



CSDL Dong Thi Bich Thuy 2020

Cơ sở dữ liệu (Đại học Khoa học Tự nhiên, Đại học Quốc gia Thành phố Hồ Chí Minh)



Scan to open on Studeersnel

ĐỒNG THỊ BÍCH THỦY
PHẠM THỊ BẠCH TUỆ - NGUYỄN TRẦN MINH THƯ

GIÁO TRÌNH CƠ SỞ DỮ LIỆU



NHÀ XUẤT BẢN KHOA HỌC KỸ THUẬT

ĐỒNG THỊ BÍCH THỦY

PHẠM THỊ BẠCH HUỆ - NGUYỄN TRẦN MINH THƯ

GIÁO TRÌNH

CƠ SỞ DỮ LIỆU



NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT
NĂM 2020

Lời nói đầu

Giáo trình Cơ sở dữ liệu đã được viết để phục vụ cho sinh viên theo học môn Cơ sở dữ liệu tại khoa Công nghệ thông tin trường Đại học Khoa học Tự nhiên (Đại học Quốc gia Thành phố Hồ Chí Minh). Nội dung giáo trình và cách thức chúng tôi trình bày nội dung này bám sát mục tiêu đề ra cho môn học. Đó là cung cấp kiến thức căn bản nhất về mô hình dữ liệu quan hệ, các ngôn ngữ truy vấn và các nguyên tắc để tối ưu hóa câu truy vấn. Đôi với hai cụm kiến thức đầu, chúng tôi tập trung trình bày sao cho bạn đọc nắm bắt được và biết sử dụng ở mức độ người dùng. Cụ thể là bạn đọc sẽ hiểu được một lược đồ cơ sở dữ liệu quan hệ, từ đó có thể phát biểu các ràng buộc toàn vẹn và đánh giá chất lượng lược đồ cơ sở dữ liệu liên quan đến yếu tố trùng lặp thông tin. Để có thể khai thác được một cách cụ thể với một phần mềm quản trị cơ sở dữ liệu, ngôn ngữ truy vấn SQL cũng được trình bày. Nội dung liên quan đến tối ưu hóa câu truy vấn được trình bày ở mức nhập môn, để bạn đọc hiểu được cách thức một hệ quản trị cơ sở dữ liệu thực hiện một câu truy vấn.

Chúng tôi hy vọng cuốn giáo trình này sẽ giúp sinh viên nắm bắt một cách chính xác những kiến thức nền tảng về cơ sở dữ liệu, hỗ trợ cho việc nghe giảng được dễ dàng và thông suốt. Mặc dù nội dung giáo trình đã được hoàn thiện dần qua nhiều năm giảng dạy bởi nhiều giảng viên trong khoa Công nghệ thông tin, nhưng phiên bản đầu tiên này chắc chắn không tránh khỏi những khiếm khuyết cũng như có thể có những phần được trình bày chưa đủ rõ cho bạn đọc. Chúng tôi mong sẽ nhận được những góp ý từ bạn đọc và sinh viên để những phiên bản sau được tốt hơn.

Tp. Hồ Chí Minh, tháng 4/2010

Nhóm tác giả

Mục lục

Lời nói đầu	2
Mục lục	3
Chương 1 TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU	11
I. GIỚI THIỆU	11
I.1 Một số ví dụ mở đầu về dữ liệu	11
I.2 Định nghĩa dữ liệu, CSDL, hệ quản trị CSDL, hệ CSDL	12
I.2.1 Dữ liệu là gì?	12
I.2.2 CSDL là gì?	12
I.2.3 Định nghĩa hệ quản trị CSDL	13
I.2.4 Định nghĩa hệ CSDL	13
I.3 Ví dụ cụ thể một CSDL	14
II. QUÁ TRÌNH PHÁT TRIỂN HỆ CSDL	16
II.1 Tiếp cận theo tập tin	16
II.2 Tiếp cận theo CSDL	18
III. MỘT SỐ ĐẶC TRƯNG CỦA HƯỚNG TIẾP CẬN CSDL	19
III.1 Tính tự mô tả	19
III.2 Tính độc lập	19
III.3 Tính trừu tượng	19
III.4 Tính nhất quán	19
III.5 Đa khung nhìn	20
IV. NGƯỜI SỬ DỤNG CƠ SỞ DỮ LIỆU	20
IV.1 Quản trị viên	20
IV.2 Thiết kế viên	20
IV.3 Người dùng cuối	21

V.	KIẾN TRÚC CỦA HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU	21
V.1	Kiến trúc ba lược đồ	21
V.2	Các thành phần của HQT CSDL	22
V.2.1	Các ngôn ngữ của HQT CSDL	22
V.2.2	Các giao tiếp của HQT CSDL	22
VI.	TÍNH NĂNG CỦA HQT CSDL.....	23
VI.1	Hạn chế sự trùng lặp	23
VI.2	Chia sẻ dữ liệu	23
VI.3	Hạn chế truy cập không được phép	23
VI.4	Cung cấp nhiều giao diện giao tiếp	23
VI.5	Đảm bảo ràng buộc toàn vẹn	24
VI.6	Khả năng sao lưu dự phòng.....	24
VI.7	Một số tính năng khác	24
VI.7.1	Chuẩn hóa	24
VI.7.2	Uyển chuyển	24
VI.7.3	Khả dụng.....	24
VI.7.4	Giảm thời gian phát triển ứng dụng	24
VII.	MÔ HÌNH DỮ LIỆU	25
VII.1	Định nghĩa	25
VII.2	Giới thiệu các loại mô hình dữ liệu	25
VII.2.1	Mô hình thực thể kết hợp	25
VII.2.2	Mô hình quan hệ.....	26
VII.2.3	Mô hình hướng đối tượng	27
VII.2.4	Mô hình mạng	28
VII.2.5	Mô hình phân cấp.....	29
VIII.	MỘT ỨNG DỤNG CƠ SỞ DỮ LIỆU	29

Chương 2 MÔ HÌNH THỰC THỂ KẾT HỢP	31
I. MÔ HÌNH DỮ LIỆU CẤP CAO.....	31
II. THỰC THỂ VÀ TẬP THỰC THỂ	31
III. THUỘC TÍNH	32
IV. MỐI KẾT HỢP VÀ TẬP MỐI KẾT HỢP	34
IV.1 Tập mối kết hợp một - một.....	35
IV.2 Tập mối kết hợp một nhiều.....	35
IV.3 Tập mối kết hợp nhiều – một.....	36
IV.4 Tập mối kết hợp nhiều – nhiều	37
IV.5 Bản số	37
IV.6 Thuộc tính trên tập mối kết hợp	38
IV.7 Tên vai trò.....	38
V. KHÓA CỦA TẬP THỰC THỂ	39
VI. TẬP THỰC THỂ YÊU	39
VII. VÍ DỤ MÔ HÌNH TT-KH	41
VIII. KẾT LUẬN	41
Chương 3 MÔ HÌNH DỮ LIỆU QUAN HỆ	48
I. GIỚI THIỆU	48
II. MÔ HÌNH DỮ LIỆU QUAN HỆ	48
II.1 Định nghĩa quan hệ	48
II.2 Một số tính chất của quan hệ.....	50
III. Các khái niệm về khóa	51
IV. CHUYỂN ĐỔI MÔ HÌNH TT-KH SANG MÔ HÌNH QUAN HỆ	53
IV.1 Tập thực thể sang quan hệ	53
IV.2 Tập mối kết hợp 2 ngôi sang quan hệ.....	53

IV.3	Tập thực thể yêu sang quan hệ	56
IV.4	Thuộc tính đa trị	57
V.	KẾT LUẬN	58
Chương 4 PHỤ THUỘC HÀM VÀ DẠNG CHUẨN		63
I.	PHỤ THUỘC HÀM	63
I.1	Định nghĩa phụ thuộc hàm	63
I.2	Một số tính chất của phụ thuộc hàm	64
II.	DẠNG CHUẨN	65
II.1	Thiết kế lược đồ CSDL tốt	65
II.1.1	Trùng lắp các giá trị trên các bộ	65
II.1.2	Giá trị rỗng trong các bộ	66
II.2	Tiếp cận phân rã quan hệ	66
II.3	Dạng chuẩn 1	66
II.4	Dạng chuẩn 2	68
II.5	Dạng chuẩn 3	71
II.6	Dạng chuẩn BCK (Boyce Codd Kent)	72
II.7	Dạng chuẩn của lược đồ CSDL	74
II.8	Kết luận	75
Chương 5 NGÔN NGỮ ĐẠI SỐ QUAN HỆ		80
I.	GIỚI THIỆU	80
I.1	Khái niệm đại số quan hệ	80
I.2	Các phép toán đại số quan hệ	80
II.	CÁC PHÉP TOÁN ĐẠI SỐ QUAN HỆ	81
II.1	Phép hội	81
II.2	Phép giao	82
II.3	Phép hiệu	82

II.4	Phép tích Đè-các.....	83
II.5	Phép chọn	84
II.6	Phép chiếu	85
II.7	Phép gán	87
II.8	Phép đổi tên	87
II.9	Phép kết	88
II.10	Phép chia.....	92
II.11	Tập hợp đầy đủ	94
III.	CÁC HÀM TÍNH TỔNG HỢP VÀ GOM NHÓM	94
IV.	PHÉP TOÁN CẬP NHẬT DỮ LIỆU.....	95
IV.1	Thêm.....	95
IV.2	Xóa.....	96
IV.3	Sửa	96
V.	KẾT LUẬN.....	96
Chương 6 NGÔN NGỮ SQL		106
I.	GIỚI THIỆU	106
II.	NGÔN NGỮ ĐỊNH NGHĨA DỮ LIỆU VÀ KIỂU DỮ LIỆU.....	107
II.1	Lệnh CREATE TABLE	107
II.2	Kiểu dữ liệu	108
II.3	Lệnh CREATE DOMAIN.....	109
III.	KHAI BÁO RÀNG BUỘC CƠ BẢN TRONG SQL	109
III.1	Ràng buộc trên thuộc tính.....	111
III.2	Ràng buộc khóa chính và ràng buộc khóa ngoại	112
III.3	Đặt tên cho ràng buộc	112
IV.	CÁC LỆNH CẬP NHẬT LUỢC ĐỒ	113
IV.1	Lệnh ALTER	113

IV.2	Lệnh DROP	115
V.	TRUY VẤN CƠ BẢN TRONG SQL.....	116
V.1	Cấu trúc SELECT-FROM-WHERE	116
V.2	Thuộc tính trùng, cách đặt bí danh và các biến bộ	118
V.3	Lệnh SELECT không sử dụng mệnh đề WHERE và sử dụng ký tự đại diện.....	120
V.4	Xử lý kết quả trả về có bộ trùng.....	123
V.5	So sánh chuỗi và các phép toán số học	124
V.6	Sắp xếp kết quả trả về với mệnh đề ORDER BY	127
V.7	Phép toán tập hợp trong SQL	129
VI.	TRUY VẤN PHỨC TẠP TRONG SQL	131
VI.1	So sánh với giá trị NULL và UNKNOWN	131
VI.2	Truy vấn lồng.....	133
VI.2.1	Truy vấn lồng phân cấp.....	134
VI.2.2	Truy vấn lồng tương quan.....	136
VI.3	Sử dụng phép kết – JOIN ở mệnh đề FROM	139
VI.4	Hàm tính tổng hợp trong SQL	140
VI.5	Mệnh đề GROUP BY và HAVING	142
VI.6	Kết luận về truy vấn trong SQL.....	143
VII.	CÁC LỆNH CẬP NHẬT DỮ LIỆU	144
VII.1	Lệnh thêm dữ liệu - INSERT.....	144
VII.2	Lệnh xóa dữ liệu – DELETE	146
VII.3	Lệnh sửa dữ liệu - UPDATE	147
VIII.	KẾT LUẬN	147
Chương 7 PHÉP TÍNH QUAN HỆ		150
I.	GIỚI THIỆU	150

II. PHÉP TÍNH QUAN HỆ CÓ BIẾN LÀ BỘ	151
II.1 Biến bộ và quan hệ miền	151
II.2 Biểu thức và công thức.....	152
II.3 Lượng từ <i>với mọi</i> (\forall) và <i>tồn tại</i> (\exists)	153
II.4 Biến tự do và biến két buộc.....	154
II.5 Ví dụ	155
II.6 Biểu thức an toàn.....	156
III. PHÉP TÍNH QUAN HỆ CÓ BIẾN LÀ MIỀN	157
III.1 Giới thiệu	157
III.2 Biểu thức.....	157
III.3 Ví dụ	158
IV. KẾT LUẬN	159
Chương 8 RÀNG BUỘC TOÀN VẸN	162
I. GIỚI THIỆU	162
II. KHÁI NIỆM VỀ RBTV	162
III. TẠI SAO PHẢI CÓ RBTV	163
IV. CÁC ĐẶC TRƯNG CỦA RBTV	163
IV.1 Bối cảnh	163
IV.2 Nội dung của RBTV	164
IV.3 Bảng tầm ảnh hưởng	164
V. PHÂN LOẠI RBTV	165
V.1 RBTV trên một quan hệ	165
V.1.1 RBTV miền giá trị	165
V.1.2 RBTV liên bộ trên một quan hệ	166
V.1.3 RBTV liên thuộc tính trên một quan hệ	167
V.2 RBTV trên nhiều quan hệ.....	168

V.2.1	RBTV tham chiếu	168
V.2.2	RBTV liên bộ trên nhiều quan hệ	169
V.2.3	RBTV liên thuộc tính liên quan hệ	170
V.2.4	RBTV trên thuộc tính tổng hợp	171
VI.	CÀI ĐẶT RBTV VỚI SQL	172
VI.1	Cài đặt RBTV với Assertion.....	172
VI.2	Cài đặt RBTV với trigger	173
VI.3	Cài đặt RBTV với Stored Procedure	175
VII.	KẾT LUẬN	176
Chương 9 TỐI ƯU TRUY VẤN		180
I.	GIỚI THIỆU	180
I.1	Quá trình xử lý truy vấn	180
I.2	Tối ưu hóa câu truy vấn.....	181
II.	TỐI ƯU TRUY VẤN DÙNG PHƯƠNG PHÁP DỰA TRÊN HEURISTIC.....	182
II.1	Thể hiện lại câu truy vấn	182
II.2	Các luật biến đổi tương đương	183
II.3	Thuật toán.....	185
II.4	Ví dụ minh họa.....	186
III.	TỐI ƯU TRUY VẤN DÙNG PHƯƠNG PHÁP ƯỚC LƯỢNG CHI PHÍ	190
III.1	Giới thiệu	190
III.2	Các loại chi phí và dữ liệu cần thiết	190
III.3	Ví dụ	192
IV.	KẾT LUẬN	192

1

Chương 1

TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU

I. GIỚI THIỆU

I.1 Một số ví dụ mở đầu về dữ liệu

Cơ sở dữ liệu (CSDL) và các hệ thống CSDL trở thành một thành phần quan trọng của cuộc sống chúng ta ngày nay. Hầu hết các hoạt động hàng ngày đều gắn liền với sự tương tác với một CSDL. Ví dụ, các giao dịch ngân hàng (như rút tiền, gửi tiền, chuyển khoản), đặt chỗ khách sạn, chuyến bay, hay truy cập đến thư viện để tìm sách, ... đều thông qua một chương trình máy tính để truy cập đến CSDL.

