

Chương III- MÔ HÌNH QUAN HỆ, CÁC RÀNG BUỘC QUAN HỆ VÀ ĐẠI SỐ QUAN HỆ

Mô hình quan hệ được Ted Codd đưa ra đầu tiên vào năm 1970 và gây được chú ý ngay tức khắc vì tính đơn giản và các cơ sở toán học của nó. Mô hình quan hệ sử dụng khái niệm quan hệ toán học như là khối xây dựng cơ sở và có cơ sở lý thuyết của nó trong lý thuyết tập hợp và logic vị từ bậc nhất. Trong chương này chúng ta sẽ nói về các đặc trưng cơ bản của mô hình, các ràng buộc của chúng và tập hợp các phép toán của mô hình quan hệ.

I- Các khái niệm của mô hình quan hệ

Mô hình quan hệ biểu thị cơ sở dữ liệu như một tập các quan hệ. Mỗi quan hệ có thể được biểu diễn như một bảng giá trị, mỗi một dòng trong bảng biểu thị một tập hợp các giá trị dữ liệu liên quan với nhau. Trong chương trước, chúng ta đã đưa ra các khái niệm về kiểu thực thể và kiểu liên kết như là các khái niệm để mô hình hoá dữ liệu của thế giới thực. Trong mô hình quan hệ, mỗi một dòng trong bảng biểu thị một sự kiện tương ứng với một thực thể hoặc một liên kết của thế giới thực. Tên bảng và tên các cột dùng để giúp giải thích ý nghĩa của các giá trị trong mỗi hàng. Mọi giá trị trong một cột đều cùng một kiểu dữ liệu

Theo thuật ngữ mô hình quan hệ hình thức, mỗi hàng được gọi là một bộ, mỗi đầu cột được gọi là một thuộc tính, và bảng được gọi là một quan hệ. Kiểu dữ liệu mô tả các kiểu của dữ liệu xuất hiện trong mỗi cột gọi là một miền

I.1- Miền, thuộc tính, bộ và quan hệ

Một **miền D** là một tập hợp các giá trị nguyên tử, điều đó có nghĩa là mỗi giá trị trong miền là không thể phân chia được trong phạm vi mô hình quan hệ. Để đặc tả một miền, người ta chỉ ra một tên, một kiểu dữ liệu và khuôn dạng dữ liệu. Một số ví dụ về định nghĩa miền:

- . Họ tên: Tập hợp các dãy chữ cái có độ dài ≤ 30 .
- . Tuổi: Tập các số nguyên nằm trong khoảng từ 1 đến 80.
- . Giới tính: Tập hợp gồm hai giá trị “Nam”, “Nữ”.

Ngoài ra, trong cơ sở dữ liệu người ta còn chỉ ra các thông tin phụ để thể hiện các giá trị của miền, chẳng hạn các đơn vị tính như tiền, trọng lượng,...

Một **lược đồ quan hệ** R , ký hiệu là $R(A_1, A_2, \dots, A_n)$, được tạo nên từ một tên quan hệ R một danh sách các thuộc tính $A_1, A_2, \dots, A_n$. Mỗi một thuộc tính A_i là tên vai trò của một miền D nào đó trong lược đồ quan hệ R . D được gọi là miền giá trị của A_i và được ký hiệu là $Dom(A_i)$. Một lược đồ quan hệ được sử dụng để mô tả một quan hệ, R được gọi là tên của quan hệ đó. Cấp của một quan hệ là số các thuộc tính của lược đồ quan hệ của nó. Ví dụ, ta có lược đồ cho quan hệ cấp 5: SINHVIÊN (Mã số, Họ tên, Ngày sinh, Giới tính, Địa chỉ). Với lược đồ quan hệ này, SINHVIÊN là tên của quan hệ.

Một **quan hệ** (hoặc trạng thái quan hệ) r của lược đồ quan hệ $R(A_1, A_2, \dots, A_n)$ được ký hiệu là $r(R)$, là tập hợp các n -bộ $r = \{t_1, t_2, \dots, t_n\}$. Mỗi n -bộ t là một danh sách có thứ tự của n giá trị, $t = \langle v_1, v_2, \dots, v_n \rangle$, trong đó mỗi $v_i, 1 \leq i \leq n$, là một phần tử của $Dom(A_i)$ hoặc là một giá trị không xác định (null value). Giá trị thứ i của bộ t , tương ứng với thuộc tính A_i được ký hiệu là $t[A_i]$. Hình III-1 chỉ ra một ví dụ của quan hệ SINHVIÊN tương ứng với lược đồ quan hệ SINHVIÊN ở trên. Mỗi bộ trong quan hệ biểu diễn một thực thể sinh viên cụ thể. Quan hệ được biểu diễn như một bảng, trong đó mỗi bộ được hiển thị như một hàng và mỗi thuộc tính tương ứng với một đầu cột chỉ ra vai trò của các giá trị trong cột đó. Các giá trị không xác định biểu thị các thuộc tính mà giá trị của nó không biết được hoặc không tồn tại đối với từng bộ SINHVIÊN cụ thể.

SINHVIÊN	Họ tên	Mã số	Ngày sinh	Giới tính	Địa chỉ
	Lê Văn	4515202	12/09/84	Nữ	Hà nội
	Hoàng Tùng	4516802	21/03/84	Nam	Bắc ninh
	Trương Định	4620503	15/05/85	Nam	Hà nam
	Phạm An	4612203	16/04/85	Nam	Nam định
	Đỗ Cung	4521402	20/01/84	Nam	Nghệ an

Hình III-1. Quan hệ SINHVIÊN

Định nghĩa quan hệ ở trên có thể phát biểu lại như sau: Một quan hệ $r(R)$ là một quan hệ toán học cấp n trên các miền giá trị $\text{dom}(A_1), \text{dom}(A_2), \dots, \text{dom}(A_n)$, đó là tập con của tích Đề các của các miền giá trị xác định R :

$$r(R) \subseteq (\text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n))$$

Tích Đề các chỉ ra mọi tổ hợp có thể có của các giá trị từ các miền đã cho. Như vậy, nếu ta ký hiệu lực lượng của một miền D là $|D|$ và giả thiết rằng mọi miền đều hữu hạn thì tổng số các bộ trong tích Đề các là:

$$|\text{dom}(A_1)| * |\text{dom}(A_2)| * \dots * |\text{dom}(A_n)|$$

Ngoài tất cả các tổ hợp có thể có này, một trạng thái quan hệ ở một thời điểm cho trước - gọi là *trạng thái quan hệ hiện tại* - chỉ phản ánh các bộ giá trị biểu diễn một trạng thái cụ thể của thế giới thực. Nói chung, do trạng thái của thế giới thực thay đổi, quan hệ cũng bị thay đổi thành trạng thái quan hệ khác. Tuy nhiên, lược đồ R là ổn định, không thay đổi, trừ phi phải thêm vào một số thuộc tính để biểu diễn một thông tin mới chưa được lưu trữ trong quan hệ.

Có thể xảy ra trường hợp nhiều thuộc tính có cùng một miền giá trị. Các thuộc tính chỉ ra các vai trò khác nhau đối với miền. Ví dụ, hai thuộc tính *Địa chỉ NV* và *Địa chỉ DV* có cùng miền giá trị nhưng thuộc tính thứ nhất tham chiếu đến địa chỉ của nhân viên còn địa chỉ thứ hai tham chiếu đến địa chỉ của đơn vị.

