đối tượng thuộc lớp đó. Khi xây dựng các ràng buộc dữ liệu cho lược đồ hướng đối tượng, Z. Tari và một số tác giả [5] lại ủng hộ khái niệm thuộc tính phức, là thuộc tính được xây dựng từ tập các thuộc tính khác.

Để làm cơ sở cho việc chuyển đổi mô hình thực thể - quan hệ thành mô hình hướng đối tượng, một số qui ước tối thiểu về mô hình hướng đối tượng cần được xét đến như:

- Trong mô hình dữ liệu hướng đối tượng, một lớp đối tượng được xác định thông qua tập các thuộc tính (properties) và các phép toán (operations) nhằm phản ánh cấu trúc dữ liệu của lớp và các hành vi được thực hiện trên mỗi đối tượng thuộc lớp đó. Nhưng do mô hình thực thể - quan hệ là không đề cập đến các thao tác trên các thực thể, vì vậy ta chỉ xét đến tập các thuộc tính trên mỗi lớp đối tượng được chuyển đổi. Tức là kết quả của việc chuyển đổi mô hình sẽ cho ta tập các lớp đối tượng, và mỗi lớp đối tượng bao gồm một tập các thuộc tính.
- Mỗi đối tượng trong một lớp được xác định thông qua tên của đối tượng (object-identifier). Chúng ta sẽ sử dụng thuộc tính định danh để xác định tên của mỗi đối tượng trong lớp.
- Mỗi thuộc tính có thể là thuộc tính đơn trị hoặc thuộc tính đa trị. Thuộc tính đa trị được khai báo bởi từ khóa Set. Trái với thuộc tính đơn trị, một thể hiện của thuộc tính đa trị là tương ứng với một tập hợp các giá trị.
- Một lớp có thể được xem như một kiểu dữ liệu trừu tượng. Một mối quan hệ giữa các lớp đối tượng cũng có thể xem như một thuộc tính của các lớp đối tượng này, gọi là thuộc tính mối quan hệ. Một cách hình thức, ta nói rằng: một lớp A có thuộc tính mối quan hệ R với kiểu dữ liệu là lớp B , có nghĩa là mỗi đối tượng thuộc lớp A có quan hệ nhiều nhất với một đối tượng thuộc lớp B , và giá trị tương ứng của thuộc tính R tại đối tượng này chính là định danh của đối tượng thuộc lớp B . Hiển nhiên rằng nếu mỗi đối tượng thuộc lớp A có quan hệ với nhiều đối tượng thuộc lớp B thì thuộc tính R được xem như thuộc tính đa trị. Khi đó khai báo kiểu dữ liệu của thuộc tính R trong lớp A sẽ là $\text{Set}(B)$, và giá trị tương ứng của chúng chính là các định danh của các đối tượng thuộc lớp B .

Ví dụ 1. Chẳng hạn ta xét một khai báo của lớp NGUOI như sau:

```
Class NGUOI
  properties
    ID_NGUOI: aIID;
    HOTEN: string;
    NGAYSINH: Date;
    DIACHI: Set(string);
    BO, ME: NGUOI;
    CON: Set(NGUOI);
End NGUOI.
```

Ở đây:

- aIID là miền trị liên quan đến tất cả các định danh đối tượng.
- ID_NGUOI, HOTEN, NGAYSINH, BO, ME là các thuộc tính đơn trị. Còn DIACHI, CON là các thuộc tính đa trị, với ngữ nghĩa rằng mỗi đối tượng thuộc lớp NGUOI cho tương ứng với tập các giá trị khác nhau của thuộc tính DIACHI và thuộc tính CON.
- Việc khai báo các thuộc tính BO, ME, CON là các khai báo đệ quy, thể hiện mối quan hệ phản xạ giữa các đối tượng trong lớp NGUOI.