Những ví dụ đề cập ở trên được gọi là các ứng dụng CSDL truyền thống. Đó là các ứng dụng mà hầu hết dữ liệu được lưu trữ và truy xuất dưới dạng kí tự hoặc số. Sau này, sự thuận lợi của công nghệ đã hướng đến một số loại ứng dụng mới cho hệ thống CSDL. CSDL đa phương tiện (*Multimedia databases*) có thể lưu trữ hình ảnh, video và âm thanh. Hệ thống thông tin địa lý (*Geographic information system - GIS*) dùng để lưu trữ và phân tích bản đồ, dữ liệu thời tiết và các hình ảnh vệ tinh. Hệ thống Kho dữ liệu (*Data warehouse*) và phân tích trực tuyến (*Online Analytical Processing - OLAP*) được sử dụng trong nhiều công ty để rút trích và phân tích thông tin từ nhiều nguồn CSDL lớn nhằm đưa ra quyết định. Kỹ thuật CSDL “active” và thời gian thực (*Real time and active database technology*) được sử dụng trong lĩnh vực quản lý công nghiệp và xử lý nhà máy.

Các kỹ thuật nghiên cứu CSDL được áp dụng qua các ứng dụng Web hoặc ứng dụng Winform để người dùng khai thác, tìm kiếm thông tin cần thiết thông qua chương trình ứng dụng. Để hiểu được nền tảng của

các kỹ thuật CSDL, trước tiên chúng ta sẽ bắt đầu tìm hiểu nền tảng các ứng dụng CSDL truyền thống.

I.2 Định nghĩa dữ liệu, CSDL, hệ quản trị CSDL, hệ CSDL

I.2.1 Dữ liệu là gì?

Dữ liệu là toàn bộ những gì được máy tính lưu trữ và xử lý. Dữ liệu có thể là các ký tự, các con số, hình ảnh hay âm thanh. Dữ liệu được chia làm hai loại: dữ liệu tĩnh và dữ liệu động. Ví dụ: tên, địa chỉ, số điện thoại của khách hàng (dữ liệu tĩnh); báo cáo doanh thu, đăng ký học phần (dữ liệu động).

Thông tin là một khái niệm phản ánh tri thức, sự hiểu biết của con người về một đối tượng.

Phân biệt giữa dữ liệu và thông tin: ta có thể ví dụ dữ liệu như là nguyên vật liệu được dùng trong một quá trình sản xuất sản phẩm, còn thông tin như là sản phẩm. Vậy dữ liệu là một khái niệm rộng, thô, rời rạc và thông tin là sản phẩm của quá trình kết xuất từ dữ liệu. Tóm lại thông tin chính là nội dung của dữ liệu sau khi được xử lý.

I.2.2 CSDL là gì?

CSDL và kỹ thuật CSDL đang ảnh hưởng lớn đến sự phát triển của máy tính. CSDL đóng một vai trò rất quan trọng trong hầu hết các lĩnh vực có sử dụng máy tính như: kinh doanh, thương mại điện tử, công nghệ, y khoa, luật, giáo dục, ngân hàng, hành chính, giải trí, thư viện...

CSDL là tập hợp dữ liệu có mối liên hệ chặt chẽ với nhau. Dữ liệu ở đây được hiểu là những gì có thể ghi nhận lại và có ý nghĩa rõ ràng. Ví dụ, xem xét tên, số điện thoại và địa chỉ của một người, ta có thể ghi nhận lại dữ liệu này trong một cuốn sổ địa chỉ, hoặc trong đĩa cứng của máy tính cá nhân có phần mềm hỗ trợ như Microsoft Access hoặc Excel. Đây là một tập hợp dữ liệu có liên quan với nhau và có ý nghĩa rõ ràng, và được xem là một CSDL.

CSDL phải có ý nghĩa rõ ràng và có những đặc tính cụ thể như sau:

- CSDL phải biểu diễn một phần của thế giới thực (thế giới thu nhỏ). Thách thức đặt ra khi xây dựng CSDL là phải phản ánh được thế giới thu nhỏ này.
- CSDL là một tập hợp có cấu trúc của những thành phần dữ liệu có liên quan với nhau được lưu trữ trong máy tính. Một tập các dữ liệu ngẫu nhiên không được xem là CSDL.
- CSDL được thiết kế, xây dựng và lưu trữ để phục vụ cho mục đích cụ thể. Mục đích ở đây có thể là hướng tới hỗ trợ nghiệp vụ cho một nhóm người dùng (như nhân viên phòng tài chính kế toán) và là các ứng dụng được xác định trước (như ứng dụng kế toán).

Dữ liệu của một CSDL thu thập được phải có nguồn gốc, có liên quan với các sự kiện ở thế giới thực.

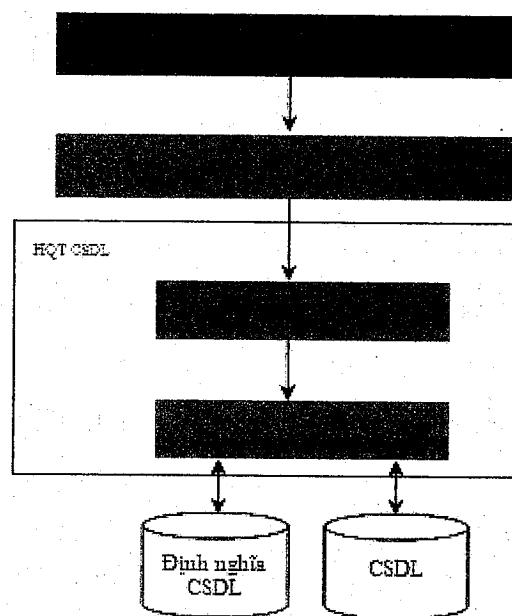
I.2.3 Định nghĩa hệ quản trị CSDL

Hệ quản trị CSDL (HQT CSDL) là các chương trình cho phép người dùng tạo ra và duy trì dữ liệu. Đó là phần mềm hệ thống cho phép *định nghĩa, xây dựng, xử lý và chia sẻ* CSDL giữa những người dùng khác nhau và ứng dụng. *Định nghĩa* CSDL gắn liền với việc chỉ định kiểu dữ liệu, cấu trúc và ràng buộc của dữ liệu được lưu trữ trong CSDL. *Xây dựng* CSDL là lưu trữ dữ liệu trên bộ nhớ phụ được điều khiển bởi một hệ quản trị CSDL. *Xử lý* CSDL bao gồm các thao tác như truy xuất dữ liệu theo yêu cầu, cập nhật dữ liệu để phản ánh sự thay đổi của thế giới thực, phát sinh báo cáo từ dữ liệu. *Chia sẻ* CSDL cho phép nhiều người dùng và nhiều ứng dụng truy xuất đến CSDL.

HQT CSDL cung cấp một giao diện giữa người sử dụng và dữ liệu. Một số HQT CSDL quan hệ đang phổ biến như: Microsoft Access, Microsoft SQL Server, Oracle, DB2,...

I.2.4 Định nghĩa hệ CSDL

Hệ CSDL là sự tích hợp giữa CSDL và HQT CSDL. Hình I-1 minh họa cho một hệ CSDL.



Hình I-1 Minh họa cho hệ CSDL.

I.3 Ví dụ cụ thể một CSDL

Chúng ta hãy xem qua một ví dụ quen thuộc: một CSDL lưu trữ thông tin liên quan đến sinh viên, môn học và kết quả học tập của sinh viên trong một trường đại học. CSDL được tổ chức trong năm bảng, mỗi bảng lưu trữ các bộ dữ liệu cùng loại. Bảng SINHVIÊN lưu trữ dữ liệu cho mỗi sinh viên, bảng MÔNHỌC lưu trữ dữ liệu cho các môn học, bảng HỌCPHẦN dùng để lưu trữ thông tin các môn học được mở ra trong từng học kỳ của từng năm học, bảng KẾTQUẢ lưu trữ thông tin điểm sinh viên đạt được cho mỗi học phần, bảng ĐIỀUKIÊN lưu trữ điều kiện học trước sau giữa các học phần.

MÔN HỌC	Tên MH	MÃ MH	SỐ TC	Khoa
	Khoa học máy tính	CS1310	4	CNTT
	Cấu trúc dữ liệu	CS3320	4	CNTT
	Toán rời rạc	MATH2410	3	TOÁN
	Cơ sở dữ liệu	CS3380	3	CNTT

Hình I-2 Minh họa cho bảng MÔN HỌC.

SINH VIÊN	Tên	MSSV	Lớp	Khoa
	Trang	17	1	CNTT
	Ngọc	8	2	CNTT

Hình I-3 Minh họa cho bảng SINH VIÊN.

HỌC PHẦN	MÃ HP	MÃ MH	HỌC KỲ	NĂM	GIÁO VIÊN
	85	MATH2410	1	2008	Anh
	92	CS1310	1	2007	Tiễn
	112	MATH2410	2	2008	Anh
	119	CS1310	2	2007	Tiễn

Hình I-4 Minh họa cho bảng HỌC PHẦN.

KẾT QUẢ	MSSV	MÃ HP	ĐIỂM
	17	112	10
	17	119	7
	8	85	6
	8	92	9

Hình I-5 Minh họa cho bảng KẾT QUẢ.

ĐIỀUKIỆN	MãMH	MãMH_Truy
	CS3380	CS3320
	CS3380	MATH2410
	CS3320	CS1310

Hình I-6 Minh họa cho bảng ĐIỀUKIỆN.

II. QUÁ TRÌNH PHÁT TRIỂN HỆ CSDL

II.1 Tiếp cận theo tập tin

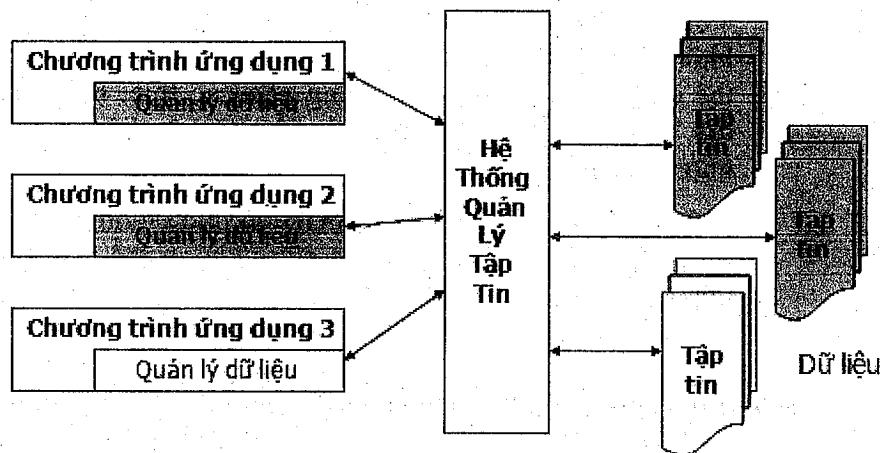
Trước đây, người ta thường sử dụng hệ thống tập tin để quản lý dữ liệu. Trong cách xử lý theo hướng tiếp cận tập tin, mỗi loại người dùng định nghĩa và xử lý các tập tin bằng một phần mềm ứng dụng cụ thể. Ví dụ, *loại người dùng thứ nhất* là các nhân viên quản lý kết quả học tập của sinh viên. Để quản lý kết quả học tập của sinh viên nhóm người đó có thể tạo tập tin để lưu trữ thông tin các sinh viên và điểm của chúng. Thông qua chương trình, những người dùng này có thể xuất ra thông tin của một sinh viên và nhập mới điểm của sinh viên. *Loại người dùng thứ hai* là các nhân viên quản lý học phí của sinh viên. Để quản lý việc đóng học phí của sinh viên, nhóm người dùng này tạo ra một tập tin lưu trữ thông tin sinh viên, học phí và tình trạng chi trả học phí của sinh viên. Mặc dù hai loại người dùng này đều quan tâm đến dữ liệu sinh viên nhưng mỗi người lưu trữ các tập tin và sử dụng các chương trình xử lý các tập tin này khác nhau. Và giả sử rằng một trường đại học có một lượng dữ liệu lớn (khoảng 500GB) về sinh viên, giáo viên, khoa, phòng ban, ... Dữ liệu này được nhiều giáo viên, sinh viên và những đối tượng người dùng khác truy xuất đồng thời. Những yêu cầu về dữ liệu phải được trả lời nhanh chóng. Việc thay đổi dữ liệu bởi các người dùng khác nhau phải tạo ra tình trạng nhất quán trên toàn CSDL. Và việc truy xuất đến dữ liệu cũng phải được quản lý theo quyền hạn cho phép. Ví dụ, chỉ cho phép một số loại người dùng phù hợp truy cập lương của giáo viên, điểm của sinh viên, ... Cách thức quản lý dữ liệu này bằng cách lưu trữ chúng trên hệ thống tập tin có thể gây ra một số hạn chế như sau:

- Với cấu hình của phần cứng đang sử dụng, chúng ta có thể không có 500GB bộ nhớ chính để lưu trữ toàn bộ dữ liệu. Vì thế chúng ta phải lưu trữ chúng vào các thiết bị lưu trữ như đĩa, băng từ và chỉ đưa những phần dữ liệu liên quan vào bộ nhớ chính để xử lý khi cần.
- Thậm chí, nếu chúng ta có 500GB bộ nhớ chính trên hệ thống máy tính 32 – bit thì chúng ta cũng không thể truy xuất trực tiếp hơn 4GB dữ liệu mà chúng ta phải lập trình theo một cách nào đó để xác định các thành phần của dữ liệu.
- Chúng ta phải viết chương trình để trả lời mỗi câu hỏi của người dùng khi họ muốn truy xuất đến dữ liệu. Chương trình này nếu viết thì cũng rất phức tạp bởi vì một số lượng lớn dữ liệu phải được tìm kiếm.
- Chúng ta phải đảm bảo dữ liệu được lưu trữ một cách nhất quán mỗi khi có sự thay đổi dữ liệu do nhiều người dùng khác nhau cùng sửa (như sửa đĩa chỉ của sinh viên do người của phòng tài vụ và người của phòng đào tạo cùng sửa trên tập tin của riêng họ).
- Cơ chế bảo mật của hệ điều hành không phù hợp cho các ứng dụng CSDL có các người dùng với quyền truy cập khác nhau trên dữ liệu.