1.2- Các đặc trưng của các quan hệ

1.2.1- Thứ tự của các bộ trong một quan hệ

Một quan hệ được định nghĩa như một tập hợp các bộ. Các phần tử trong một tập hợp không có thứ tự, vì vậy các bộ trong một quan hệ không có một thứ tự cụ thể. Tuy nhiên, trong một tệp, các bản ghi được lưu trữ một cách vật lý trên đĩa vì vậy luôn có một thứ tự giữa các bản ghi. Thứ tự này chỉ rõ bản ghi thứ nhất, bản ghi thứ hai, ..., bản ghi thứ n . Một cách tương tự, khi ta biểu diễn một quan hệ như là một bảng, các hàng được hiển thị theo một thứ tự nhất định.

Thứ tự các bộ không phải là một phần của định nghĩa quan hệ bởi vì một quan hệ cố gắng biểu diễn các sự vật ở mức trừu tượng hoặc logic. Có thể có nhiều thứ tự logic trên một quan hệ. Ví dụ, các bộ giá trị trong quan hệ *SINHVIÊN* ở hình III-1 có thể sắp xếp theo nhiều cách khác nhau: theo thứ tự logic của *Họ tên*, theo thứ tự logic của *Mã số*,... Định nghĩa quan hệ không chỉ ra thứ tự logic nào cả, vì

vậy không có thứ tự logic nào hơn thứ tự logic khác. Các quan hệ chứa cùng một số hàng như nhau nhưng các hàng được sắp xếp khác nhau được xem như đồng nhất với nhau. Khi một quan hệ được cài đặt như một tệp, một thứ tự vật lý có thể được chỉ ra trên các bản ghi của tệp.

I.2.2- Thứ tự của các giá trị bên trong một bộ

Theo định nghĩa quan hệ ở trên, một n-bộ là một danh sách có thứ tự của n giá trị. Như vậy thứ tự của các giá trị trong một bộ là quan trọng, từ đó suy ra thứ tự của các thuộc tính trong một lược đồ quan hệ cũng quan trọng. Tuy nhiên, ở mức logic, thứ tự của các thuộc tính và các giá trị của nó là không thực sự quan trọng khi giữ được sự tương ứng giữa các thuộc tính và các giá trị.

Có thể đưa ra một định nghĩa khác về quan hệ, định nghĩa này sẽ làm cho thứ tự của các giá trị trong một bộ là không cần thiết. Theo định nghĩa này, một lược đồ quan hệ $R = \{A_1, A_2, \dots, A_n\}$ là một tập hợp các thuộc tính và một quan hệ $r(R)$ là một tập hợp hữu hạn các ánh xạ $r = \{t_1, t_2, \dots, t_m\}$, trong đó mỗi t_i là một ánh xạ từ R vào D , trong đó $D = \text{dom}(A_1) \cup \text{dom}(A_2) \cup \dots \cup \text{dom}(A_n)$. Trong định nghĩa này, $t[A_i]$ phải ở trong $\text{dom}(A_i)$ với $1 \leq i \leq n$ với mỗi ánh xạ t_i trong r . Mỗi ánh xạ t_i được gọi là một bộ.

Theo định nghĩa này, một bộ có thể xem như một tập hợp các cặp ($\langle$ thuộc tính $\rangle$, $\langle$ giá trị $\rangle$), trong đó mỗi cặp cho một giá trị của ánh xạ từ một thuộc tính A_i đến một giá trị v_i của $\text{dom}(A_i)$. Vì tên thuộc tính xuất hiện cùng với giá trị của nó nên thứ tự của các thuộc tính là không quan trọng. Điều này làm nên ý nghĩa ở mức trừu tượng hoặc logic vì chẳng có lý do gì để thích có một giá trị thuộc tính xuất hiện trước một giá trị thuộc tính khác trong một bộ.

Khi một quan hệ được cài đặt như một tệp, các thuộc tính được sắp xếp một cách vật lý như là các trường trong một bản ghi. Trong trường hợp đó chúng ta sẽ sử dụng định nghĩa thứ nhất của quan hệ, trong đó các giá trị của các thuộc tính trong một bộ là có thứ tự vì nó làm đơn giản rất nhiều khái niệm. Tuy nhiên, định nghĩa thứ hai là tổng quát hơn.

I.2.3- Các giá trị trong một bộ

Mỗi giá trị trong một bộ là một giá trị nguyên tử, điều đó có nghĩa là nó không phân chia được thành các thành phần trong phạm vi của mô hình quan hệ. Như vậy, trong mô hình quan hệ không cho phép có các thuộc tính phức hợp và các thuộc

tính đa trị. Các thuộc tính đa trị phải được biểu diễn bằng các quan hệ còn các thuộc tính phức hợp chỉ được biểu diễn bằng các thuộc tính thành phần đơn của nó.

Các giá trị của một vài thuộc tính trong một bộ cụ thể có thể không biết được hoặc không thích ứng cho nó. Trường hợp đó, người ta sử dụng một giá trị đặc biệt gọi là giá trị null. Ví dụ, giả sử quan hệ SINHVIÊN có thuộc tính SỐđiệnthoạịnà. Trong một tập thể sinh viên, có người có điện thoại ở nhà, có người không có và cũng có người có nhưng không biết chắc. Với những trường hợp không có hoặc không biết chắc, thuộc tính SỐđiệnthoạịnà có giá trị null.

I.2.4- Thể hiện của một quan hệ

Một lược đồ quan hệ có thể được thể hiện như là một tuyên bố hoặc một khẳng định. Ví dụ lược đồ quan hệ SINHVIÊN ở trên khẳng định rằng, nói chung, một thực thể sinh viên có một mã số, họ tên, ngày sinh, giới tính, địa chỉ. Mỗi bộ trong quan hệ được thể hiện như là một sự kiện hoặc như một thể hiện cụ thể của một khẳng định. Ngoài các quan hệ biểu diễn các sự kiện về các thực thể, một số quan hệ có thể biểu diễn các sự kiện về mối liên kết. Ví dụ, lược đồ quan hệ NHÂNVIÊN_DỰÁN(Mã sốNV, Mã sốDA, Số giờ) khẳng định các nhân viên làm việc với các dự án. Mỗi bộ trong quan hệ này liên kết một nhân viên với một dự án mà anh ta làm việc cho nó.

Như vậy, mô hình quan hệ biểu diễn các sự kiện về thực thể và các sự kiện về liên kết dưới dạng duy nhất là các quan hệ.

II- Các ràng buộc quan hệ, lược đồ cơ sở dữ liệu quan hệ

Trong phần này chúng ta thảo luận về các hạn chế trên các dữ liệu trong một lược đồ cơ sở dữ liệu quan hệ. Các hạn chế đó được gọi là các ràng buộc. Có các loại ràng buộc: ràng buộc miền, ràng buộc khoá, ràng buộc toàn vẹn thực thể và ràng buộc toàn vẹn tham chiếu.