Với những qui ước đã xét ở trên, sẽ cho phép ta xem quá trình chuyển đổi mô hình thực thể - quan hệ thành mô hình hướng đối tượng là quá trình chuyển đổi các tập thực thể và các mối quan hệ bên trong mô hình thực thể - quan hệ thành các lớp của mô hình hướng đối tượng.

2. PHƯƠNG PHÁP CHUYỂN ĐỔI

Phương pháp chuyển đổi mô hình thực thể - quan hệ thành mô hình hướng đối tượng sẽ được bàn luận dựa vào các qui tắc chuyển đổi. Mỗi qui tắc sẽ nêu cách chuyển đổi các tập thực thể và các mối quan hệ thành các lớp. Các qui tắc này bao gồm qui tắc chuyển đổi mối quan hệ is-a giữa hai tập thể, qui tắc chuyển đổi mối quan hệ phản xạ, qui tắc chuyển đổi mối quan hệ nhị nguyên không có thuộc tính, qui tắc chuyển đổi mối quan hệ nhị nguyên có kèm thuộc tính và qui tắc chuyển đổi mối quan hệ đa nguyên. Cuối cùng nếu xem tập các qui tắc này là “đầy đủ tác nghiệp” thì sẽ cho phép chúng ta xây dựng một thuật toán thực hiện việc chuyển đổi một sơ đồ thực thể - mối quan hệ thành các lớp đối tượng của mô hình hướng đối tượng.

2.1. Các qui tắc

Do một tập thực thể trong mô hình thực thể - quan hệ được xem như một lớp các đối tượng trong thế giới thực, ta qui ước rằng bất kỳ một tập thực thể của mô hình thực thể - quan hệ sẽ chuyển đổi thành một lớp đối tượng có tên tương ứng. Vấn đề còn lại của việc chuyển đổi là xây dựng các thuộc tính bên trong mỗi lớp được tạo lập này và bổ sung các lớp cần thiết khác.

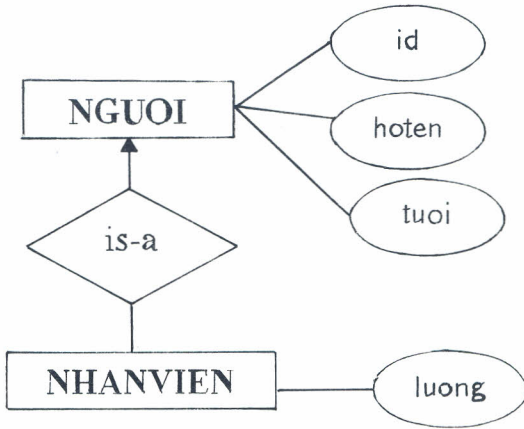
Mỗi qui tắc chuyển đổi chỉ ra một dạng của mối quan hệ cần chuyển đổi. Ta giả thiết rằng một dạng mối quan hệ được xét đến trong sơ đồ thực thể - quan hệ chỉ có thể là một trong các dạng thuộc các qui tắc sau đây.

Qui tắc 1. (Qui tắc chuyển đổi quan hệ is-a)

Nếu tập thực thể A là có mối quan hệ is-a với tập thực thể B thì lớp A sẽ kế thừa tất cả các thuộc tính trong lớp B , đồng thời bổ sung các thuộc tính riêng của lớp A . Đây chính là tính kế thừa trong mô hình dữ liệu hướng đối tượng.

Ví dụ 2.

Mô hình thực thể - quan hệ



Hình 1

Mô hình quan hệ

NGUOI(id, hoten, tuoi)
NHANVIEN(id, luong)

Mô hình hướng đối tượng

```
Class NGUOI
properties
  id: aIID;
  hoten: String;
  tuoi: Integer;
End NGUOI.

Class NHANVIEN
inherits: NGUOI
proties
  luong: Integer;
End NHANVIEN.
```

Qui tắc 2. (Qui tắc chuyển đổi mối quan hệ nhị nguyên không có thuộc tính)