Chính vì vậy với hướng tiếp cận này đã gây nhiều hạn chế:

- Dữ liệu dễ bị trùng lặp và dư thừa.
- Dữ liệu dễ bị thiếu nhất quán.
- Chia sẻ dữ liệu bị hạn chế.
- Truy xuất khó khăn (vì phải thông qua một phần mềm ứng dụng và phải biết cách tổ chức dữ liệu bên trong).
- Khó khôi phục dữ liệu khi có sự cố.

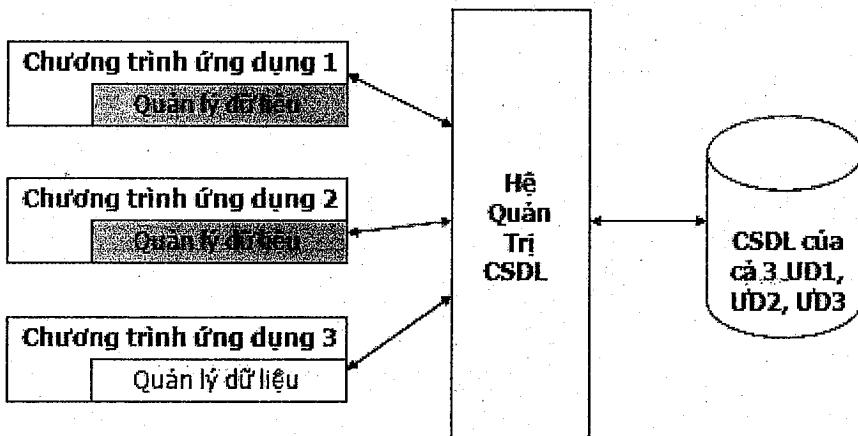
Hình I-7 minh họa việc quản lý dữ liệu bằng cách sử dụng hệ thống tập tin.



Hình I-7 Minh họa hướng tiếp cận theo hệ thống tập tin.

II.2 Tiếp cận theo CSDL

Việc quản lý dữ liệu của ứng dụng theo hướng tiếp cận sử dụng hệ thống tập tin có quá nhiều hạn chế. Chính vì vậy mà hướng tiếp cận theo CSDL đã ra đời và đã khắc phục được những hạn chế mà hướng tiếp cận theo tập tin mắc phải. Hình I-8 minh họa cho hướng tiếp cận theo hướng CSDL.



Hình I-8 Minh họa hướng tiếp cận theo CSDL.

III. MỘT SỐ ĐẶC TRƯNG CỦA HƯỚNG TIẾP CẬN CSDL

III.1 Tính tự mô tả

Một đặc tính cơ bản của hướng tiếp cận CSDL là hệ thống không chỉ chứa CSDL mà còn chứa dữ liệu mô tả cấu trúc của CSDL và các ràng buộc liên quan. Dữ liệu này được lưu trữ trong từ điển của HQT CSDL và được gọi là *siêu dữ liệu* (*meta-data*). Các chương trình ứng dụng có thể truy xuất đến CSDL nhờ cấu trúc được lưu trữ trong từ điển.

III.2 Tính độc lập

Trong xử lý tập tin truyền thông, cấu trúc của các tập tin dữ liệu được sử dụng bên trong các chương trình ứng dụng. Vì thế bất kỳ sự thay đổi nào của cấu trúc tập tin có thể dẫn tới sự thay đổi toàn bộ các chương trình truy cập đến tập tin này. Ngược lại, trong hầu hết trường hợp, ta không cần thay đổi chương trình ứng dụng khi cấu trúc CSDL bị thay đổi. Cấu trúc của CSDL được lưu trữ trong từ điển của HQT CSDL và tách biệt với các chương trình truy xuất. Tính chất này được gọi là tính độc lập giữa các chương trình và dữ liệu hay là sự độc lập chương trình – dữ liệu.

III.3 Tính trừu tượng

Hướng tiếp cận theo CSDL cho phép trình bày dữ liệu ở một mức trừu tượng cho phép, nhằm che bớt những chi tiết lưu trữ thật sự bên trong của CSDL. Ở đây, các loại mô hình dữ liệu được sử dụng để thể hiện các cách nhìn ở mức quan niệm của CSDL.

III.4 Tính nhất quán

Tính nhất quán trong hướng tiếp cận theo CSDL được thể hiện ở đặc tính:

- Lưu trữ dữ liệu thống nhất, tránh được thông tin trùng lặp.
- Cơ chế điều khiển truy xuất hợp lý, tránh được việc tranh chấp dữ liệu và đảm bảo dữ liệu luôn đúng tại mọi thời điểm.

III.5 Đa khung nhìn

Một CSDL có nhiều người dùng, mỗi người có khung nhìn (view) khác nhau về CSDL. Mỗi khung nhìn có thể là một phần của CSDL hoặc có thể là dữ liệu ảo – dữ liệu được kết xuất từ CSDL nhưng không được lưu trữ tường minh.

IV. NGƯỜI SỬ DỤNG CƠ SỞ DỮ LIỆU

IV.1 Quản trị viên

Trong bất kỳ tổ chức nào, nơi có nhiều người dùng cùng sử dụng cùng một nguồn tài nguyên thì cần phải có người quản trị viên để giám sát và quản lý các tài nguyên đó. Trong môi trường CSDL, nguồn tài nguyên thứ nhất là CSDL và nguồn tài nguyên thứ hai là HQT CSDL và phần mềm ứng dụng liên quan. Quản trị viên là người có trách nhiệm quản trị các nguồn tài nguyên này. Quản trị viên có trách nhiệm cấp quyền truy cập đến CSDL, điều phối và giám sát việc sử dụng chúng và yêu cầu các tài nguyên về phần mềm, phần cứng khi cần thiết.

IV.2 Thiết kế viên

Thiết kế viên có trách nhiệm xác định dữ liệu được lưu trữ trong CSDL và chọn cấu trúc phù hợp để thể hiện và lưu trữ dữ liệu này. Tất cả các nhiệm vụ này phải được thực hiện trước khi CSDL được lưu trữ và khai thác. Người thiết kế phải có trách nhiệm liên lạc với người dùng CSDL để nắm bắt yêu cầu của họ, để đưa ra một thiết kế thỏa các yêu cầu này. Trong một vài trường hợp, người thiết kế đóng vai trò người quản trị để gán quyền cho các nhân viên khác cho CSDL đã được thiết kế hoàn tất. Người thiết kế còn có trách nhiệm tương tác với nhóm người dùng và xây dựng các khung nhìn dữ liệu phù hợp theo các yêu cầu của họ. Rồi mỗi khung nhìn phải được phân tích và tích hợp với các khung nhìn của nhiều nhóm người dùng khác nhau. Cuối cùng, người thiết kế phải có khả năng hỗ trợ các yêu cầu cho tất cả các nhóm người dùng.

IV.3 Người dùng cuối

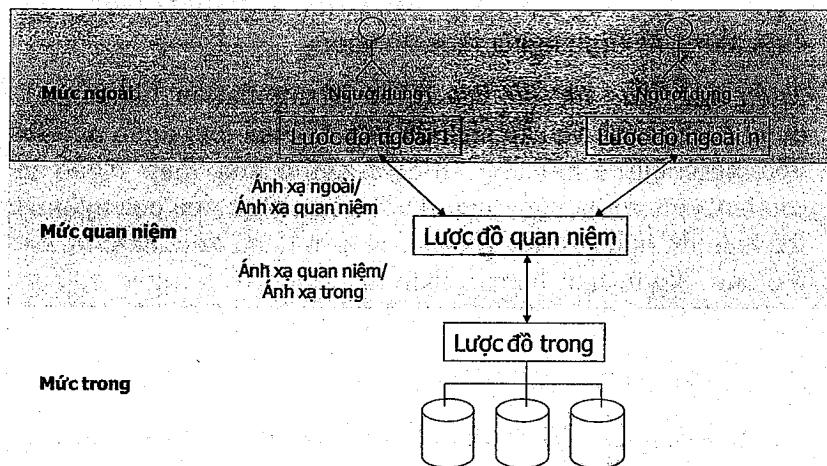
Người dùng cuối là những người mà công việc của họ yêu cầu truy cập đến CSDL để truy vấn, cập nhật và phát sinh các báo cáo. Người dùng cuối được phân thành các loại:

- Người ít sử dụng: là những người thỉnh thoảng mới truy cập đến CSDL, nhưng họ có thể cản thông tin khác nhau vào các thời điểm. Họ thường sử dụng các câu truy vấn phức tạp để mô tả các yêu cầu của họ.
- Người sử dụng thường xuyên: là người thường xuyên truy vấn và cập nhật CSDL nhờ vào một số chức năng đã được xây dựng sẵn.
- Người sử dụng đặc biệt: là nhóm người thông thạo về HQT CSDL, tự xây dựng các câu truy vấn phức tạp cho công việc.

V. KIẾN TRÚC CỦA HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

V.1 Kiến trúc ba lược đồ

Dữ liệu trong HQT CSDL được mô tả theo ba mức trừu tượng: lược đồ quan niệm, lược đồ vật lý, và lược đồ ngoài được minh họa ở Hình I-9.



Hình I-9 Minh họa kiến trúc ba lược đồ.

- Mức trong (lược đồ trong): mô tả cấu trúc lưu trữ vật lý dữ liệu.
- Mức quan niệm (lược đồ quan niệm): Mô tả cấu trúc của toàn thể CSDL cho một cộng đồng người sử dụng. Lược đồ ở mức này che bớt các chi tiết của cấu trúc lưu trữ vật lý.
- Mức ngoài (lược đồ ngoài): mức này còn gọi là mức khung nhìn. Ở mức này mô tả một phần của CSDL mà một nhóm người quan tâm đến và che giấu phần còn lại của CSDL đối với nhóm người dùng đó.

V.2 Các thành phần của HQT CSDL

V.2.1 Các ngôn ngữ của HQT CSDL

Ngôn ngữ định nghĩa dữ liệu (Data Definition Language - DDL) là ngôn ngữ được sử dụng bởi các quản trị viên và người thiết kế CSDL. Ngôn ngữ này dùng để xác định lược đồ quan niệm của một CSDL. Trong một số HQT CSDL, DDL cũng được dùng để xác định lược đồ trong và lược đồ ngoài (các khung nhìn).

Ngôn ngữ lưu trữ dữ liệu (Storage Definition Language – SDL) được dùng để định nghĩa lược đồ trong.

Ngôn ngữ định nghĩa khung nhìn (View Definition Language – VDL) được dùng để định nghĩa lược đồ ngoài.

Ngôn ngữ thao tác dữ liệu (Data Manipulation Language) được sử dụng để rút trích và cập nhật dữ liệu. Có hai loại ngôn ngữ thao tác dữ liệu: ngôn ngữ ở mức cao và ngôn ngữ ở mức thấp. Ngôn ngữ ở mức cao hay còn gọi là ngôn ngữ phi thủ tục, mục tiêu của ngôn ngữ này là thể hiện dữ liệu gì được rút trích ra hơn là dữ liệu được rút trích ra như thế nào. Ngôn ngữ ở mức thấp hay còn gọi là ngôn ngữ có thủ tục, ngôn ngữ này chỉ ra dữ liệu gì và được rút trích ra như thế nào.

V.2.2 Các giao tiếp của HQT CSDL

Các giao tiếp thân thiện của HQT CSDL đối với người dùng dựa trên các loại giao tiếp sau:

- Thực đơn: các giao tiếp này thể hiện cho người dùng một danh sách các xử lý.
- Hình thức: các giao tiếp này thể hiện các hình thức cụ thể cho người dùng.
- Đồ họa: hiển thị các lược đồ đến người dùng bằng hình thức biểu đồ. Với giao tiếp này người dùng chỉ việc kéo thả hoặc nhấp chuột, ...
- Ngôn ngữ tự nhiên: với giao tiếp này yêu cầu người dùng phải viết bằng tiếng Anh hoặc bằng một ngôn ngữ cụ thể được quy định.
- Kết hợp các loại giao tiếp trên.

VI. TÍNH NĂNG CỦA HQT CSDL

VI.1 Hạn chế sự trùng lặp

HQT CSDL kiểm soát được tính dư thừa của dữ liệu, nghĩa là tích hợp các nhu cầu dữ liệu của người dùng để xây dựng một CSDL thống nhất.

VI.2 Chia sẻ dữ liệu

Trong môi trường đa người dùng, các HQT CSDL cho phép truy xuất dữ liệu đồng thời.

VI.3 Hạn chế truy cập không được phép

HQT CSDL cho phép từng người dùng và nhóm người dùng có một tài khoản và một mật khẩu để truy xuất dữ liệu như vậy sẽ hạn chế những truy cập không được phép.

VI.4 Cung cấp nhiều giao diện giao tiếp

HQT CSDL cung cấp ngôn ngữ giao tiếp giữa CSDL và người dùng.

VI.5 Đảm bảo ràng buộc toàn vẹn

Ràng buộc toàn vẹn là những quy định cần được thỏa mãn để đảm bảo dữ liệu luôn phản ánh đúng ngữ nghĩa của thế giới thực. Một số ràng buộc toàn vẹn có thể khai báo với HQT CSDL và nhờ HQT CSDL kiểm tra. Một số ràng buộc toàn vẹn khác có thể kiểm tra nhờ chương trình ứng dụng.

VI.6 Khả năng sao lưu dự phòng

HQT CSDL có khả năng khôi phục dữ liệu khi có sự hư hỏng về phần cứng hoặc phần mềm nhờ vào khả năng sao lưu dự phòng.

VI.7 Một số tính năng khác

VI.7.1 Chuẩn hóa

Cho phép quản trị viên định nghĩa và bắt buộc áp dụng một chuẩn thống nhất cho mọi người.

VI.7.2 Uyển chuyển

Khi nhu cầu công việc thay đổi, cấu trúc CSDL rất có khả năng bị thay đổi. HQT CSDL cho phép thêm hoặc mở rộng cấu trúc dữ liệu mà không làm ảnh hưởng đến chương trình ứng dụng hiện hành.

VI.7.3 Khả dụng

Hệ quản trị cho phép khi có một thay đổi trên CSDL thì tất cả người dùng đều thấy được sự thay đổi đó.

VI.7.4 Giảm thời gian phát triển ứng dụng

HQT CSDL hỗ trợ nhiều tính năng cho người dùng khi khai báo, quản lý và khai thác dữ liệu. Vì thế sử dụng HQT CSDL để quản lý dữ liệu sẽ làm giảm bớt thời gian phát triển ứng dụng.