II.1- Các ràng buộc miền

Các ràng buộc miền chỉ ra rằng giá trị của mỗi thuộc tính A phải là một giá trị nguyên tử thuộc miền giá trị dom(A). Các kiểu dữ liệu liên kết với các miền bao gồm: các kiểu dữ liệu số chuẩn cho các số nguyên (short integer, integer, long integer), các số thực (float, double precision float). Ngoài ra còn các kiểu dữ liệu ký tự (dãy ký tự với độ dài cố định, dãy ký tự với độ dài thay đổi), ngày, thời gian và

tiền tệ. Các loại miền khác có thể là các miền con của một kiểu dữ liệu hoặc một kiểu dữ liệu đếm được trong đó mọi giá trị có thể được liệt kê rõ ràng

II.2- Ràng buộc khoá và ràng buộc trên các giá trị không xác định (null)

Một quan hệ được định nghĩa như một tập hợp các bộ. Theo định nghĩa, các phần tử của một tập hợp là khác nhau, vì vậy, mọi bộ trong quan hệ phải khác nhau. Điều đó có nghĩa là không có hai bộ có cùng một tổ hợp giá trị cho tất cả các thuộc tính của chúng. Thông thường, có tồn tại các tập con của các thuộc tính của một lược đồ quan hệ có tính chất là không có hai bộ nào ở trong mọi trạng thái quan hệ r của R có cùng một tổ hợp giá trị cho các thuộc tính của nó. Giả sử chúng ta ký hiệu một tập con như vậy là SK , khi đó với hai bộ khác nhau bất kỳ t_1 và t_2 trong một trạng thái quan hệ r của R chúng ta có ràng buộc là $t_1[SK] \neq t_2[SK]$.

Tập hợp thuộc tính SK như vậy được gọi là một *siêu khoá* của lược đồ quan hệ R . Một siêu khoá SK xác định rõ một ràng buộc về tính duy nhất, phát biểu rằng không có hai bộ khác nhau trong một trạng thái r của R có cùng một giá trị cho SK . Mỗi quan hệ có ít nhất là một siêu khoá mặc định, đó là tập hợp tất cả các thuộc tính của nó. Một *khoá* K của một lược đồ quan hệ R là một siêu khoá của R với tính chất là nếu bỏ đi bất kỳ thuộc tính A nào ra khỏi K thì sẽ còn lại một tập K không phải là siêu khoá của R . Như vậy, một khoá là một siêu khoá tối thiểu, nghĩa là đó là một siêu khoá mà ta không thể rút bỏ thuộc tính nào ra khỏi nó mà vẫn giữ được ràng buộc về tính duy nhất.

Ví dụ, xét quan hệ $SINHVIÊN$ với các thuộc tính $MãSỐ$, $HọTên$, $Ngàysinh$, $Giới tính$, $Địa chỉ$. Thuộc tính $\{MãSỐ\}$ là một khoá của $SINHVIÊN$ bởi vì không có hai bộ sinh viên có cùng một giá trị cho $MãSỐ$. Mọi tập hợp thuộc tính có chứa $MãSỐ$, ví dụ $\{MãSỐ, HọTên, Ngàysinh\}$, đều là một siêu khoá. Tuy nhiên, siêu khoá $\{MãSỐ, HọTên, Ngàysinh\}$ không phải là khoá bởi vì nếu bỏ đi thuộc tính $HọTên$ hoặc $Ngàysinh$ hoặc cả hai thì nó vẫn còn là một siêu khoá.

Giá trị của một thuộc tính khoá có thể được sử dụng để xác định một cách duy nhất mỗi bộ trong một quan hệ. Ví dụ, giá trị 4515202 của $MãSỐ$ xác định một cách duy nhất bộ giá trị tương ứng với sinh viên Lê Văn trong quan hệ $SINHVIÊN$. Chú ý rằng một tập hợp thuộc tính tạo nên một khoá là một tính chất của lược đồ quan hệ. Điều ràng buộc là tính chất đó phải thỏa mãn trên mọi trạng thái của lược đồ. Một khoá được xác định từ ý nghĩa của các thuộc tính và tính chất là bất biến, tính chất đó phải thỏa mãn khi chúng ta chèn thêm các bộ mới vào quan hệ. Ví dụ, ta

không thể và không được chỉ định thuộc tính Họ tên của quan hệ SINHVIÊN là khoá bởi vì không có gì đảm bảo rằng không tồn tại hai sinh viên có cùng họ tên.

Nói chung, một lược đồ quan hệ có thể có nhiều hơn một khoá. Trong trường hợp đó, mỗi một khoá được gọi là một *khoá dự tuyển*. Thông thường ta phải chỉ định một trong các khoá dự tuyển làm *khoá chính* của quan hệ. Khoá chính là một khoá dự tuyển mà các giá trị của chúng được dùng để xác định các bộ trong quan hệ. Ta quy ước rằng, các thuộc tính tạo nên khoá chính của một lược đồ quan hệ được gạch dưới. Ví dụ:

SINHVIÊN(Mã số, Họ tên, Ngày sinh, Giới tính, Địa chỉ).

Chú ý rằng khi một lược đồ quan hệ có nhiều khoá dự tuyển, việc lựa chọn một khoá dự tuyển để làm khoá chính là tùy ý, tuy nhiên tốt nhất là chọn khoá chính gồm một thuộc tính hoặc có số các thuộc tính ít nhất.

Một ràng buộc khác trên các thuộc tính chỉ rõ khi nào thì cho phép các giá trị null. Những thuộc tính luôn luôn phải có một giá trị xác định và hợp lệ thì bị ràng buộc là NOT NULL.

II.3- Cơ sở dữ liệu quan hệ và lược đồ cơ sở dữ liệu quan hệ

Ở trên, chúng ta đã nói đến các lược đồ quan hệ đơn lẻ và các quan hệ đơn lẻ. Một cơ sở dữ liệu quan hệ thường gồm nhiều quan hệ với các bộ giá trị trong các quan hệ được liên kết với nhau theo nhiều cách. Trong phần này chúng ta sẽ định nghĩa một cơ sở dữ liệu quan hệ và một lược đồ cơ sở dữ liệu quan hệ. Một lược đồ cơ sở dữ liệu quan hệ S là một tập hợp các lược đồ quan hệ

$S = \{R_1, R_2, \dots, R_n\}$ và một tập các ràng buộc toàn vẹn.

Một trạng thái cơ sở dữ liệu quan hệ (hoặc một cơ sở dữ liệu quan hệ) DB của S là một tập hợp các trạng thái quan hệ:

$DB = \{r_1, r_2, \dots, r_n\}$

sao cho mỗi r_i là một trạng thái của R_i và sao cho các trạng thái quan hệ r_i thoả mãn các ràng buộc toàn vẹn chỉ ra trong tập các ràng buộc toàn vẹn.

Ví dụ, Hình III-2 trình bày một lược đồ cơ sở dữ liệu CÔNGTY và hình III-3 trình bày một cơ sở dữ liệu công ty.