Nếu hai tập thực thể A và B có mối quan hệ R (R không có thuộc tính), thì mỗi lớp A và B , ngoài các thuộc tính trong tập thực thể A và B , sẽ được bổ sung thuộc tính R (gọi là thuộc tính mối quan hệ). Đối với việc khai báo kiểu dữ liệu của thuộc tính này trong mỗi lớp A và B , không mất tính tổng quát, ta xét các trường hợp sau:

* Trường hợp 1: Nếu thông qua R mỗi đối tượng thuộc lớp A có quan hệ nhiều nhất với một đối tượng thuộc lớp B , chẳng hạn cung nối A và R có bản số là $(1,1)$, thì thuộc tính R trong lớp A sẽ được khai báo:

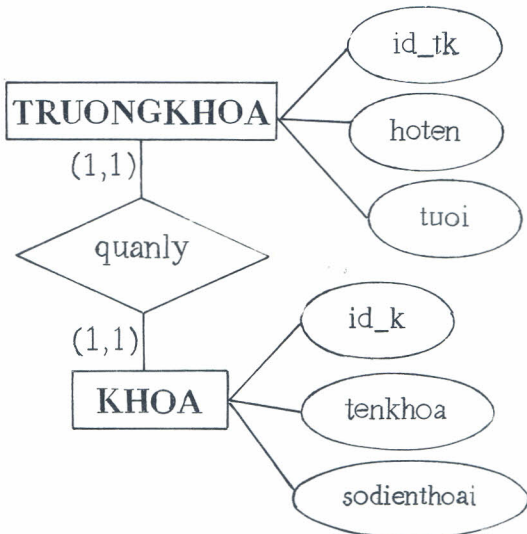
(Tên thuộc tính R): (Lớp B);

* Trường hợp 2: Nếu thông qua R mỗi đối tượng thuộc lớp A có thể quan hệ với nhiều đối tượng thuộc lớp B , chẳng hạn cung nối A và R có bản số là $(1,n)$, thì thuộc tính R trong lớp A sẽ được khai báo

(Tên thuộc tính R): set((Lớp B));

Ví dụ 3. (Mối quan hệ 1-1)

Mô hình thực thể - quan hệ



Hình 2

Mô hình quan hệ

TRUONGKHOA(id_tk, hoten, tuoi)
KHOA(id_k, tenkhoa, sodienthoai, id_tk)

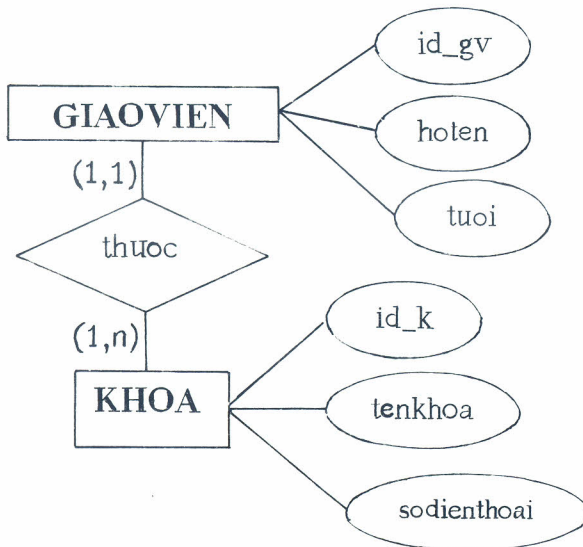
Mô hình hướng đối tượng

```
Class TRUONGKHOA
properties
  id_tk: aIID;
  Hoten: String;
  Tuoi: Integer;
  quanly: KHOA;
End TRUONGKHOA.

Class KHOA
properties
  id_k: aIID;
  Tenkhoa: String;
  Sodienthoai: String;
  quanly: TRUONGKHOA;
End KHOA.
```

Ví dụ 4. (Mối quan hệ 1-n)