VII. MÔ HÌNH DỮ LIỆU

VII.1 Định nghĩa

Mô hình dữ liệu là một tập các khái niệm để mô tả cấu trúc của CSDL, cũng như các ràng buộc trên CSDL đó.

Các xử lý trên mô hình dữ liệu là các xử lý rút trích, cập nhật trên CSDL thông qua các khái niệm của mô hình dữ liệu. Các xử lý trên mô hình dữ liệu có thể là các xử lý cơ sở hoặc là các xử lý do người dùng định nghĩa.

VII.2 Giới thiệu các loại mô hình dữ liệu

Mô hình dữ liệu được phân chia thành ba loại mô hình chính sau:

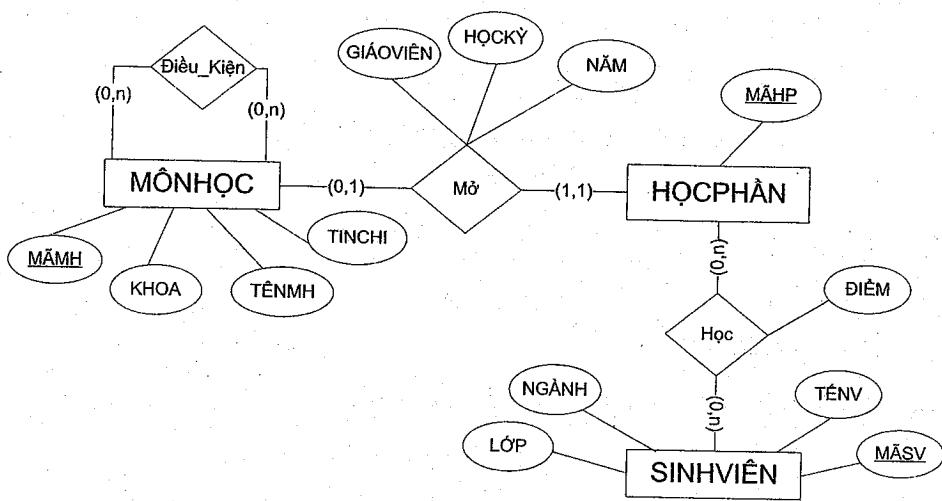
- Mô hình ở mức quan niệm (hay còn gọi là mức cao, mức ngữ nghĩa): cung cấp các khái niệm gần gũi với người dùng. Mô hình phải tự nhiên và giàu ngữ nghĩa. Ví dụ như mô hình thực thể kết hợp, mô hình đối tượng,...
- Mô hình ở mức cài đặt: đưa ra các khái niệm người dùng có thể hiểu được nhưng không khác biệt với dữ liệu được tổ chức thật sự trên máy tính. Ví dụ như mô hình dữ liệu quan hệ, mô hình mạng, mô hình phân cấp.
- Mô hình vật lý (hay còn gọi là mô hình ở mức thấp): đưa ra các khái niệm mô tả chi tiết về cách thức dữ liệu được lưu trữ trên đĩa cứng của máy tính.

VII.2.1 Mô hình thực thể kết hợp

Mô hình thực thể kết hợp (TT-KH) được giới thiệu bởi [CHEN, 1976] và ngày càng trở nên phổ biến. Đến năm 1988, ANSI chọn mô hình thực thể kết hợp là mô hình chuẩn cho hệ thống từ điển tài nguyên thông tin (IRDSS: *Information Resource Dictionary System*).

Ứng dụng CSDL được mô tả ở phần I.3 được mô tả theo mô hình TT-KH ở Hình I-10.

Chương 1 Tổng quan về CSDL



Hình I-10 Mô hình TT-KH.

VII.2.2 Mô hình quan hệ

Mô hình quan hệ được đề xuất bởi [E.F.Codd, 1970]. Các HQT CSDL quan hệ được đưa ra vào những năm 1980. Có thể kể một vài HQT CSDL tiêu biểu cài đặt theo lý thuyết mô hình dữ liệu quan hệ: Oracle, MS SQL, Sybase,...

Hình I-11 minh họa cho ứng dụng CSDL được mô tả ở phần I.3 bằng mô hình quan hệ.

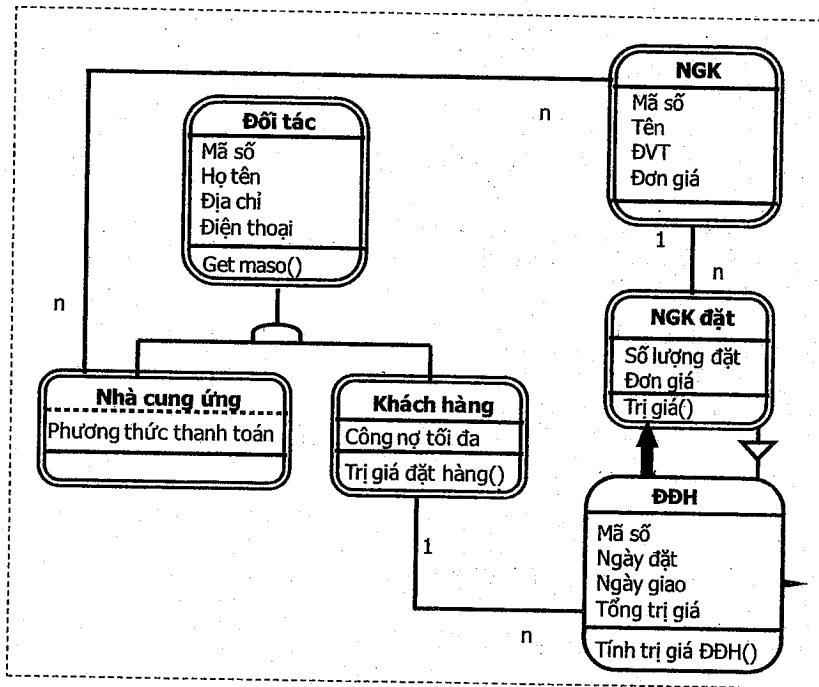
TÊN SV	MÃ SV	LỚP	NGÀNH	
MÃ NHM	TÊN NHM	MÃ NHM	KHOA	TINCHI
ĐIỂM VIEN	MÃMH TRƯỚC		MÃMH	
HỌC SINH	MÃHP	GIÀO VIÊN	HỌC KỲ	NĂM
KODE HỌC	MÃ SV	MÃHP	ĐIỂM	

Hình I-11 Minh họa cho mô hình quan hệ.

VII.2.3 Mô hình hướng đối tượng

Mô hình hướng đối tượng bắt đầu phát triển vào thập niên 90 dựa trên cơ sở sự phát triển của các phương pháp lập trình hướng đối tượng. Hình I-12 minh họa cho mô hình dữ liệu hướng đối tượng.

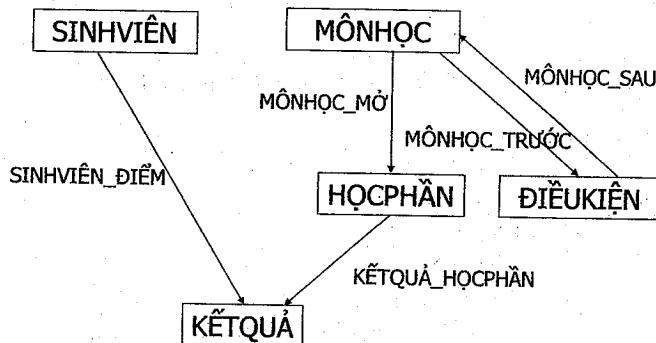
Chương 1 Tổng quan về CSDL



Hình I-12 Minh họa cho mô hình hướng đối tượng.

VII.2.4 Mô hình mạng

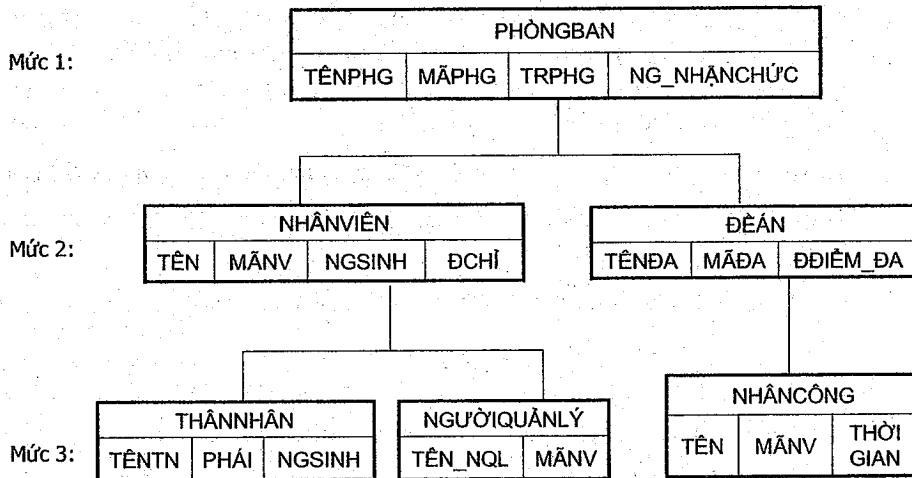
Mô hình mạng đầu tiên được thực hiện bởi Honeywell vào năm 1964-65. Mô hình mạng là một mô hình khá đơn giản vì chỉ dùng mối liên kết 1:1 và 1:n. Hình I-13 minh họa cho mô hình mạng.



Hình I-13 Minh họa cho mô hình mạng.

VII.2.5 Mô hình phân cấp

Mô hình phân cấp là mô hình được thực hiện bởi sự liên kết của công ty IBM và Rockwell vào khoảng năm 1965, đây là mô hình khá phổ biến và được sử dụng trong một số hệ thống như IMS. Hình I-14 minh họa cho mô hình phân cấp.



Hình I-14 Minh họa cho mô hình phân cấp.

VIII. MỘT ỨNG DỤNG CƠ SỞ DỮ LIỆU

Trong phần này chúng tôi trình bày một ứng dụng CSDL *Quản lý đề tài nghiên cứu khoa học*. Ứng dụng nhằm xây dựng một CSDL dùng để lưu trữ thông tin giáo viên, bộ môn, khoa cũng như các đề tài nghiên cứu khoa học mà giáo viên tham gia.

Mỗi giáo viên gồm có các thông tin về họ tên, địa chỉ, điện thoại, ngày sinh, lương, phái và thuộc về một bộ môn cụ thể. Mỗi giáo viên có thể có nhiều số điện thoại và địa chỉ gồm có các thông tin số nhà, đường, quận, thành phố.

Mỗi bộ môn gồm có các thông tin tên bộ môn, phòng bộ môn làm việc, điện thoại và do một giáo viên làm trưởng bộ môn, ngày nhận chức trưởng bộ môn của giáo viên đó. Mỗi khoa gồm có thông tin tên

khoa, năm thành lập, phòng làm việc, điện thoại và do một giáo viên làm trưởng khoa, ngày nhận chức trưởng khoa của giáo viên đó.

Mỗi đề tài gồm có các thông tin tên đề tài, cấp quản lý, kinh phí, ngày bắt đầu, ngày kết thúc, và thuộc về một chủ đề cụ thể. Mỗi chủ đề gồm có tên chủ đề. Mỗi đề tài có thể chia làm nhiều công việc. Mỗi công việc gồm có tên công việc, ngày bắt đầu, ngày kết thúc. Mỗi giáo viên có thể tham gia vào nhiều công việc cụ thể của các đề tài và mỗi công việc cũng có thể cho phép nhiều giáo viên tham gia. Khi giáo viên tham gia vào công việc thì có ghi nhận lại kết quả thực hiện công việc cũng như phụ cấp cho giáo viên.

Ví dụ này được chúng tôi sử dụng để minh họa cho hai loại loại mô hình là mô hình TT-KH và mô hình dữ liệu quan hệ ở các chương sau của giáo trình.

Câu hỏi ôn tập

1. Hãy định nghĩa các khái niệm: *dữ liệu, cơ sở dữ liệu, hệ quản trị cơ sở dữ liệu, hệ cơ sở dữ liệu*.
2. Hãy nêu một cách ngắn gọn ba hoạt động chính liên quan đến các cơ sở dữ liệu.
3. Hãy trình bày những thuận lợi trong việc quản lý dữ liệu do hướng tiếp cận cơ sở dữ liệu mang lại so với hướng tiếp cận tập tin.
4. Hãy nêu một số mô hình dữ liệu mà bạn biết?
5. Người sử dụng cơ sở dữ liệu gồm có những loại nào? Hãy cho biết vai trò của từng loại người dùng.
6. Người dùng cuối là gì? Trình bày một số công việc chính của họ.

Bài tập

1. Hãy tìm một vài ứng dụng thực tế có thể áp dụng cơ sở dữ liệu để quản lý dữ liệu. Với mỗi ứng dụng, hãy trình bày ngắn gọn mô tả về ứng dụng, và xác định sơ lược một vài bảng dữ liệu và thuộc tính cho các bảng dữ liệu đó (cách thức trình bày theo Hình I-11 sách giáo khoa).

Chương 2

MÔ HÌNH THỰC THẾ KẾT HỢP

I. MÔ HÌNH DỮ LIỆU CẤP CAO

Mô hình quan niệm dữ liệu là một loại mô hình dữ liệu cấp cao dùng để thể hiện ngữ nghĩa của dữ liệu. Trong chương này chúng tôi sẽ trình bày một loại mô hình quan niệm dữ liệu đó là mô hình TT-KH [CHEN,1976].

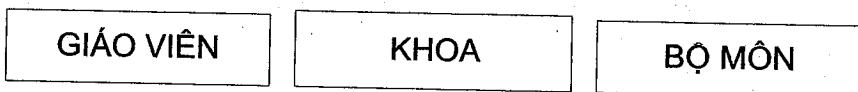
Mô hình TT-KH cơ bản chỉ có khái niệm tập thực thể, mối kết hợp, và thuộc tính. Về sau, người ta bổ sung thêm các khái niệm như thuộc tính kết hợp, thuộc tính suy diễn, thuộc tính đa trị, mối kết hợp tổng quát hóa chuyên biệt hóa, nhằm làm cho quá trình mô hình hóa trở nên dễ dàng hơn, hình thành mô hình TT-KH mở rộng. Trong chương hai, chúng tôi trình bày tất cả các khái niệm này.

Một mô hình TT-KH thường được biểu diễn bằng sơ đồ và được gọi là *sơ đồ TT-KH (ERD - Entity Relationship Diagram)*. Để biểu diễn các khái niệm trong sơ đồ này, hiện nay có nhiều hệ thống ký hiệu khác nhau nhưng chúng tôi sử dụng hệ thống ký hiệu cơ bản và có thể dễ dàng chuyển đổi chúng sang hệ thống ký hiệu khác.