NHÂNVIÊN(Họ tên, Tên, Mã số NV, Ngày sinh, Địa chỉ, Giới tính, Lương,
Mã số NGS, Mã số ĐV)

ĐƠN VỊ(TênĐV, Mã sốĐV, Mã sốNQL, Ngàybắtđầu)

ĐƠN VỊ_ ĐỊA ĐIỂM(Mã sốĐV, Địa điểmĐV)

DỰ ÁN(TênDA, Mã sốDA, Địa điểmDA, Mã sốĐV)

NHÂN VIÊN_DỰ ÁN(Mã sốNV, Mã sốDA, Số giờ)

CON(Mã sốNV, TênPT, Giới tính, Ngày sinh)

Hình III-2. Lược đồ cơ sở dữ liệu “CÔNG TY”

NHÂN VIÊN	Mã sốNV	Họ đệm	Tên	Ngày sinh	Địa chỉ	Giới tính	Lương	Mã sốNGS	Mã sốĐV
	NV001	Lê	Vân	12/02/79	Hà nội	Nam	3000	NV002	5
	NV002	Trần Đức	Nam	14/02/66	Hà nội	Nam	4000	NV061	5
	NV010	Hoàng	Thanh	05/08/79	Nghệ an	Nữ	2500	NV014	4
	NV014	Phạm	Bằng	26/06/52	Bắc ninh	Nam	4300	NV061	4
	NV016	Nguyễn	Son	14/08/73	Hà nam	Nam	3800	NV002	5
	NV018	Vũ Hương	Giang	26/03/83	Nam định	Nữ	2500	NV002	5
	NV025	Trần Lê	Hoa	15/03/80	Phú thọ	Nữ	2500	NV014	4
	NV061	Hoàng	Giáp	02/05/47	Hà tĩnh	Nam	5500	Null	1

ĐƠN VỊ	Mã sốĐV	TênĐV	Mã sốNQL	Ngày bắt đầu
	5	Nghiên cứu	NV002	15/09/2000
	4	Hành chính	NV014	24/06/1997
	1	Lãnh đạo	NV061	25/01/1992

ĐƠN VỊ_ ĐỊA ĐIỂM	Mã số ĐV	Địa điểm ĐV
	1	Hà nội
	4	Hà nội
	5	Nam định
	5	Hà nội
	5	Bắc ninh

DỰ ÁN	Tên DA	Mã số DA	Địa điểm DA	Mã số ĐV
	DA01	1	Hà nội	5
	DA02	2	Nam định	5
	DA03	3	Bắc Ninh	5
	DA04	10	Hà nội	4
	DA05	20	Hà nội	1
	DA06	30	Hà nội	4

NHÂN VIÊN_ DỰ ÁN	Mã số NV	Mã số DA	Số giờ
	NV001	1	32
	NV001	2	7
	NV016	3	40
	NV018	1	20
	NV018	2	20
	NV002	2	10
	NV002	3	10
	NV002	10	10
	NV002	20	10

NV010	30	30
NV010	10	10
NV025	10	35
NV025	30	5
NV014	30	20
NV014	20	15
NVO61	20	null

CON	Mã số NV	Tên con	Giới tính	Ngày sinh
	NV002	Giang	Nữ	04/05/1997
	NV002	Bình	Nam	25/10/1994
	NV002	Hoa	Nữ	03/05/1969
	NV014	Lan	Nữ	29/02/1953
	NV001	Bình	Nam	04/01/1999
	NV001	Hòa	Nữ	04/01/1999
	NV001	Hương	Nữ	05/05/1981

Hình III-3. Cơ sở dữ liệu “CÔNG TY”

Trong một lược đồ cơ sở dữ liệu quan hệ, các thuộc tính biểu diễn cùng một khái niệm thế giới thực có thể (hoặc không) có cùng tên như nhau trong các quan hệ khác nhau. Ngược lại, các thuộc tính biểu diễn các khái niệm khác nhau có thể có tên như nhau trong các quan hệ khác nhau. Ví dụ, trong cơ sở dữ liệu CÔNG TY ở trên, các thuộc tính Mã số NV, Mã số NGS, Mã số NQL có tên khác nhau nhưng đều biểu diễn một khái niệm đó là mã số nhân viên (bởi vì người giám sát hoặc người quản lý cũng là nhân viên). Trong lúc đó, thuộc tính Giới tính có mặt trong hai quan hệ NHÂN VIÊN và CON, tuy nhiên Giới tính trong quan hệ NHÂN VIÊN là biểu thị giới tính của nhân viên còn Giới tính trong CON là biểu thị giới tính của người con.

Trong một số phiên bản trước của mô hình quan hệ, người ta yêu cầu rằng các thuộc tính biểu diễn cùng một khái niệm của thế giới thực thì phải có tên như nhau

trong mọi quan hệ. Điều đó sẽ gây ra khó khăn khi cùng một khái niệm thể giới thực được sử dụng trong các vai trò khác nhau.

Một hệ quản trị cơ sở dữ liệu phải có ngôn ngữ định nghĩa dữ liệu (Data definition language DDL) để định nghĩa lược đồ cơ sở dữ liệu quan hệ. Các hệ quản trị cơ sở dữ liệu hiện nay hầu như sử dụng SQL cho mục đích này.

Các ràng buộc toàn vẹn được chỉ ra trên một lược đồ cơ sở dữ liệu và được tôn trọng làm thoả mãn trên mỗi trạng thái cơ sở dữ liệu của lược đồ này. Ngoài các ràng buộc miền và ràng buộc khoá còn có thêm các ràng buộc được xem như một phần của mô hình quan hệ, đó là ràng buộc toàn vẹn thực thể và ràng buộc toàn vẹn tham chiếu.

II.4- Toàn vẹn thực thể, toàn vẹn tham chiếu và khoá ngoài

Ràng buộc toàn vẹn thực thể được phát biểu là: khoá chính phải luôn luôn có giá trị xác định, nghĩa là không được phép có giá trị null. Sở dĩ có điều đó là do giá trị của khoá chính được sử dụng để xác định các bộ giá trị riêng biệt trong một quan hệ. Việc có giá trị null cho khoá chính kéo theo việc chúng ta không thể xác định được một số bộ giá trị. Ví dụ, nếu có hai hay nhiều hơn các bộ giá trị có giá trị null cho khoá chính thì chúng ta không có khả năng phân biệt chúng.

Các ràng buộc khoá và ràng buộc toàn vẹn thực thể được chỉ ra trên các quan hệ riêng rẽ. *Ràng buộc toàn vẹn tham chiếu* được chỉ ra giữa hai quan hệ để duy trì sự tương ứng giữa các bộ của hai quan hệ. Một cách không hình thức, ràng buộc toàn vẹn tham chiếu được phát biểu là: một bộ giá trị trong một quan hệ có liên kết đến một quan hệ khác phải liên kết đến một bộ giá trị tồn tại trong quan hệ đó.

Để định nghĩa toàn vẹn tham chiếu một cách hình thức hơn, trước tiên chúng ta đưa ra khái niệm khoá ngoài: Một tập hợp các thuộc tính FK trong một lược đồ quan hệ R_1 là một khoá ngoài của R_1 tham chiếu đến quan hệ R_2 nếu nó thoả mãn hai quy tắc sau:

qt1. Các thuộc tính trong FK có cùng miền giá trị như các thuộc tính của khoá chính PK của R_2 . Các thuộc tính FK được gọi là tham chiếu đến (hoặc là liên hệ đến) quan hệ R_2 .