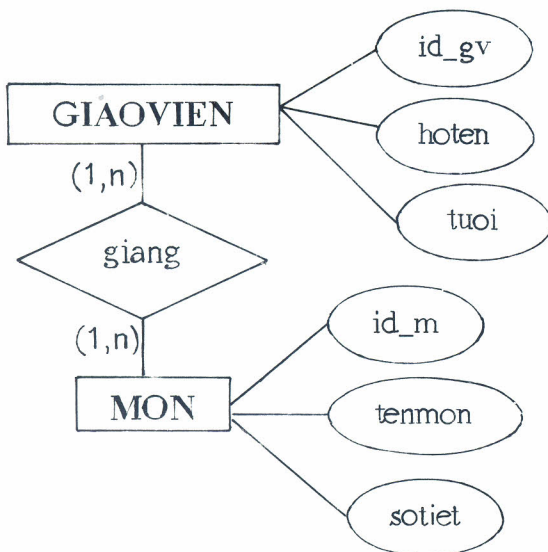
Mô hình thực thể - quan hệ



Hình 3

Ví dụ 5. (Mối quan hệ n-n)

Mô hình thực thể - quan hệ



Hình 4

Mô hình quan hệ

GIAOVIEN(id_gv, hoten, tuoi, id_k)
KHOA(id_k, tenkhoa, sodienthoai)

Mô hình hướng đối tượng

Class **GIAOVIEN**

properties

Id_gv: aIID;

Hoten: String;

Tuoi: Integer;

Thuoc: KHOA;

End GIAOVIEN.

Class **KHOA**

properties

Id_k: aIID;

Tenkhoa: String;

Sodienthoai: String;

Thuoc: set(GIAOVIEN);

End KHOA.

Mô hình quan hệ

GIAOVIEN(id_gv, hoten, tuoi)
MON(id_m, tenmon, sotiet)
GIANG(id_gv, id_m)

Mô hình hướng đối tượng

Class **GIAOVIEN**

propeties

Id_gv: aIID;

Hoten: String;

Tuoi: Integer;

Giang: set(MON);

End GIAOVIEN.

Class **MON**

propeties

Id_m: aIID;

Tenmon: String;

Sotiet: Integer;

Giang: set(GIAOVIEN);

End MON.

Qui tắc 3. (Qui tắc chuyển đổi quan hệ phản xạ)

Xét một tập thực thể A có mối quan hệ R vào chính tập A . Nếu thông qua R mỗi đối tượng thuộc lớp A có quan hệ nhiều nhất với một đối tượng thuộc lớp A thì thuộc tính R trong lớp A sẽ được khai báo: **(Tên thuộc tính R): (Lớp A)**. Trong trường hợp mỗi đối tượng thuộc lớp A có thể quan hệ với nhiều đối tượng thuộc lớp A , thì thuộc tính R có khai báo **(Tên thuộc tính R): set((Lớp A))**.

Ví dụ 1 đã xét trước đây cho thấy việc khai báo các thuộc tính BO, ME, CON là các khai báo đệ qui, thể hiện mối quan hệ phản xạ giữa các đối tượng trong lớp NGUOI.

Qui tắc 4. (Qui tắc chuyển đổi quan hệ nhị nguyên có kèm thuộc tính)

Nếu hai tập thực thể A_1 và A_2 có mối quan hệ R và mỗi quan hệ có kèm các thuộc tính, thì ngoài hai lớp A_1 và A_2 ta cần bổ sung lớp mới C đóng vai trò trung gian (lớp C có mối quan hệ 1-1 hay n-1 với các lớp A_1 và A_2 là tùy theo chỉ số cực đại trong bản số của cung nối các tập thực thể A_1 và A_2 với mối quan hệ R). Cụ thể:

- Lớp A_1 được bổ sung thuộc tính R_1 có khai báo: **(Tên thuộc tính R_1): (Lớp C)** nếu cung nối tập thực thể A_1 và mối quan hệ R có chỉ số cực đại của bản số là 1; hoặc khai báo:

(Tên thuộc tính R_1): set((Lớp C)) nếu cung nối tập thực thể A_1 và mối quan hệ R có chỉ số cực đại của bản số là n .