II. THỰC THẾ VÀ TẬP THỰC THẾ

Thực thể (entity) là một đối tượng ngoài thế giới thực, có thể cụ thể hoặc trừu tượng. *Tập thực thể (entity set)* là một tập hợp các thực thể có tính chất giống nhau. Ký hiệu của tập thực thể là hình chữ nhật, bên trong có đặt tên của tập thực thể. Tên của tập thực thể là danh từ/cụm danh từ. Hình II-1 thể hiện ví dụ cho các tập thực thể GIÁO VIÊN, KHOA, BỘMÔN.

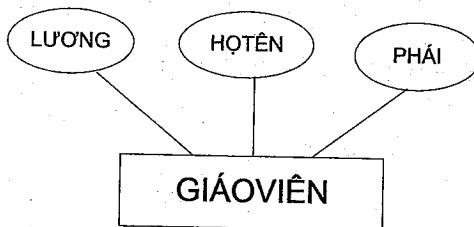
Chương 2 Mô hình thực thể kết hợp



Hình II-1 Minh họa cho tập thực thể.

III. THUỘC TÍNH

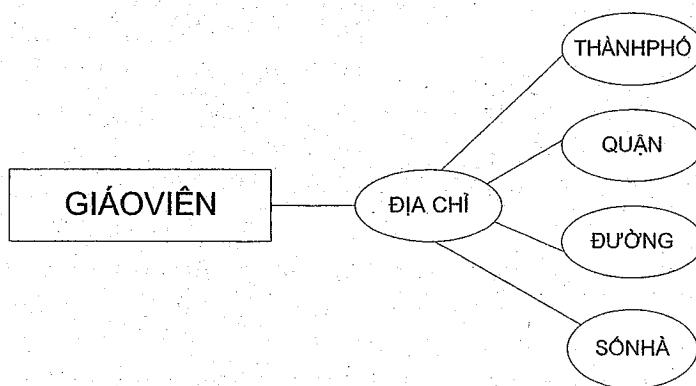
Thuộc tính là tập hợp các đặc trưng mô tả cho tập thực thể hay mối kết hợp. Ký hiệu của thuộc tính là một hình elip bên trong có đặt tên của thuộc tính, và tên của thuộc tính phải là danh từ/cụm danh từ. Các thuộc tính được gắn vào thực thể bằng một cung vô hướng. Hình II-2 biểu diễn cho thuộc tính họ tên (HOTEN), lương (LUONG), phái (PHAI) của tập thực thể GIÁO VIÊN.



Hình II-2 Minh họa cho tập thuộc tính của tập thực thể

Có nhiều loại thuộc tính: thuộc tính đơn, thuộc tính kết hợp, thuộc tính đơn trị, thuộc tính đa trị và thuộc tính suy diễn. Các thuộc tính minh họa ở Hình II-2 là thuộc tính đơn, chúng không thể hoặc không cần được chia làm nhiều thành phần nhỏ hơn.

Thuộc tính **kết hợp** là thuộc tính gồm nhiều thành phần nhỏ hơn. Ví dụ thuộc tính **ĐỊA CHỈ** của tập thực thể **GIÁO VIÊN** có thể chia nhỏ thành các thành phần: **SỐ NHÀ, ĐƯỜNG, QUẬN, THÀNH PHỐ** được minh họa ở **Hình II-3**.



Hình II-3 Minh họa cho thuộc tính kết hợp.

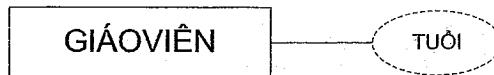
Thuộc tính *đơn trị* là thuộc tính chỉ nhận một giá trị đơn đối với một thực thể cụ thể. Ví dụ thuộc tính **HỌ TÊN** của tập thực thể **GIÁO VIÊN** là thuộc tính đơn trị, mỗi giáo viên chỉ có một họ tên, được minh họa ở Hình II-2.

Thuộc tính *đa trị* là thuộc tính nhận nhiều giá trị đối với một thực thể cụ thể. Ví dụ thuộc tính **ĐIỆN THOẠI** của tập thực thể **GIÁO VIÊN** là thuộc tính đa trị. Mỗi giáo viên có thể có nhiều số điện thoại được minh họa Hình II-4.



Hình II-4 Minh họa cho thuộc tính đa trị.

Thuộc tính *suy diễn* là thuộc tính mà giá trị của nó được tính toán từ giá trị của các thuộc tính khác. Ví dụ thuộc tính **TUỔI** của **GIÁO VIÊN** có thể tính toán từ thuộc tính **NGÀY SINH** của **GIÁO VIÊN** được minh họa ở Hình II-5.

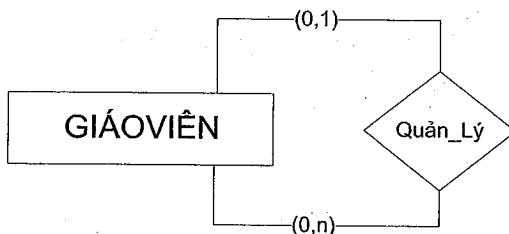


Hình II-5 Minh họa cho thuộc tính suy diễn.

IV. MỐI KẾT HỢP VÀ TẬP MỐI KẾT HỢP

Mối kết hợp là sự liên kết giữa hai hay nhiều thực thể. Tập hợp các mối kết hợp tương tự nhau được gọi là *tập mối kết hợp*. Tập mối kết hợp được ký hiệu bằng hình thoi và bên trong hình thoi có đặt tên cho tập mối kết hợp đó. Tên của tập mối kết hợp thường là động từ/cụm danh từ/liên từ.

Tập mối kết hợp phản thân (hay còn gọi là *tập mối kết hợp vòng*) là tập mối kết hợp giữa một tập thực thể với chính nó. Ví dụ như mối kết Hình II-6.

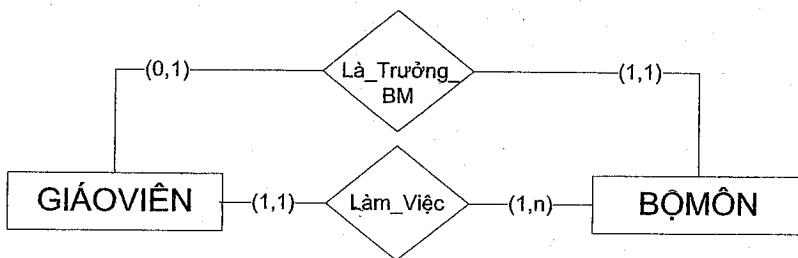


Hình II-6 Minh họa mối quan hệ phản thân.

Tập mối kết hợp được chia làm 4 loại chính: tập mối kết hợp một – một (1-1), tập mối kết hợp một – nhiều (1-n), tập mối kết hợp nhiều – một (n-1) và tập mối kết hợp nhiều – nhiều (n-n).

Hình II-7 minh họa giữa tập thực thể GIÁO VIÊN và BỘ MÔN với các tập mối kết hợp sau:

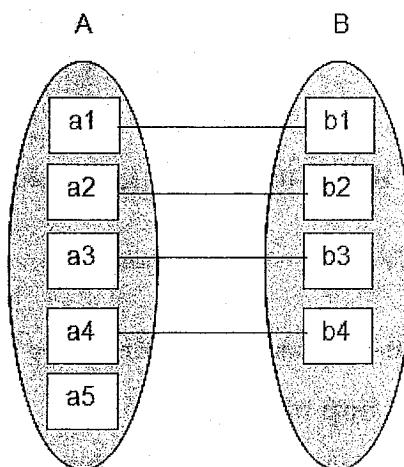
- Một giáo viên thuộc về một bộ môn nào đó.
- Một bộ môn do một giáo viên làm trưởng trưởng bộ môn.



Hình II-7 Minh họa cho tập mối kết hợp.

IV.1 Tập mối kết hợp một - một

Một thực thể trong tập thực thể A kết hợp với một thực thể trong tập thực thể B, và một thực thể trong B kết hợp với một thực thể trong tập thực thể A được minh họa ở Hình II-8. Cụ thể như một giáo viên làm trưởng cho một bộ môn và một bộ môn do một giáo viên làm trưởng được minh họa ở Hình II-9.



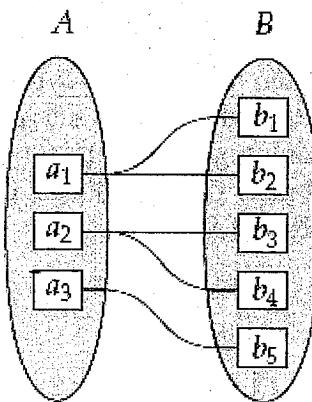
Hình II-8 Minh họa thẻ hiện tập mối kết hợp 1-1.



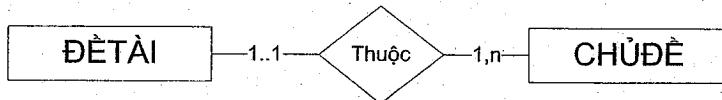
Hình II-9 Ví dụ thực tiễn cho tập mối kết hợp 1 – 1.

IV.2 Tập mối kết hợp một nhiều

Một thực thể trong tập thực thể A kết hợp với một số thực thể trong tập thực thể B (không hoặc nhiều). Tuy nhiên, một thực thể trong B chỉ kết hợp với một thực thể trong A được minh họa ở Hình II-10. Ví dụ cụ thể ở Hình II-11 cho thấy một chủ đề có thể có nhiều đề tài, nhưng mỗi đề tài chỉ thuộc một chủ đề.



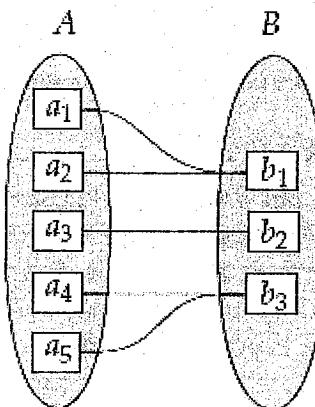
Hình II-10 Minh họa tập mối kết hợp 1-n.



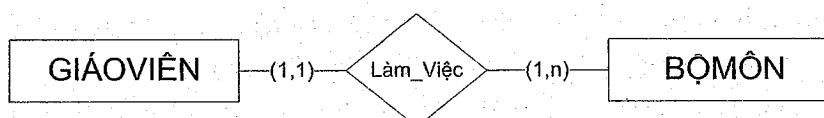
Hình II-11 Minh họa thực tiễn cho tập mối kết hợp 1-n.

IV.3 Tập mối kết hợp nhiều – một

Một thực thể trong tập thực thể A kết hợp với một thực thể trong tập thực thể B. Tuy nhiên, một thực thể trong B kết hợp với một số thực thể (không hoặc nhiều) của tập thực thể A được minh họa ở Hình II-12. Ví dụ ở Hình II-13 thể hiện một giáo viên chỉ thuộc về một bộ môn và mỗi bộ môn thì có thể có nhiều giáo viên.



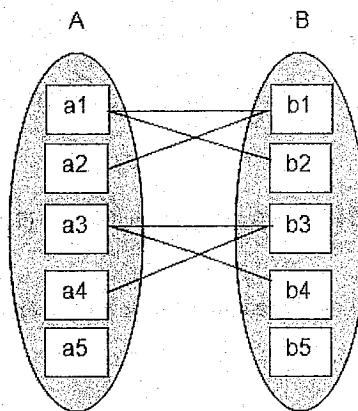
Hình II-12 Minh họa cho tập mối kết hợp n-1.



Hình II-13 Minh họa cho tập mối kết hợp n-1.

IV.4 Tập mối kết hợp nhiều – nhiều

Một thực thể trong tập thực thể A có thể kết hợp với một số thực thể (không hoặc nhiều) trong tập thực thể B và ngược lại một thực thể trong tập thực thể B có thể kết hợp với bất kỳ số lượng thực thể (một hoặc nhiều) trong tập thực thể B được minh họa ở Hình II-14. Ví dụ ở Hình II-15 minh họa cho tập mối kết hợp n-n, mỗi giáo viên có thể tham gia nhiều công việc của đè tài, mỗi công việc cũng có thể dành cho nhiều giáo viên tham gia.



Hình II-14 Minh họa tập mối kết hợp n-n.



Hình II-15 Minh họa cho tập mối kết hợp n-n.

IV.5 Bản số

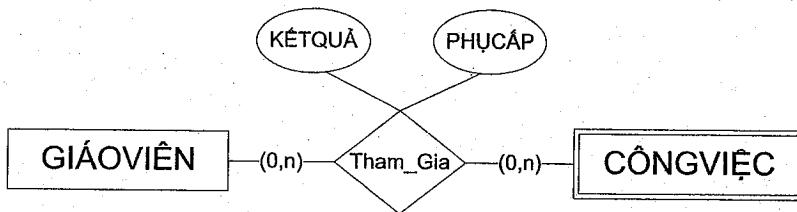
Bản số cho phép chúng ta ràng buộc giới hạn khả năng kết nối trên sự kết hợp giữa các tập thực thể. Các ràng buộc này được xác định từ các

tình huống ở thế giới thực và được thể hiện qua các tập mồi kết hợp. Ví dụ mỗi giáo viên quy định chỉ thuộc về một bộ môn.

Bản số được xác định bởi một cặp chỉ số (*min, max*) để quy định ràng buộc mỗi thực thể của tập thực thể E tham gia vào tập mồi kết hợp R, giá trị *min* thể hiện số lượng tối thiểu các thực thể thuộc tập thực thể E tham gia vào tập mồi kết hợp R và giá trị *max* thể hiện số lượng tối đa các thực thể của tập thực thể E tham gia vào tập mồi kết hợp R. Ví dụ như ở Hình II-16, tập thực thể giáo viên tham gia vào tập mồi kết hợp Tham_gia với bản số (0,n) (*min = 0, max = n*).

IV.6 Thuộc tính trên tập mồi kết hợp

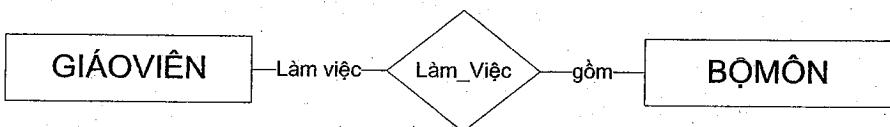
Tập mồi kết hợp cũng có thể có thuộc tính tương tự như các tập thực thể. Ví dụ để ghi nhận thời gian làm việc của một giáo viên với một công việc trong dự án, chúng tôi minh họa thuộc tính PHỤCẤP, KẾTQUẢ trên tập mồi kết hợp Tham_gia.