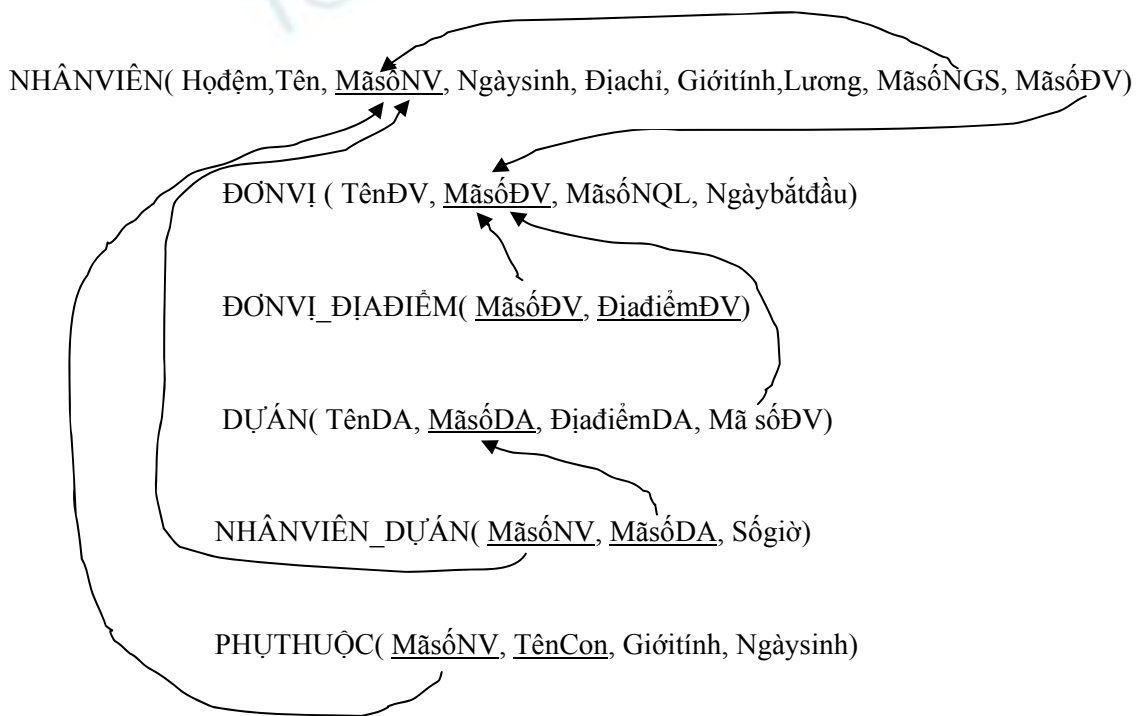
qt2. Một giá trị của FK trong một bộ t_1 của trạng thái hiện tại $r_1(R_1)$ hoặc có mặt như một giá trị của khoá chính của một bộ t_2 nào đó trong trạng thái hiện tại $r_2(R_2)$, hoặc là null. Trong trường hợp này ta có $t_1[FK] = t_2[PK]$ và ta nói rằng bộ t_1

liên hệ (tham chiếu) đến bộ t_2 . R_1 được gọi là quan hệ tham chiếu và R_2 được gọi là quan hệ bị tham chiếu.

Trong một cơ sở dữ liệu có nhiều quan hệ thường có nhiều ràng buộc toàn vẹn tham chiếu. Để chỉ ra các ràng buộc này, trước tiên ta phải có một hiểu biết rõ ràng về ý nghĩa hoặc vai trò của mỗi tập thuộc tính ở trong các lược đồ quan hệ khác nhau của cơ sở dữ liệu. Các ràng buộc toàn vẹn quy chiếu thường nảy sinh từ các mối liên kết giữa các thực thể được biểu diễn bằng các lược đồ quan hệ.

Chú ý rằng một khoá ngoài có thể tham chiếu đến quan hệ của chính nó. Trong trường hợp đó, khoá ngoài biểu thị một liên kết đệ quy.

Chúng ta có thể biểu diễn các ràng buộc tham chiếu bằng sơ đồ. Để làm điều đó ta vẽ một cạnh có hướng từ mỗi khoá ngoài đến quan hệ mà nó tham chiếu đến. Hình III-4 biểu diễn lược đồ ở hình 3.2 với các ràng buộc quy chiếu được biểu diễn theo cách này.



Hình III-4. Lược đồ và sơ đồ tham chiếu

Ngoài các ràng buộc toàn vẹn ở trên, cơ sở dữ liệu còn phải thoả mãn một số ràng buộc khác, như ràng buộc trạng thái, ràng buộc chuyển tiếp... Các ràng buộc trạng thái xác định các ràng buộc mà một trạng thái vững chắc của cơ sở dữ liệu phải thoả mãn. Ví dụ về các ràng buộc đó là: “lương của một nhân viên không được

vượt quá lương của người giám sát nhân viên đó” hoặc “số giờ nhiều nhất mà một nhân viên có thể làm việc trong một tuần trên tất cả các dự án là 56 giờ”. Các ràng buộc như vậy có thể được đặc tả và bắt tuân theo bằng cách sử dụng một ngôn ngữ đặc tả ràng buộc. Người ta có thể sử dụng các cơ cấu như là trigger hoặc assertion. Các ràng buộc chuyển tiếp có thể được định nghĩa để làm việc với những thay đổi trạng thái trong cơ sở dữ liệu. Ví dụ về ràng buộc này là: “ lương của một nhân viên chỉ có thể tăng”. Các ràng buộc như vậy thường được định nghĩa bằng cách sử dụng các quy tắc hoặc bằng các trigger.

III- Các phép toán trên mô hình quan hệ

Trong phần này chúng ta thảo luận về các phép toán của mô hình quan hệ. Các phép toán của mô hình quan hệ có thể phân thành hai loại: các phép toán cập nhật và các phép toán đại số quan hệ. Các phép toán cập nhật được sử dụng để tạo ra một quan hệ đúng đắn. Các phép toán đại số quan hệ được sử dụng để đặc tả các phép lấy thông tin ra.

III.1- Các phép toán cập nhật

Các phép toán cập nhật gồm ba phép toán cơ bản là chèn, xoá và sửa đổi. Phép chèn được dùng để chèn một bộ giá trị hoặc nhiều bộ giá trị vào một quan hệ. Phép xoá dùng để loại bỏ các bộ giá trị và phép sửa đổi dùng để sửa đổi các giá trị của một số thuộc tính trong các bộ giá trị đã có. Mỗi khi các phép toán cập nhật được áp dụng, các ràng buộc trên lược đồ cơ sở dữ liệu có thể bị vi phạm. Trong phần này chúng ta sẽ nói đến khả năng vi phạm các ràng buộc của từng phép toán và các kiểu hành động có thể thực hiện khi một ràng buộc bị vi phạm.