- Lớp A_2 được bổ sung thuộc tính R_2 có khai báo tương tự như lớp R_1 .

- Lớp C bao gồm các thuộc tính sau:

- + Thuộc tính định danh của lớp C ,
- + Các thuộc tính của mối quan hệ R ,
- + Và hai thuộc tính R_1, R_2 có khai báo:

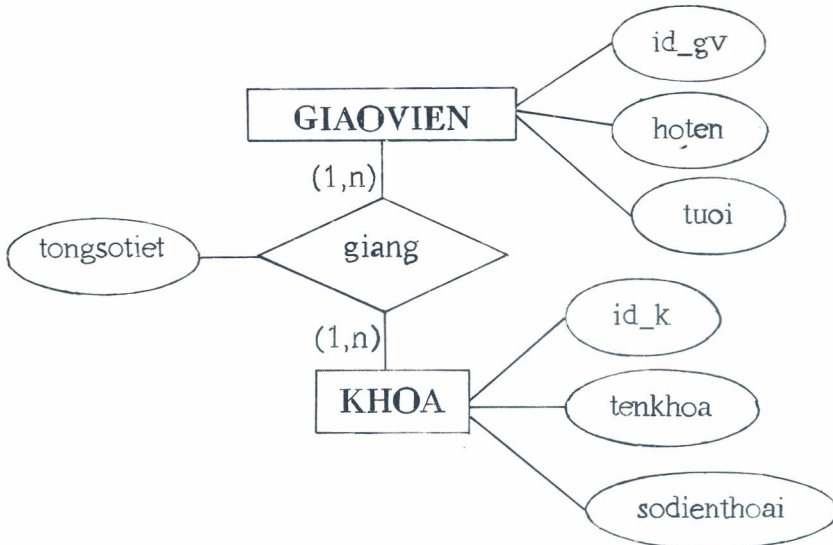
(Tên thuộc tính R_1): (Lớp A_1);

(Tên thuộc tính R_2): (Lớp A_2).

Hiển nhiên rằng qui tắc này cũng có thể áp dụng để chuyển đổi các mối quan hệ nhị nguyên không có thuộc tính. Lúc này lớp C chỉ bao gồm ba thuộc tính: thuộc tính định danh của lớp C và hai thuộc tính R_1, R_2 có khai báo như trên. Tuy nhiên trong trường hợp này thì việc ứng dụng Qui tắc 3 là được khuyến khích, vì ta không cần bổ sung lớp C đóng vai trò trung gian.

Ví dụ 6. (Mối quan hệ n-n mở rộng)

Mô hình thực thể - quan hệ



Hình 5

Mô hình quan hệ**GIAOVIEN**(id_gv, hoten, tuoi)**KKOA**(id_k, tenkhoa, sodienthoai)**GVIEN_KHOA**(id_gv, id_k, tongsotiet)**Mô hình hướng đối tượng**Class **GIAOVIEN**

```

properties
  Id_gv: aIID;
  Hoten: String;
  Tuoi: Integer;
  Giang 1: set(GVIEN_KHOA)
End GIAOVIEN.

```

Class **KHOA**

```

properties
  Id_k: aIID;
  Tenkhoa: String;
  Sodienthoai: String;
  Giang2: set(GVIEN_KHOA);
End KHOA.

```

Class **GVIEN_KHOA**

```

properties
  Id_gvien_khoa: aIID;
  Giang 1: GIAOVIEN
  Giang 2: KHOA
  Tongsotiet: Integer;
End GVIEN_KHOA.

```

Từ Qui tắc 4 ta có thể mở rộng để thực hiện việc chuyển đổi mối quan hệ đa nguyên theo Qui tắc 5 như sau.