Hình II-16 Minh họa thuộc tính trên tập mồi kết hợp.

IV.7 Tên vai trò

Tên vai trò của tập mồi kết hợp dùng để biểu diễn ngữ nghĩa của một thực thể tham gia vào tập mồi kết hợp. Mỗi vai trò có một tên gọi. Thông thường tên vai trò được che giấu, không được hiển thị trên lược đồ TT-KH, thay vào đó tên tập mồi kết hợp sẽ lấy tên một vai trò hoặc một tên ghép các tên thực thể. Ví dụ cho tên vai trò được thể hiện ở Hình II-17.



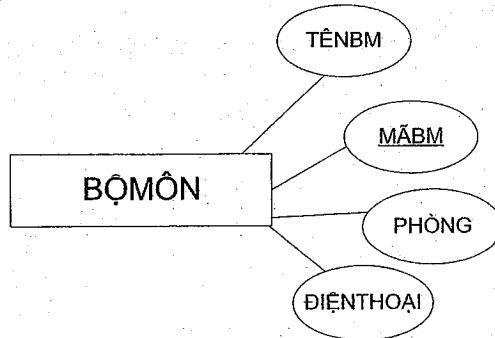
Hình II-17 Minh họa tên vai trò.

V. KHÓA CỦA TẬP THỰC THỂ

Một ràng buộc quan trọng giữa các thực thể trong một tập thực thể là không có hai thực thể nào có giá trị giống nhau hoàn toàn. Chúng ta phải có cách chỉ định rõ các thực thể được phân biệt với nhau trong một tập thực thể ở mức quan niệm. Cách chỉ định này thông thường được chỉ định bởi ràng buộc về khóa.

Khóa là một tập ít nhất các thuộc tính giúp ta xác định duy nhất một thực thể trong một tập thực thể. Khóa cũng giúp chúng ta xác định mỗi kết hợp là duy nhất trong một tập mối kết hợp.

Khóa của tập thực thể gồm một hoặc nhiều thuộc tính. Các thuộc tính tham gia tạo khóa được gọi là các thuộc tính khóa. Các thuộc tính khóa được ký hiệu bằng cách gạch chân dưới tên của thuộc tính. Ví dụ ở Hình II-18, Mã bộ môn (MÃBM) của tập thực thể BỘMÔN là thuộc tính khóa.



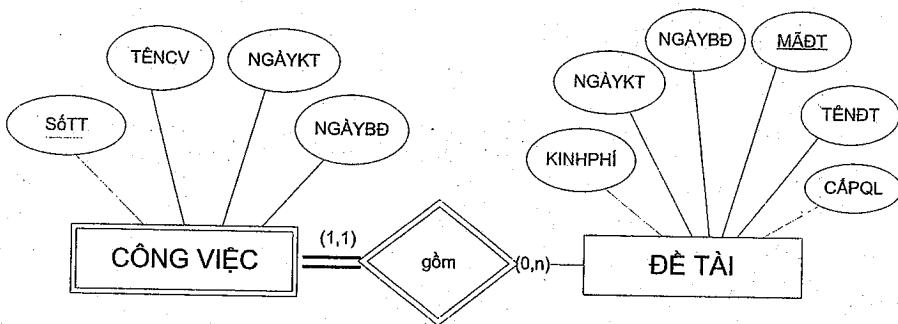
Hình II-18 Minh họa cho thuộc tính khóa.

VI. TẬP THỰC THỂ YẾU

Tập thực thể mà tập thuộc tính của nó không chứa khóa được gọi là tập thực thể yếu. (Tập thực thể mà tập thuộc tính của nó có chứa khóa như ta đã biết còn được gọi là tập thực thể mạnh). Tập thực thể yếu thường có mối kết hợp với một tập thực thể chủ (còn được gọi là tập thực thể xác định) thông qua tập kết hợp xác định. Tập thực thể yếu luôn tham gia toàn phần vào tập mối kết hợp xác định. Nhờ đó, ta có thể xác định một thực thể yếu dựa vào thực thể chủ mà nó có mối

Chương 2 Mô hình thực thể kết hợp

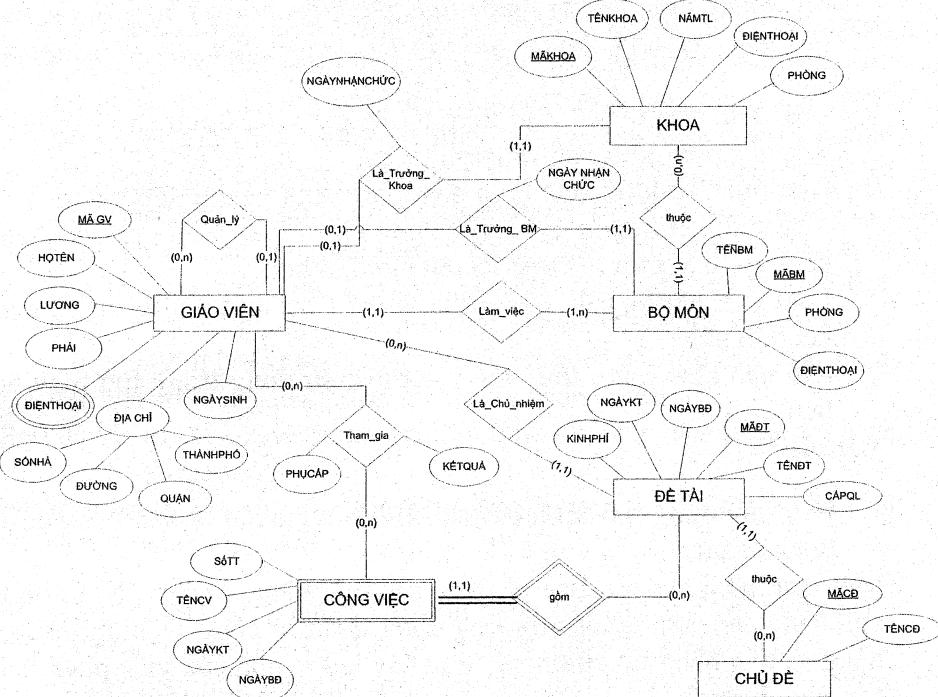
quan hệ. Tập thực thể yếu có khóa riêng phần (*partial key*), là tập hợp ít nhất các thuộc tính của tập thực thể yếu để xác định duy nhất một thực thể yếu trong các thực thể yếu cùng có mối kết hợp với một thực thể chủ. Tập thuộc tính tham gia làm khóa riêng phần của tập thực thể yếu được gạch chân đứt nét. Ví dụ CÔNGVIỆC là một thực thể yếu kết hợp với thực thể chủ ĐỀTÀI được minh họa ở Hình II-19. Trong đó, khóa riêng phần của CÔNGVIỆC là SốTT.



Hình II-19 Minh họa cho tập thực thể yếu.

VII. VÍ DỤ MÔ HÌNH TT-KH

Hình II-20 minh họa sử dụng mô hình TT-KH để biểu diễn dữ liệu được phân tích cho ứng dụng Quản lý đề tài nghiên cứu khoa học được trình bày ở chương I phần VIII.



Hình II-20 Mô hình TT-KH cho ứng dụng Quản lý đề tài nghiên cứu khoa học.

VIII. KẾT LUẬN

Như vậy trong chương này chúng tôi đã trình bày các khái niệm cơ bản của mô hình TT-KH. Đây là mô hình rất phổ biến được dùng để biểu diễn dữ liệu ở mức quan niệm. Trong chương 3 tiếp theo, sau khi trình bày lý thuyết về mô hình dữ liệu quan hệ chúng tôi sẽ trình bày sự liên hệ giữa mô hình TT-KH và mô hình dữ liệu quan hệ, cụ thể là cách thức chuyển đổi từ mô hình TT-KH sang mô hình dữ liệu quan hệ.

Câu hỏi ôn tập

1. Cho biết vai trò của mô hình dữ liệu cấp cao trong quá trình thiết kế cơ sở dữ liệu?
2. Hãy định nghĩa các khái niệm của mô hình thực thể kết hợp: *thực thể, thuộc tính, giá trị thuộc tính, thẻ hiện môi kết hợp, thuộc tính của tập môi kết hợp, thuộc tính đa trị, thuộc tính suy diễn, thuộc tính khóa*.
3. Loại thực thể là gì? Tập thực thể là gì? Giải thích sự khác biệt giữa thực thể, loại thực thể và tập thực thể.
4. Giải thích sự khác biệt giữa một thuộc tính và một tập giá trị.
5. Loại môi kết hợp là gì? Hãy giải thích sự khác biệt giữa thẻ hiện môi kết hợp, loại môi kết hợp và tập môi kết hợp.
6. Loại môi kết hợp đệ qui nghĩa là gì? Cho một vài ví dụ về loại môi kết hợp này.
7. Khi nào khái niệm thực thể yếu được sử dụng trong mô hình hóa dữ liệu.

Bài tập

1. Hãy xây dựng mô hình ER cho Hệ thống quản lý bán hàng được mô tả như sau:

Một cửa hàng chuyên bán sỉ và lẻ các mặt hàng đủ loại. Người quản lý cửa hàng cần xây dựng một ứng dụng quản lý công việc đặt hàng, giao hàng và bán hàng tại cửa hàng. Sau đây là kết quả của việc phân tích yêu cầu ứng dụng:

- Cửa hàng bán ra trên 300 mặt hàng với nguồn hàng lấy từ các nhà cung cấp. Một nhà cung cấp có một mã nhà cung cấp, tên, địa chỉ và điện thoại của nhà cung cấp. Mỗi nhà cung cấp có thể cung ứng nhiều mặt hàng khác nhau và mỗi mặt hàng cũng có thể được cung cấp bởi nhiều nhà cung cấp khác nhau, cần ghi nhận lại nhà cung cấp nào có thể cung ứng những mặt hàng gì.
- Cần lưu lại thông tin về tất cả các mặt hàng mà cửa hàng có mua bán: mã mặt hàng, tên hàng, hàng thuộc loại nào, đơn vị tính, quy cách, số lượng tồn. Mỗi loại hàng có mã loại hàng, tên loại hàng.
- Mỗi lần đặt hàng, cửa hàng sẽ điền một phiếu đặt hàng gửi đến nhà cung cấp. Cửa hàng phải điền các thông tin sau vào đơn đặt

hàng: số đơn đặt hàng, ngày đặt hàng, đặt tại nhà cung cấp nào, số lượng cần đặt đối với từng mặt hàng là bao nhiêu. Cuối mỗi đơn đặt hàng có thông tin về tổng số mặt hàng cần đặt. Trên phiếu đặt hàng chỉ có các mặt hàng mà nhà cung cấp có thể cung ứng.

- Mỗi khi đến giao hàng, nhà cung cấp giao cho cửa hàng một phiếu giao hàng, gồm các thông tin sau: số phiếu giao, ngày giao, giao cho đơn đặt hàng nào, mỗi mặt hàng số lượng giao và đơn giá là bao nhiêu. Ứng với 1 lần đặt hàng, nhà cung cấp có thể giao hàng tối đa là 3 lần và không được trễ hơn 7 ngày so với ngày đặt. Nhà cung cấp chỉ được giao các mặt hàng mà nhà cung cấp có đặt với số lượng giao không lớn hơn số lượng đặt.
 - Khi khách hàng đến mua hàng, cửa hàng sẽ lưu lại thông tin tất cả các hóa đơn bán hàng để tiện việc kiểm hàng. Thông tin hóa đơn gồm: số hóa đơn, ngày lập hóa đơn, tên khách hàng, địa chỉ khách hàng, điện thoại khách hàng, khách hàng đã mua những mặt hàng nào với số lượng mua, đơn giá mua là bao nhiêu.
2. Sau đây là mô tả về Hệ thống quản lý tồn kho. Hãy xây dựng mô hình ER cho hệ thống này.

Hệ thống quản lý tồn kho của các mặt hàng trong các kho hàng của một công ty được mô tả như sau:

- Kho hàng là nơi quản lý các mặt hàng của công ty. Mỗi kho hàng có một mã số duy nhất (MÃ_KHO) dùng để phân biệt các kho hàng khác, một tên kho hàng và một loại hàng mà kho đó chứa. Mỗi kho có một địa điểm nhất định được xác định bởi mã số địa điểm (MÃ_ĐĐ), địa chỉ của địa điểm, có một nhân viên phụ trách địa điểm và số điện thoại để liên lạc với kho tại địa điểm trên. Một kho chỉ chứa một loại hàng, một địa điểm có thể có nhiều kho.
- Một mặt hàng có một mã số duy nhất (MÃ_MH) để phân biệt các mặt hàng khác, có một tên hàng. Một mặt hàng được xếp vào một loại hàng, và một loại hàng có nhiều mặt hàng. Mỗi loại hàng có một mã số duy nhất để phân biệt (MÃ_LH), và có một tên loại hàng. Một mặt hàng có thể chứa ở nhiều kho, một kho có thể chứa nhiều mặt hàng cùng loại.

- Số lượng tồn kho của mỗi mặt hàng được xác định bởi phiếu nhập và phiếu xuất hàng. Mỗi phiếu nhập hàng có số phiếu nhập (SỐ_PN) duy nhất để phân biệt, và có ngày lập phiếu, phiếu nhập cho biết nhập tại kho nào, và có chữ ký của nhân viên phụ trách địa điểm của kho đó. Trong chi tiết của phiếu nhập cho biết số lượng nhập cho các mặt hàng của một phiếu nhập.
- Mỗi phiếu xuất hàng có số phiếu xuất (SỐ_PX) duy nhất để phân biệt, và có ngày lập phiếu, phiếu xuất cho biết xuất tại kho nào, chữ ký của nhân viên đi nhận hàng tại kho đó. Trong chi tiết của phiếu xuất cho biết số lượng xuất cho các mặt hàng của một phiếu xuất.

Thông tin của nhân viên phụ trách địa điểm tại các kho và nhân viên đi nhận hàng từ các kho bao gồm: Mã số nhân viên (MÃ_NV) để phân biệt giữa các nhân viên, có họ tên, phái, năm sinh, địa chỉ thường trú, số điện thoại của nhân viên.