III.1.1- Phép chèn (Insert)

Phép chèn cung cấp một danh sách các giá trị cho một bộ mới t được chèn vào trong một quan hệ R . Phép chèn có thể vi phạm các kiểu ràng buộc được mô tả ở trên. Các ràng buộc miền có thể bị vi phạm nếu một giá trị thuộc tính được cho không thuộc vào miền tương ứng. Các ràng buộc khoá có thể bị vi phạm nếu một giá trị khoá trong bộ mới t đã tồn tại trong một bộ khác ở trong quan hệ $r(R)$. Sự toàn vẹn thực thể có thể bị vi phạm nếu khoá chính của bộ mới t là null. Sự toàn vẹn tham chiếu có thể bị vi phạm nếu một giá trị của một khoá ngoài trong t tham chiếu đến một bộ không tồn tại trong một quan hệ được tham chiếu. Ví dụ (với các bảng trong cơ sở dữ liệu CÔNGTY) :

- Chèn bộ giá trị <null, 'Vũ', 'Hải', '15/07/81', 'Hà Nội', 'Nam', 3200, null, 4> vào quan hệ NHÂNVIÊN. Phép chèn này vi phạm ràng buộc toàn vẹn thực thể (giá trị null cho khoá chính). Phép chèn bị loại bỏ.
- Chèn bộ giá trị <'NV002', 'Trương', 'Phi', '15/07/81', 'Hà Nội', 'Nam', 3200, 'NV067', 4> vào quan hệ NHÂNVIÊN. Phép chèn này vi phạm ràng buộc khoá, bởi vì giá trị 'NV002' đã có ở trong bảng. Phép chèn bị loại bỏ.
- Chèn bộ giá trị <'NV072', 'Vũ', 'Hải', '15/07/81', 'Hà Nội', 'Nam', 3200, 'NV002', 7> vào quan hệ NHÂNVIÊN. Phép chèn này vi phạm ràng buộc toàn vẹn tham chiếu, trong quan hệ ĐƠN VỊ không có đơn vị có mã số = 7. Phép chèn bị loại bỏ.
- Chèn bộ giá trị <'NV045', 'Vũ', 'Hải', '15/07/81', 'Hà Nội', 'Nam', 3200, 'NV002', 4> vào quan hệ NHÂNVIÊN. Phép chèn thoả mãn tất cả các ràng buộc, vì thế nó được chấp nhận.

Nếu một phép chèn vi phạm một hoặc nhiều ràng buộc, tùy chọn mặc định là loại bỏ phép chèn. Trong trường hợp này, thường là các hệ quản trị cơ sở dữ liệu có thể thông báo cho người sử dụng nguyên nhân của việc loại bỏ phép chèn.

III.1.2- Phép xoá (Delete)

Phép xoá được sử dụng để xoá một hoặc nhiều bộ giá trị của một quan hệ. Phép xoá chỉ có thể vi phạm ràng buộc tham chiếu trong trường hợp bộ bị xoá được tham chiếu bởi một khoá ngoài từ các bộ khác trong cơ sở dữ liệu. Để chỉ rõ một phép xoá, cần phải đưa ra một điều kiện trên các thuộc tính của quan hệ để chọn các bộ sẽ bị xoá. Ví dụ:

- Xoá một bộ giá trị của quan hệ NHÂNVIÊN_DỰÁN có Mã số NV = 'NV010' và Mã số DA = 10. Phép xoá này được chấp nhận.
- Xoá bộ giá trị của NHÂNVIÊN có Mã số NV = 'NV010'. Phép xoá này không chấp nhận được bởi vì có các bộ trong NHÂNVIÊN_DỰÁN tham chiếu đến bộ này, như vậy là vi phạm ràng buộc toàn vẹn tham chiếu.
- Xoá bộ giá trị của NHÂNVIÊN có Mã số NV = 'NV002'. Phép xoá này cũng vi phạm ràng buộc toàn vẹn tham chiếu.

Ba tùy chọn được sẵn sàng được sử dụng nếu một phép xoá gây ra sự vi phạm. Tùy chọn thứ nhất là loại bỏ phép xoá. Tùy chọn thứ hai là cố gắng lan truyền phép

xoá (cascade the deletion) bằng cách xoá đồng thời các bộ tham chiếu đến bộ bị xoá. Tùy chọn thứ ba là sửa đổi các giá trị của các thuộc tính tham chiếu gây ra sự vi phạm. Mỗi giá trị như vậy hoặc là làm cho bằng null hoặc được thay đổi thành bộ có hiệu lực tham chiếu khác. Chú ý rằng, nếu một thuộc tính tham chiếu gây ra sự vi phạm là một phần của khoá chính thì không thể làm cho thành null, bởi vì nếu làm vậy thì sẽ vi phạm ràng buộc toàn vẹn thực thể. Có thể kết hợp cả ba tùy chọn ở trên.

III.1.3- Phép sửa đổi (Update)

Phép toán sửa đổi được dùng để thay đổi các giá trị của một hoặc nhiều thuộc tính trong một (hoặc nhiều) bộ của một quan hệ R nào đấy. Để lựa chọn các bộ cần được thay đổi, người sử dụng phải chỉ ra một điều kiện trên các thuộc tính. Ví dụ:

- Sửa đổi Lương của bộ NHÂNVIÊN có MăsốNV = 'NV018' thành 2800. Phép sửa đổi này được chấp nhận.
- Sửa đổi MăsốĐV của bộ NHÂNVIÊN có MăsốNV = 'NV018' thành 7. Phép sửa đổi này vi phạm ràng buộc toàn vẹn tham chiếu.
- Sửa đổi MăsốNV của bộ NHÂNVIÊN có MăsốNV = 'NV018' thành 'NV014'. Phép sửa đổi này vi phạm ràng buộc toàn vẹn thực thể và toàn vẹn tham chiếu.

Việc sửa đổi một thuộc tính không phải là một khoá chính hoặc một khoá ngoài thường không gây ra các vi phạm ràng buộc, hệ quản trị cơ sở dữ liệu chỉ cần kiểm tra để khẳng định rằng giá trị mới là thuộc miền và kiểu giá trị đúng đắn. Việc sửa đổi giá trị một khoá chính tương tự như việc xoá một bộ và chèn bộ khác vào chỗ của nó. Như vậy chúng ta trở về trường hợp đã thảo luận với phép chèn và phép xoá. Nếu một thuộc tính khoá ngoài bị sửa đổi thì hệ quản trị cơ sở dữ liệu phải đảm bảo rằng giá trị mới tham chiếu đến một bộ có tồn tại trong quan hệ được tham chiếu (hoặc là null).

III.2- Các phép toán đại số quan hệ

Ngoài việc định nghĩa cấu trúc cơ sở dữ liệu và các ràng buộc, một mô hình dữ liệu phải chứa một tập hợp phép toán để thao tác dữ liệu. Tập hợp cơ sở các phép toán mô hình quan hệ tạo nên đại số quan hệ. Các phép toán này giúp cho người sử dụng xác định rõ các yêu cầu lấy tin cơ bản. Kết quả của một phép lấy tin

là một quan hệ mới, có thể được tạo ra từ một hoặc nhiều quan hệ. Các quan hệ đó có thể được thao tác tiếp theo bằng cách sử dụng các phép toán của cùng đại số. Một dãy các phép toán quan hệ tạo nên một biểu thức đại số quan hệ mà kết quả của nó cũng là một quan hệ.