Qui tắc 5. (Qui tắc chuyển đổi quan hệ đa nguyên)

Nếu k tập thực thể A_1, A_2, \dots, A_k ($k \geq 2$) có quan hệ với nhau thông qua quan hệ đa nguyên R (có k cung nối mỗi tập thực thể A_1, A_2, \dots, A_k với mỗi quan hệ R), thì ngoài k lớp A_1, A_2, \dots, A_k ta sẽ bổ sung lớp mới C đóng vai trò trung gian. Cụ thể:

- Mỗi lớp A_i được bổ sung thuộc tính R_i có khai báo: **(Tên thuộc tính R_i): (Lớp C)** nếu cung nối tập thực thể A_i và mỗi quan hệ R có chỉ số cực đại của bản số là 1; hoặc khai báo:

(Tên thuộc tính R_i): set((Lớp C)) nếu cung nối tập thực thể A_i và mỗi quan hệ R có chỉ số cực đại của bản số là n (với $i = 1, 2, \dots, k$).

- Lớp C bao gồm các thuộc tính sau:

- + thuộc tính định danh của lớp C ,
- + các thuộc tính của mỗi quan hệ R ,
- + và các thuộc tính R_i có khai báo:

(Tên thuộc tính R_i): (Lớp A_i) (với $i = 1, 2, \dots, k$).

Chúng ta có thể thực hiện việc chuyển đổi một sơ đồ thực thể - quan hệ thành các lớp đối tượng của mô hình hướng đối tượng nếu xem tập các qui tắc trên là “đầy đủ tác nghiệp” (mà thực chất là bốn qui tắc: 1, 2, 3 và 5). Bằng cách xuất phát từ một mối quan hệ trong sơ đồ ta thực hiện việc chuyển đổi, từ đó thực hiện tiếp việc chuyển đổi cho đối với các mối quan hệ “lân cận”, cho đến khi tất cả các mối quan hệ trong sơ đồ đều được chuyển đổi xong. Một cách hình thức ta xây dựng thuật toán chuyển đổi như sau.

2.2. Thuật toán chuyển đổi

Dựa vào các qui tắc đã được nêu ở trên, chúng ta có thể xây dựng một thuật toán nhằm thực hiện việc chuyển đổi mô hình thực thể - quan hệ thành mô hình hướng đối tượng. Ta sẽ lần lượt sử

dụng các qui tắc trong 2.1 để thực hiện việc chuyển đổi này. Mỗi lượt sử dụng một trong các qui tắc này để thực hiện việc chuyển đổi được gọi là một phép chuyển đổi.

Input: Sơ đồ thực thể - quan hệ.

Output: Tập các lớp của mô hình hướng đối tượng.

Method:

Trước hết ta xem sơ đồ thực thể - quan hệ này như một đồ thị, với các nút là các tập thực thể và các mối quan hệ (các nút chữ nhật và nút hình thoi), và các cạnh là cung nối các nút này trong sơ đồ. Với qui định rằng một mối quan hệ được xét đến trong sơ đồ này chỉ có thể là một trong các mối quan hệ được chỉ ra trong các qui tắc ở 2.1, thì rõ ràng nút kề của nút chữ nhật là nút hình thoi, và ngược lại. Ngoài ra ta có thể giả thiết rằng đồ thị này là liên thông.

Trong quá trình thực hiện thuật toán, một cạnh trên đồ thị được gọi là “cạnh đã duyệt” nếu một phép chuyển đổi đã được thực hiện đối với mối quan hệ có liên quan đến một cạnh đó, ngược lại được gọi là “cạnh chưa được duyệt”.

Một nút được gọi là “nút đã duyệt” nếu một phép chuyển đổi đã được thực hiện đối với mối quan hệ có liên quan đến nút đó, ngược lại ta gọi là “nút chưa được duyệt”. Một nút được gọi là “nút đã xét xong” nếu tất cả các cạnh nối đến nút đó là cạnh đã duyệt.

Một nút đã xét xong hiển nhiên cũng là nút đã duyệt. Ta nhận thấy rằng sau một phép chuyển đổi cho một mối quan hệ, thì nút hình thoi (mối quan hệ) sẽ trở thành nút đã xét xong. Nhưng các nút chữ nhật (các thực thể) liên quan chưa chắc là các nút đã xét xong mặc dù chúng là các nút đã duyệt.