3. Hãy xây dựng mô hình ER cho Hệ thống quản lý chuyên đề được mô tả như sau:

Phòng giáo vụ tại một trường đại học muốn tin học hóa việc quản lý đăng ký học các chuyên đề của sinh viên. Sau đây là kết quả của việc phân tích thiết kế ứng dụng trên:

- Mỗi sinh viên có một mã số duy nhất, có họ tên, thuộc một phái, có một ngày sinh, một địa chỉ và theo học một ngành duy nhất.
- Mỗi ngành có một mã ngành duy nhất, có một tên ngành duy nhất và một con số cho biết tổng số sinh viên đã từng theo học ngành này. Đôi với từng ngành, số lượng chuyên đề mà 1 sinh viên phải hoàn tất đã được trường quy định trước, và con số này không được vượt quá 5. Ngoài ra, trường cũng quy định trước danh sách các chuyên đề đối với từng ngành cụ thể để một sinh viên thuộc một ngành biết được mình phải học những chuyên đề nào.
- Mỗi chuyên đề có một mã duy nhất và có một tên duy nhất. Cần lưu lại thông tin về số sinh viên tối đa có thể chấp nhận được mỗi khi có 1 lớp mở cho chuyên đề cụ thể.
- Vào đầu mỗi học kỳ của mỗi năm học, phòng giáo vụ lên danh sách các chuyên đề được mở để sinh viên có thể đăng ký học. Sinh viên chỉ được đăng ký học những chuyên đề có mở.

- Khi sinh viên đăng ký học, cần ghi nhận lại việc đăng ký học một chuyên đề của một sinh viên vào một năm của một học kỳ nào đó. Mỗi năm có 2 học kỳ. Sinh viên chỉ được đăng ký tối đa là 3 chuyên đề trong một học kỳ mà thôi.
4. Hãy xây dựng mô hình ER cho Hệ thống quản lý cửa hàng nước giải khát được mô tả như sau:

Cửa hàng bán lẻ nước giải khát đủ loại (nước suối, rượu, nước ngọt, bia,...). Các loại nước giải khát này thuộc nhiều hiệu khác nhau (ví dụ: nước cam hiệu Tribeco và Rừng Hương). Mỗi loại nước trong mỗi hiệu có một giá bán lẻ khác nhau. Cửa hàng có một số khách quen mua nước đều đặn ở cửa hàng – đối với số khách này, cửa hàng ghi nhận tên, địa chỉ và số điện thoại. Mỗi lần khách đến mua nước, sau khi kiểm tra các mặt hàng và số lượng cần mua, cửa hàng lập một hóa đơn trong đó có hóa đơn trong đó có ghi các thông tin về khách hàng và chi tiết các loại nước trong mỗi hiệu cùng số lượng (đơn vị tính là chai) và số tiền tương ứng. Ở cuối hóa đơn ghi tổng số tiền phải trả. Khách sẽ thanh toán và nhận hàng ở bộ phận giao hàng. Riêng đối với khách quen, có trong hồ sơ của khách hàng, thì cửa hàng chấp nhận cho lấy hàng trước (tại cửa hàng) và thanh toán hóa đơn trong vòng 3 ngày.

Cuối mỗi ngày, cửa hàng kiểm tra lượng hàng còn trong mỗi loại nước của mỗi hiệu. Nếu lượng tồn ở dưới mức tối thiểu thì cửa hàng sẽ đặt mua thêm ngày hôm sau. Lượng tồn tối thiểu này được xác định dựa trên kinh nghiệm kinh doanh của cửa hàng. Mỗi loại nước trong mỗi hiệu được cung cấp tại một nơi duy nhất gọi là đơn vị cung ứng. Đơn vị cung ứng này có thể là xí nghiệp sản xuất hay công ty cung ứng nước giải khát. Mỗi lần đặt hàng thì cửa hàng sẽ điền vào một phiếu đặt hàng trong đó có ghi ngày đặt, số lượng cho từng loại. Đơn vị cung ứng sẽ áp dụng cho những đơn giá khác nhau cho mỗi lần đặt hàng. Đơn vị cung ứng có thể giao hàng làm nhiều lần, tối đa là 3 lần trong vòng một tuần. Mỗi lần giao hàng sẽ có một phiếu giao hàng kiêm hóa đơn trong đó có chi tiết các loại nước giải khát, nhắc lại tổng lượng đặt, lượng đã giao, lượng giao đợt này, đơn giá, số tiền tương ứng cho loại đó và số tiền tổng cộng phải trả. Cửa hàng phải thanh toán ngay khi nhận hàng.

Chương 2 Mô hình thực thể kết hợp

- Sau đây là mô tả về việc quản lý bán hàng tại một siêu thị. Hãy xây dựng mô hình ER cho hệ thống này.

Siêu thị hoạt động kinh doanh bán tất cả các mặt hàng tiêu dùng, ăn uống, thời trang... Sau đây là mô tả hoạt động của siêu thị:

Bán hàng: Khách đến mua hàng tại siêu thị có thể tự do vào siêu thị chọn hàng, hoặc nếu có yêu cầu về hàng hóa thì có thể báo cho nhân viên bán hàng tìm hộ hoặc vào kho lấy thêm. Sau đó, khách hàng sẽ đến quầy tính tiền để thanh toán. Nếu những khách hàng nào có thẻ VIP (very important person) thì hóa đơn thanh toán sẽ được giảm giá theo tỉ lệ phần trăm ghi trên thẻ. Mỗi thẻ VIP sẽ có giá trị trong một số lần thanh toán nhất định (số lần được giảm giá tối đa được ghi rất rõ trên thẻ).

Hậu mãi: Sau khi khách hàng mua hàng trong vòng 30 ngày, nếu khách hàng không vừa ý với mặt hàng mình mua thì có thể đem hàng để đổi hoặc trả lại. Và khách hàng chỉ có thể trả hoặc đổi hàng nếu khách hàng đáp ứng đầy đủ 3 yêu cầu sau:

- Hàng đổi và trả phải có chất lượng giống như lúc mua.
- Khách hàng phải có hóa đơn mua hàng của những mặt hàng muốn đổi hoặc trả lại.
- Khách hàng đã sử dụng thẻ VIP khi mua những mặt hàng muốn đổi hoặc trả lại.

Nếu hàng trả, nhân viên tính tiền sẽ kiểm tra thời gian hợp lệ, và lập phiếu chi cho khách hàng. Trên phiếu chi sẽ ghi rất rõ về ngày, số phiếu chi, lý do, họ tên khách, số tiền, lý do chi và phiếu chi này là của hóa đơn mua hàng nào.

Nếu đổi hàng thì nhân viên tính tiền sau khi kiểm tra sẽ lập phiếu đổi hàng trên đó gồm số phiếu đổi, ngày lập, mã số hàng trả, số tiền trả, mã số hàng nhận, số tiền hàng nhận, số tiền chênh lệch. Nếu số tiền hàng trả lớn hơn số tiền hàng nhận thì khách sẽ nhận lại số tiền chênh lệch. Nếu không khách hàng sẽ phải trả thêm số tiền chênh lệch cho siêu thị.

Tồn kho: cuối mỗi ngày, nhân viên thống kê sẽ xem xét tồn kho cuối ngày. Nếu số lượng tồn kho nhỏ hơn số lượng tồn kho tối thiểu thì sẽ lập phiếu xin nhập hàng.

Quản lý thẻ VIP: cứ mỗi kỳ siêu thị sẽ tặng và bán một lượng thẻ VIP. Khách hàng sẽ dùng thẻ này để giảm giá khi mua hàng và có thể sử dụng cho dịch vụ hậu mãi. Khi khách hàng mua hàng, siêu thị sẽ cập nhật lại số lần còn sử dụng được của thẻ.

Chương 3

MÔ HÌNH DỮ LIỆU QUAN HỆ

I. GIỚI THIỆU

Mô hình dữ liệu quan hệ được Edgar Frank Codd đề xuất vào năm 1970. Do tính chất đơn giản và được xây dựng trên nền tảng toán học vững chắc nên mô hình quan hệ được sử dụng rộng rãi từ thập niên 80 cho đến ngày nay.

Hiện tại, có nhiều hệ thống thương mại sử dụng mô hình quan hệ: DB2 của IBM, Oracle của Oracle, SQL Server và Access của Microsoft.

II. MÔ HÌNH DỮ LIỆU QUAN HỆ

II.1 Định nghĩa quan hệ

Cho *lược đồ quan hệ (relation scheme)* $R(A_1, A_2, \dots, A_n)$, trong đó R là tên của lược đồ quan hệ và A_1, A_2, \dots, A_n là các *thuộc tính* của lược đồ quan hệ.

Một *quan hệ (relation)* r của lược đồ quan hệ $R(A_1, A_2, \dots, A_n)$, ký hiệu là $r(R)$, là một tập hợp các *bộ* t_i , $r = \{t_1, t_2, \dots, t_m\}$. Mỗi bộ t là một danh sách có thứ tự n giá trị $t = \langle v_1, v_2, \dots, v_n \rangle$ trong đó, từng giá trị v_j ($1 \leq j \leq n$) thuộc tập hợp các phần tử mà thuộc tính A_j có thể nhận lấy hoặc bằng rỗng (*null*). Một quan hệ thực ra là một bảng dữ liệu hai chiều được đặt tên, có một số cột và một số dòng dữ liệu.

Ví dụ: Hình III-1 là quan hệ KHOA, được mô tả bởi lược đồ quan hệ KHOA(MÃKHOA, TÊNKHOA, NAMTL, PHÒNG, ĐIỆN THOẠI, TRƯỞNGKHOA, NGÀYNHÂNCHỨC). Quan hệ KHOA có bảy thuộc tính là MÃKHOA, TÊNKHOA, NAMTL, PHÒNG, ĐIỆN

THOẠI, TRƯỜNGKHOA, NGÀYNHẠNCÚC và bốn bộ dữ liệu. *<CNTT, Công nghệ thông tin, 1995, B11, 838123456, 2, 20/02/2005>* là một bộ dữ liệu của quan hệ KHOA, gồm các giá trị liên quan đến khoa ‘Công nghệ thông tin’.

KHOA

MÃKHOA	TÊNKHOA	NÃMTL	PHÒNG	ĐIỆNTHOAI	TRƯỜNGKHOA	NGÀYNHẠNCÚC
CNTT	Công nghệ thông tin	1995	B11	838123456	2	20/02/2005
VL	Vật lý	1976	B21	838223223	5	18/09/2003
SH	Sinh học	1980	B31	838454545	4	11/10/2000
HH	Hóa học	1980	B41	838456456	7	15/10/2001

Hình III-1 Quan hệ KHOA.

Ta dùng khái niệm ‘lược đồ quan hệ’ để đề cập đến cấu trúc của một quan hệ trong khi khái niệm ‘quan hệ’ để cập đến thành phần dữ liệu của quan hệ đó.

Mỗi dòng trong quan hệ (trừ dòng tiêu đề) được gọi là một *bộ* (*tuple*). Mỗi bộ là một tập hợp các giá trị mô tả về một thực thể hoặc một mối kết hợp giữa các thực thể trong thế giới thực.

Mỗi tiêu đề cột của một quan hệ được gọi là một *thuộc tính* (*attribute*). Mỗi thuộc tính cho biết ý nghĩa của từng giá trị dữ liệu trong một bộ. Đó là một đặc điểm của một tập thực thể hoặc một tập mối kết hợp. Tên quan hệ và tên các thuộc tính giúp ta hiểu rõ hơn về ngữ nghĩa của các giá trị dữ liệu trong từng dòng. Thứ tự các thuộc tính trong một quan hệ là không quan trọng. Tất cả các giá trị tại một thuộc tính có cùng kiểu dữ liệu.

Thuộc tính của một quan hệ không thể là thuộc tính kết hợp hay đa trị.

Mô hình dữ liệu quan hệ thể hiện CSDL bằng một tập hợp các quan hệ.

Tập hợp các giá trị mà thuộc tính A_i có thể nhận lấy được gọi là *miền giá trị* (*domain*) của thuộc tính A_i , ký hiệu là $\text{dom}(A_i)$. Một miền giá trị D là một tập hợp các giá trị nguyên tố, nghĩa là ta không cần đề cập đến thành phần chia nhỏ hơn nữa của giá trị đó. Người ta thường dùng kiểu dữ liệu (ví dụ ký tự, chuỗi ký tự, số nguyên, số thực, ...), hoặc

định dạng (ví dụ có 5 ký số, hoặc có 3 ký tự đầu tiên và 5 ký số sau đó, ...) để thể hiện miền giá trị.

Ví dụ: Miền giá trị của MÃKHOA, TÊNKHOA là tập hợp các chuỗi ký tự. Thuộc tính PHÁI trong quan hệ GIÁOVIÊN (MÃGV, HỌTÊN, LUÔNG, PHÁI, MÃBM) chỉ có thể nhận lấy các giá trị trong miền giá trị là một tập hợp có 2 phần tử là: {'Nam', 'Nữ'}.

Bậc của một quan hệ là số thuộc tính n của lược đồ quan hệ mô tả quan hệ đó.

Một quan hệ r còn được gọi là một *thể hiện quan hệ* (*relation instance*).

Khi muốn đề cập đến giá trị thứ i của bộ t, tức là giá trị tương ứng với thuộc tính A_i , thì ta dùng ký hiệu $t[A_i]$ hoặc $t.A_i$.

II.2 Một số tính chất của quan hệ

Các bộ trong một quan hệ là duy nhất

Trong toán học, ta biết rằng các phần tử trong một tập hợp là không trùng nhau. Vì một quan hệ là một tập hợp các bộ nên không tồn tại hai bộ dữ liệu trùng nhau.

Về thứ tự các bộ trong quan hệ

Cũng dựa trên lý thuyết tập hợp, không tồn tại một thứ tự giữa các phần tử của một tập hợp. Định nghĩa quan hệ không quan tâm đến thứ tự các bộ trong quan hệ.

Thứ tự các giá trị trong một bộ

Khi đã liệt kê các thuộc tính của quan hệ theo một thứ tự nào đó thì các giá trị trong một bộ dữ liệu phụ thuộc vào thứ tự của các thuộc tính.

Giá trị của một bộ dữ liệu

Mỗi bộ bậc n là một danh sách có thứ tự gồm có n giá trị nguyên tố. Một thuộc tính có thể mang giá trị rỗng, hàm ý là chưa biết hoặc không tồn tại giá trị đó tại một bộ cụ thể.

III. Các khái niệm về khóa

Cho lược đồ quan hệ R (A_1, A_2, \dots, A_n) và $r(R)$. Ta sẽ lần lượt định nghĩa các khái niệm liên quan đến khóa trên quan hệ r này.

Siêu khóa (super key)

Ta biết rằng, trong một quan hệ không tồn tại hai bộ dữ liệu giống nhau hoàn toàn trên tất cả các thuộc tính. Gọi $S \subseteq \{A_1, A_2, \dots, A_n\}$. S là siêu khóa của R nếu S thỏa điều kiện sau đây:

$$\forall t_1, t_2 \in r(R), t_1 \neq t_2 \text{ thì } t_1[S] \neq t_2[S]$$

Điều kiện này có nghĩa là không tồn tại hai bộ bất kỳ có giá trị giống nhau hoàn toàn trên tập S .