Các phép toán đại số quan hệ được chia thành hai nhóm. Một nhóm bao gồm các phép toán tập hợp lấy từ lý thuyết tập hợp toán học. Các phép toán đó là phép hợp, phép giao, phép trừ tập hợp và phép tích Đề các. Nhóm kia bao gồm các phép toán được xây dựng đặc biệt cho các cơ sở dữ liệu quan hệ. Các phép toán đó là phép chọn, phép chiếu, phép nối và một số các phép toán khác.

III.2.1- Phép chọn (SELECT)

Phép chọn đưallọc sử dụng để chọn một tập hợp các bộ thoả mãn điều kiện chọn từ một quan hệ. Ta có thể xem phép chọn như một bộ lọc, nó chỉ giữ lại các bộ thoả mãn điều kiện đặt ra.

Phép chọn được ký hiệu là

$$\sigma_{\langle \text{điều kiện chọn} \rangle}(R)$$

trong đó ký hiệu σ được dùng để ký hiệu phép chọn, còn điều kiện chọn là một biểu thức logic được chỉ ra trên các thuộc tính của R. Chú ý rằng R nói chung là một biểu thức đại số quan hệ. Kết quả của một biểu thức đại số quan hệ là một quan hệ. Biểu thức đơn giản nhất chính là tên của một quan hệ của một cơ sở dữ liệu. Quan hệ kết quả của phép chọn có cùng thuộc tính như R. Ví dụ, để chọn các bộ NHÂNVIÊN thuộc về đơn vị có mã số là 4 hoặc các bộ NHÂNVIÊN có lương lớn hơn 3000 ta có thể viết một cách riêng rẽ như sau:

$$\sigma_{\langle \text{Mã số} = 4 \rangle}(\text{NHÂNVIÊN})$$

$$\sigma_{\langle \text{Lương} > 3000 \rangle}(\text{NHÂNVIÊN})$$

Biểu thức logic chỉ ra trong $\langle \text{điều kiện chọn} \rangle$ được tạo nên từ một số hạng mục có dạng :

$\langle \text{tên thuộc tính} \rangle \langle \text{phép so sánh} \rangle \langle \text{giá trị hằng} \rangle$

hoặc $\langle \text{tên thuộc tính} \rangle \langle \text{phép so sánh} \rangle \langle \text{tên thuộc tính} \rangle$

trong đó $\langle \text{tên thuộc tính} \rangle$ là tên của một thuộc tính trong R, $\langle \text{phép so sánh} \rangle$ là một trong các phép toán so sánh $\{<, <=, =, >=, >, \neq\}$ còn $\langle \text{giá trị hằng} \rangle$ là một giá

trị hằng từ miền giá trị của thuộc tính. Các hạng mục có thể được nối với nhau bằng các phép toán lô gic AND, OR, NOT để tạo ra một điều kiện chọn chung. Ví dụ, để chọn ra các nhân viên làm việc ở đơn vị có mã số là 4 và có lương lớn hơn 3000 hoặc các nhân viên làm việc ở đơn vị có mã số là 5 và có lương lớn hơn 4000 ta có thể viết phép chọn như sau:

$$\sigma_{\langle \text{Mã số ĐV} = 4 \rangle \text{ AND } \langle \text{lương} > 3000 \rangle \text{ OR } \langle \text{Mã số ĐV} = 5 \rangle \text{ AND } \langle \text{lương} > 3500 \rangle}(\text{NHÂNVIÊN})$$

Kết quả chỉ ra ở hình III-5.

Mã số NV	Họ đệm	Tên	Ngày sinh	Địa chỉ	Giới tính	Lương	Mã số NGS	Mã số ĐV
NV002	Trần Đức	Nam	14/02/66	Hà nội	Nam	4000	NV061	5
NV014	Phạm	Bằng	26/06/52	Bắc ninh	Nam	4300	NV061	4
NV016	Nguyễn	Son	14/08/73	Hà nam	Nam	3800	NV002	5

Hình III-5. Kết quả phép chọn

Chú ý rằng các phép toán so sánh trong tập hợp $\{<, <=, =, >=, >, \neq\}$ áp dụng cho các thuộc tính có miền giá trị là các giá trị có thứ tự như là miền giá trị số. Miền giá trị các dãy ký tự được xem như có thứ tự dựa trên việc so sánh các dãy ký tự. Nếu miền giá trị của một thuộc tính là một tập hợp các giá trị không có thứ tự thì chỉ có các phép so sánh trong tập hợp $\{=, \neq\}$ là có thể áp dụng được. Ngoài ra, có thể còn các phép so sánh bổ sung, chẳng hạn như “là một dãy con của...” hoặc “trong khoảng từ... đến...”.

Kết quả một phép chọn được xác định như sau: <Điều kiện chọn> được áp dụng cho mỗi bộ t trong R một cách độc lập. Điều đó được thực hiện bằng cách thay thế mỗi thuộc tính A_i trong điều kiện chọn bằng giá trị $t[A_i]$ của nó trong bộ. Nếu điều kiện chọn cho giá trị đúng thì bộ t sẽ được chọn. Tất cả các bộ được chọn xuất hiện trong kết quả của phép chọn. Các phép toán logic AND, OR, NOT được thực hiện theo quy tắc bình thường của chúng.

Phép chọn là phép toán một ngôi, nghĩa là nó được áp dụng cho một quan hệ. Hơn nữa, phép chọn được áp dụng cho từng bộ một cách độc lập, vì vậy, các điều kiện chọn không thể liên quan đến nhiều bộ. Quan hệ kết quả của phép chọn có cấp giống như cấp của R. Số các bộ trong quan hệ kết quả luôn luôn nhỏ hơn hoặc bằng số các bộ trong R.

Phép chọn là một phép toán có tính chất giao hoán, nghĩa là

$$\sigma_{\langle \text{Điều kiện 1} \rangle} (\sigma_{\langle \text{Điều kiện 2} \rangle} (R)) = \sigma_{\langle \text{Điều kiện 2} \rangle} (\sigma_{\langle \text{Điều kiện 1} \rangle} (R))$$

Hơn nữa ta có thể kết hợp một loạt các phép chọn thành một phép chọn đơn giản bằng cách sử dụng phép toán AND. Ví dụ:

$$\sigma_{\langle \text{Điều kiện 1} \rangle} (\sigma_{\langle \text{Điều kiện 2} \rangle} (R)) = \sigma_{\langle \text{Điều kiện 2} \rangle \text{ AND } \langle \text{Điều kiện 1} \rangle} (R)$$

III.2.2- Phép chiếu (PROJECT)

Nếu ta coi một quan hệ như một bảng thì phép chọn chọn một số hàng của bảng thoả mãn điều kiện chọn và bỏ qua các hàng không thoả mãn điều kiện chọn. Phép chiếu là phép toán chọn một số cột của bảng. Nếu chúng ta chỉ quan tâm đến một số thuộc tính của quan hệ, chúng ta dùng phép chiếu để chiếu lên các thuộc tính đó. Phép chiếu được ký hiệu là:

$$\pi_{\langle \text{danh sách các thuộc tính} \rangle} (R)$$

trong đó π là ký hiệu dùng để biểu diễn phép chiếu và $\langle \text{danh sách các thuộc tính} \rangle$ là một danh sách con các thuộc tính của quan hệ R. Nói chung R là một biểu thức đại số quan hệ. Trường hợp đơn giản nhất nó là tên của một quan hệ của cơ sở dữ liệu. Kết quả của phép chiếu là một quan hệ chỉ có các thuộc tính nằm trong $\langle \text{danh sách các thuộc tính} \rangle$ và có cùng thứ tự như thứ tự của chúng có trong danh sách. Như vậy, cấp của quan hệ kết quả là số các thuộc tính có trong $\langle \text{danh sách các thuộc tính} \rangle$.