Sau một phép chuyển đổi ta sẽ thu được một lớp tương ứng với phép chuyển đổi này. Lớp thu được từ một tập thực thể được gọi là “lớp hoàn thành” nếu nút tương ứng với tập thực thể này là nút đã xét xong, còn ngược lại được gọi là “lớp chưa hoàn thành”.

Với các qui ước trên thì thuật toán chuyển đổi sẽ được thực hiện theo các bước sau:

Bước 1: Xuất phát từ một nút hình thoi (quan hệ) thực hiện phép chuyển đổi dựa vào một trong các qui tắc của 2.1. Khi đó nút hình thoi này trở thành nút đã xét xong, còn các cạnh và các nút hình chữ nhật (tập thực thể) liên quan là các cạnh và các nút đã duyệt.

Bước 2:

While “Tồn tại nút A là nút đã duyệt, nhưng không là nút đã xét xong” **do**

Begin

- Định vị đến nút đã xét A ;
{nút này hiển nhiên là nút chữ nhật (tập thực thể), và lớp A tương ứng với nút này là lớp chưa hoàn thành}
- Tìm nút B là nút kề của nút A và là nút chưa được duyệt;
{nút B hiển nhiên là tồn tại và là nút hình thoi (mối quan hệ)}
- Thực hiện phép chuyển đổi đối với nút B ;
{lưu ý rằng đối với các lớp chưa hoàn thành (như lớp A) thì ta chỉ bổ sung các thuộc tính còn thiếu so với qui tắc đề ra}

End;

Khi bước 2 hoàn thành có nghĩa là mọi nút đã duyệt thuộc đồ thị này đồng thời cũng là nút đã xét xong. Thuật toán dừng và các lớp thu được là kết quả của việc chuyển đổi mô hình.

3. KẾT LUẬN

Chuyển đổi mô hình là một trong những vấn đề đáng quan tâm. Việc đề xuất phương pháp “Chuyển đổi mô hình thực thể - quan hệ thành mô hình hướng đối tượng” cũng góp phần giải quyết vấn đề chuyển đổi mô hình dữ liệu quan hệ thành mô hình đối tượng. Bởi lẽ trong thực tế đa phần các mô hình quan hệ là được thiết kế xuất phát từ mô hình thực thể - quan hệ. Việc chuyển đổi

mô hình dữ liệu quan hệ thành mô hình hướng đối tượng sẽ cho phép các cơ sở dữ liệu hướng đối tượng kế thừa được phần dữ liệu trên các hệ quản trị cơ sở dữ liệu quan hệ, đồng thời lại thúc đẩy các hệ multi-database phát triển. Điều đó cho phép mở ra xu hướng tích hợp các hệ quản trị cơ sở dữ liệu theo những mô hình dữ liệu khác nhau.

TÀI LIỆU THAM KHẢO

- [1] Batini C., S. Ceri, S. B. Navathe, *Conceptual Database Design*, Benjamin/Cummings, 1992.
- [2] Cattell R., *The Object Database Standard: ODMG-93 Release 1.2*, Morgan-Kaufmann, 1992.
- [3] Fong J., B. Siu, *Multimedia, Knowledge-Base and Object-Oriented Databases*, Springer, 1996.
- [4] Kim W., Object-oriented databases: definition and research directions, *IEEE Transactions on Knowledge and Data Engineering* **2** (3) (1990).
- [5] Tari Z., J. Stokes, S. Spaccapietra, Object normal forms and dependency constraints for object-oriented schemata, *ACM Transactions on Database Systems* **22** (4) (1997).
- [6] Ullman J. D., J. Widom, *A First Course in Database System*, Prentice Hall, 1997.

Nhận bài ngày 4 - 5 - 2001

Nhận lại sau khi sửa ngày 20 - 7 - 2001

Trường Đại học Khoa học, Đại học Huế.