Nếu $\langle \text{danh sách các thuộc tính} \rangle$ chỉ bao gồm các thuộc tính không phải thuộc tính khoá của R thì quan hệ kết quả có thể có những bộ trùng nhau. Phép chiếu loại bỏ mọi bộ trùng lặp, và như vậy, kết quả của phép chiếu là một tập hợp các bộ và là một quan hệ đúng đắn.

Ví dụ, phép chiếu:

$$\pi_{\langle \text{Mã số NV, Họ tên, Tên, Lương} \rangle} (\text{NHÂNVIÊN})$$

cho kết quả là một quan hệ có các thuộc tính Mã số NV, Họ tên, Tên, Lương (hình III-6).

Mã số NV	Họ tên	Tên	Địa chỉ	Lương
NV001	Lê	Vân	Hà nội	3000

NV002	Trần Đức	Nam	Hà nội	4000
NV010	Hoàng	Thanh	Nghệ an	2500
NV014	Phạm	Bằng	Bắc ninh	4300
NV016	Nguyễn	Sơn	Hànam	3800
NV018	Vũ Hương	Giang	Nam định	2500
NV025	Trần Lê	Hoa	Phúthọ	2500
NV061	Hoàng	Giáp	Hà tĩnh	5500

Hình III-6. Kết quả phép chiếu

Số các bộ trong quan hệ kết quả từ một phép chiếu luôn luôn nhỏ hơn hoặc bằng số các bộ trong R. Nếu danh sách chiếu là một siêu khoá của R (nghĩa là nó chứa một khoá nào đó của R) thì quan hệ kết quả có cùng một số bộ như R. Ngoài ra, nếu <danh sách 2> chứa tất cả các thuộc tính có trong <danh sách 1> thì

$$\pi_{\langle \text{danh sách 1} \rangle}(\pi_{\langle \text{danh sách 2} \rangle}(R)) = \pi_{\langle \text{danh sách 1} \rangle}(R)$$

Phép chiếu không có tính giao hoán.

III.2.3- Phép đặt lại tên (RENAME)

Chúng ta có thể áp dụng nhiều phép toán quan hệ liên tiếp nhau. Trong trường hợp đó hoặc chúng ta có thể viết các phép toán như là một biểu thức đại số quan hệ đơn bằng cách xếp lồng các phép toán lại với nhau, hoặc chúng ta có thể áp dụng mỗi phép toán tại một thời điểm và tạo ra các quan hệ kết quả trung gian. Trong trường hợp tạo các quan hệ trung gian, ta phải đặt tên cho quan hệ đó. Ví dụ: Để đưa ra Họ tên và Lương của các Nhân viên làm việc ở đơn vị có Mã số là 4 chúng ta phải áp dụng một phép chọn và một phép chiếu. Chúng ta có thể viết một biểu thức đại số quan hệ đơn như sau :

$$\pi_{\langle \text{Họ tên, Lương} \rangle}(\sigma_{\langle \text{Mã số} = 4 \rangle}(\text{NHÂNVIÊN}))$$

Một cách khác, chúng ta có thể tạo ra kết quả trung gian và viết biểu thức trên thành dãy các phép toán như sau:

$$\text{KQTG} \leftarrow \sigma_{\langle \text{Mã số} = 4 \rangle}(\text{NHÂNVIÊN})$$

$$\text{Ketqua} \leftarrow \pi_{\langle \text{Họ tên, Lương} \rangle}(\text{KQTG})$$

Thông thường việc phân tích một dãy phức tạp các phép toán bằng cách chỉ ra các quan hệ kết quả trung gian là dễ hơn việc viết một biểu thức đại số quan hệ đơn. Chúng ta có thể dùng kỹ thuật này để đặt lại tên (rename) cho các thuộc tính trong các quan hệ trung gian và kết quả. Để đặt lại tên cho các thuộc tính của một quan hệ, chúng ta liệt kê các tên mới của các thuộc tính trong cặp dấu ngoặc. Ví dụ:

$$R(\text{Họ và tên}, \text{Lương}) \leftarrow \pi_{\langle \text{Họ tên}, \text{Lương} \rangle}(KQTG)$$

Cho kết quả là quan hệ R, trong đó thuộc tính Họ tên được đặt lại tên thành Họ và tên.

Nếu không có việc đặt lại tên thì tên của các thuộc tính trong quan hệ kết quả của một phép chọn là giống như các tên trong quan hệ ban đầu và có cùng một thứ tự như thứ tự của các thuộc tính đó. Đối với phép chiếu, nếu không có việc đặt lại tên thì quan hệ kết quả có các tên thuộc tính giống như các tên trong danh sách chiếu và có cùng thứ tự như chúng xuất hiện trong danh sách.

Chúng ta có thể định nghĩa một phép toán đặt lại tên, nó có thể đặt lại tên cho một tên quan hệ hoặc các tên thuộc tính hoặc cả hai. Phép đặt lại tên được ký hiệu là:

$$\rho_{S(B_1, B_2, \dots, B_n)}(R) \quad \text{hoặc} \quad \rho_S(R) \quad \text{hoặc} \quad \rho_{(B_1, B_2, \dots, B_n)}(R)$$

trong đó ký hiệu ρ được dùng để ký hiệu phép toán đặt lại tên, S là tên quan hệ mới, $B_1, B_2, \dots, B_n$ là các tên thuộc tính mới. Biểu thức thứ nhất đặt lại tên quan hệ và các thuộc tính của nó. Nếu các thuộc tính của R là $A_1, A_2, \dots, A_n$ thì sau khi đặt lại tên, quan hệ có tên mới là S còn các thuộc tính có tên mới là $B_1, B_2, \dots, B_n$. Biểu thức thứ hai chỉ đặt lại tên quan hệ, nghĩa là sau phép đặt lại tên, quan hệ có tên mới là S, còn các thuộc tính vẫn mang tên cũ. Biểu thức thứ ba chỉ đặt lại tên các thuộc tính, nếu các thuộc tính của R là $A_1, A_2, \dots, A_n$ thì sau khi đặt lại tên chúng có tên là $B_1, B_2, \dots, B_n$.

III.2.4- Các phép toán lý thuyết tập hợp

Nhóm tiếp theo của các phép toán đại số quan hệ là các phép toán toán học thông thường trên các tập hợp. Đó là các phép toán hợp, giao và trừ tập hợp. Các phép toán này là các phép toán hai ngôi, nghĩa là mỗi phép toán được áp dụng cho hai tập hợp. Khi áp dụng các phép toán này cho cơ sở dữ liệu quan hệ, hai quan hệ tham gia vào một trong các phép toán trên phải có kiểu của các bộ như nhau, hay nói cách khác, chúng phải có cùng một cấu trúc. Điều kiện này được gọi là *trùng*