Một quan hệ có thể có nhiều siêu khóa và có một siêu khóa mặc định là tập hợp tất cả các thuộc tính của quan hệ đó.

Khóa (key)

Gọi $K \subseteq \{A_1, A_2, \dots, A_n\}$. K là khóa của R nếu K thỏa 2 điều kiện sau đây:

- (1). $\forall t_1, t_2 \in r(R), t_1 \neq t_2 \text{ thì } t_1[K] \neq t_2[K]$.
- (2). $\neg \exists K' \subset K, K' \text{ thỏa (1)}$.

Khóa K của lược đồ quan hệ R là một siêu khóa của R , nhưng K phải thỏa mãn thêm điều kiện là nếu bỏ bớt đi bất kỳ một thuộc tính A nào của K thì tập còn lại là K' sẽ không còn là siêu khóa của R nữa.

Thuộc tính $A \in K$ được gọi là *thuộc tính khóa (key attribute)*.

Ví dụ: $\{\text{MÃKHOA}\}$ là khóa của quan hệ KHOA, $\{\text{MÃKHOA}, \text{TÊNKHOA}\}$ là siêu khóa. $\{\text{MÃKHOA}, \text{TÊNKHOA}\}$ không phải là khóa của KHOA vì khi bỏ đi thuộc tính TÊNKHOA từ tập này thì tập còn lại là $\{\text{MÃKHOA}\}$ vẫn còn thỏa điều kiện là một siêu khóa.

Khóa ứng viên (candidate key)

Một lược đồ quan hệ có thể có nhiều khóa. Mỗi khóa được gọi là khóa ứng viên.

Ví dụ: Quan hệ BỘMÔN (MÃBM, TÊNBM, TRƯỞNGBM, MÃKHOA) có hai khóa ứng viên là MÃBM và TÊNBM, vì không có hai bộ môn nào có cùng mã hoặc có cùng tên.

Khóa chính (primary key)

Trong các khóa ứng viên, có một khóa ứng viên được chọn làm khóa chính cho quan hệ và được gạch dưới.

Ví dụ: MÃBM được chọn là khóa chính cho quan hệ BỘMÔN. Khi đó, lược đồ quan hệ BỘMÔN được thể hiện như sau:

BỘMÔN (MÃBM, TÊNBM, TRƯỞNGBM, MÃKHOA)

Khóa ngoại (foreign key)

Khóa ngoại cho biết mối quan hệ giữa các bộ dữ liệu trong hai quan hệ.

Cho hai lược đồ quan hệ R_1 , R_2 và hai quan hệ tương ứng $r_1(R_1)$, $r_2(R_2)$. Gọi FK là một tập các thuộc tính của R_1 và PK là khóa chính của R_2 . Ta nói FK là khóa ngoại của R_1 tham chiếu đến R_2 nếu hai điều kiện sau đây thỏa mãn:

1. Các thuộc tính của FK tương ứng cùng miền giá trị với khóa chính PK.
2. Với mọi bộ t_1 của quan hệ r_1 , thì giá trị tại FK bằng rỗng hoặc luôn tìm thấy một bộ t_2 trong quan hệ r_2 thỏa điều kiện $t_1[FK] = t_2[PK]$. Trong trường hợp FK gồm các thuộc tính khóa thì không được nhận giá trị rỗng.

Ta nói rằng bộ t_1 tham chiếu đến bộ t_2 .

Ví dụ: Thuộc tính TRƯỞNGKHOA trong quan hệ:

KHOA (MÃKHOA, TÊNKHOA, NAMTL, TRƯỞNGKHOA)

là khóa ngoại tham chiếu đến thuộc tính MÃGV của quan hệ:

GIÁOVIÊN (MÃGV, HỌTÊN, LUONG, PHÁI, MÃBM)

Điều này có nghĩa là giá trị tại thuộc tính TRƯỞNGKHOA của bất cứ bộ dữ liệu nào trong quan hệ KHOA hoặc là bằng *null* (nếu tại một thời điểm đang xét ta chưa biết giáo viên nào làm trưởng khoa đó)

hoặc là bằng một giá trị tại một bộ nào đó trên thuộc tính là khóa chính của lược đồ quan hệ GIÁOVIÊN – thuộc tính MÃGV.

Lược đồ CSDL (database schema) S là một tập hợp các lược đồ quan hệ $S = \{R_1, R_2, \dots, R_m\}$.

Thể hiện CSDL (database instance) DB của lược đồ CSDL S là một tập hợp các thể hiện quan hệ $DB = \{r_1, r_2, \dots, r_m\}$ trong đó r_i là thể hiện của R_i .

IV. CHUYỂN ĐỔI MÔ HÌNH TT-KH SANG MÔ HÌNH QUAN HỆ

IV.1 Tập thực thể sang quan hệ

Đối với từng tập thực thể mạnh E trong mô hình TT-KH, tạo một quan hệ R tương ứng, gồm tất cả các thuộc tính đơn của E. Chọn một trong các khóa của E làm khóa chính cho R. Nếu khóa được chọn liên quan đến thuộc tính kết hợp thì các thuộc tính đơn liên quan đến thuộc tính kết hợp đó là khóa của R.

Trong ví dụ, có các quan hệ được tạo như sau:

1. GIÁOVIÊN (MÃGV, HỌTÊN, LUONG, PHÁI, NGÀYSINH, SÔNHÀ, ĐƯỜNG, QUẬN, THÀNHPHỐ)
2. BỘMÔN (MÃBM, TÊNBM, PHÒNG, ĐIỆNTHOẠI, NGÀYNHẬNCHỨC)
3. KHOA (MÃKHOA, TÊNKHOA, NĂMTL, PHÒNG, ĐIỆNTHOAI, NGÀYNHẬNCHỨC)
4. ĐỀTÀI (MÃĐT, TÊNĐT, KINHPHÍ, CÁPQL, NGÀYBD, NGÀYKT)
5. CHỦĐỀ (MÃCĐ, TÊNCĐ)

IV.2 Tập mối kết hợp 2 ngôi sang quan hệ

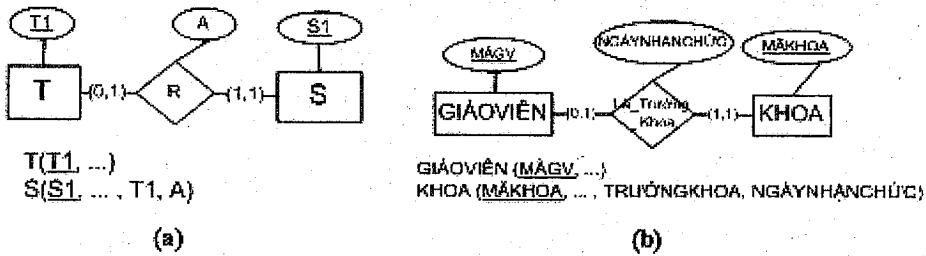
Tùy thuộc vào ràng buộc trên tập mối kết hợp mà ta có cách chuyển đổi chúng sang mô hình quan hệ khác nhau.

Tập mối kết hợp có 2 chỉ số *max* trên 2 bản số tương ứng là 1-1

Chương 3 Mô hình dữ liệu quan hệ

Ta muốn đề cập đến tập mối kết hợp bậc 2 có bản số là $(-,1):(-,1)$, với $-$ là chỉ số *min* mà ta không quan tâm giá trị của nó. Đôi với từng tập mối kết hợp R thuộc loại này trong mô hình TT-KH, xác định các quan hệ S và T ứng với các tập thực thể tham gia vào R. Gọi S là quan hệ ứng với tập thực thể tham gia toàn phần vào R. Bổ sung khóa chính của quan hệ T vào S và tập thuộc tính này giữ vai trò là khóa ngoại của S. Tất cả các thuộc tính đơn của R là các thuộc tính của S.

Trong ví dụ, tập mối kết hợp Là_Trưởng_Khoa (Hình III-2) thuộc loại này. KHOA đóng vai trò S vì tập thực thể KHOA tham gia toàn phần vào mối kết hợp này. Vì vậy, ta thêm khóa chính của quan hệ GIÁOVIÊN vào quan hệ KHOA, đặt lại tên thuộc tính này là TRƯỞNGKHOA và đây là khóa ngoại cho quan hệ KHOA. Thuộc tính NGÀYNHẬNCHỨC là thuộc tính của quan hệ KHOA.

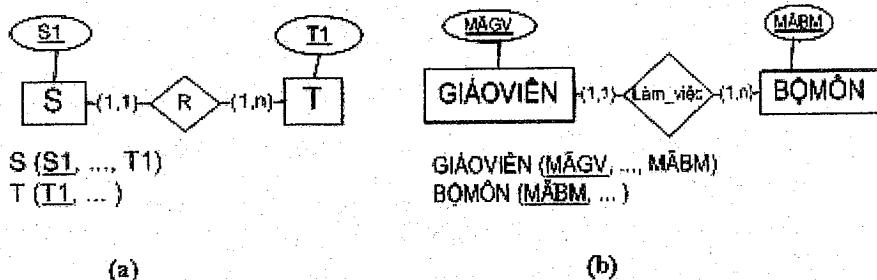


Hình III-2 Cách chuyển đổi tập mối kết hợp 1-1 sang quan hệ. (a) Trường hợp tổng quát. (b) Minh họa với tập mối kết hợp Là_Trưởng_Khoa.

Tập mối kết hợp có 2 chỉ số *max* trên 2 bản số tương ứng là 1-n

Ta muốn đề cập đến các tập mối kết hợp bậc 2 có bản số tương ứng là: $(-,1):(-,n)$ với $-$ là chỉ số *min* mà ta không quan tâm giá trị của nó. Gọi S, T là hai quan hệ ứng với hai tập thực thể tham gia vào tập mối kết hợp R, T là quan hệ ứng với tập thực thể phía bên n. Bổ sung khóa chính của T vào S và tập thuộc tính này giữ vai trò khóa ngoại của S. Các thuộc tính đơn của R là thuộc tính của S.

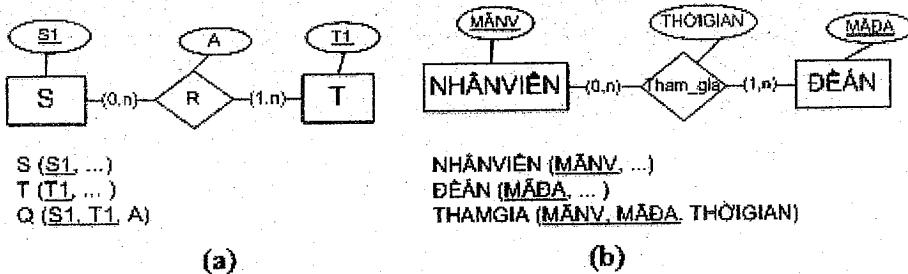
Với mỗi kết hợp Làm_việc (Hình III-3), ta bổ sung khóa chính của BỘMÔN (MÃBM) vào quan hệ GIÁOVIÊN và MÃBM trong quan hệ GIÁOVIÊN giữ vai trò khóa ngoại của quan hệ này.



Hình III-3 Cách chuyển đổi tập mối kết hợp 1-n sang quan hệ. (a) Trường hợp tổng quát. (b) Minh họa với loại mối kết hợp Làm_việc.

Tập mối kết hợp có 2 chỉ số max trên 2 bản số tương ứng là n-m

Ta muốn đề cập đến các tập mối kết hợp bậc 2 có bản số tương ứng là: $(-,n):(-,m)$ với $-$ là chỉ số *min* mà ta không quan tâm giá trị của nó. Đối với từng tập mối kết hợp R thuộc loại này, ta tạo ra một quan hệ mới Q ứng với R. Thuộc tính của Q là tổ hợp khóa của các quan hệ ứng với các tập thực thể tham gia vào R và các thuộc tính riêng của tập mối kết hợp. Khóa của Q được xác định từ các thuộc tính khóa của quan hệ ứng với các tập thực thể tham gia vào R và dựa trên các bản số *max* của mối kết hợp.



Hình III-4 Cách chuyển đổi tập mối kết hợp n-m sang quan hệ. (a) Trường hợp tổng quát. (b) Ví dụ minh họa.

Trong ví dụ, mỗi quan hệ Tham_gia giữa hai tập thực thể GIÁOVIÊN và CÔNGVIỆC thuộc loại này (lưu ý CÔNGVIỆC là tập thực thể yếu). Ta tạo ra 1 quan hệ tên là THAMGIAĐT. Quan hệ THAMGIAĐT có:

- Tập thuộc tính là khóa của GIÁOVIÊN là {MÃGV}.
- Tập thuộc tính là khóa của CÔNGVIỆC là {MÃĐT, STT}.
- Tập thuộc tính riêng của mỗi kết hợp là {PHỤCẤP, KẾTQUẢ}.

Như vậy quan hệ THAMGIAĐT là như sau:

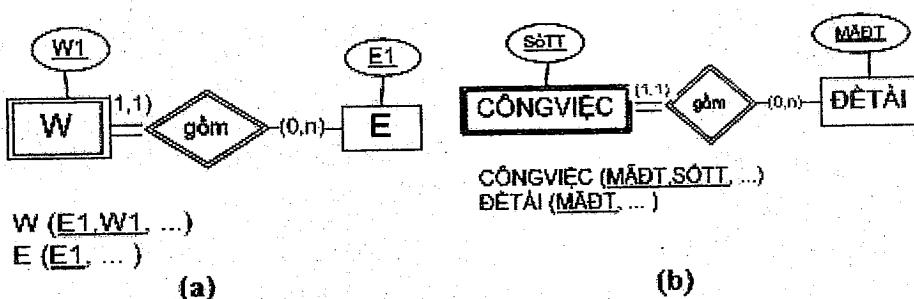
THAMGIAĐT (MÃGV, MÃĐT, STT, PHỤCẤP, KẾTQUẢ)

MÃGV và {MÃĐT, STT} trong lược đồ quan hệ THAMGIAĐT là hai khóa ngoại. Khóa của THAMGIAĐT là tập thuộc tính {MÃGV, MÃĐT, STT}.

IV.3 Tập thực thể yếu sang quan hệ

Đối với từng tập thực thể yếu W có mỗi kết hợp với tập thực thể mạnh E, tạo một quan hệ R tương ứng. Tất cả các thuộc tính đơn của W là các thuộc tính của R. Bổ sung khóa chính của quan hệ ứng với tập thực thể mạnh E vào R và là khóa ngoại của R. Khóa chính của R là sự kết hợp khóa chính của quan hệ ứng với tập thực thể mạnh E và khóa riêng phần của tập thực thể yếu W.

Trong ví dụ, CÔNGVIỆC (Hình III-5) là tập thực thể yếu và ĐỀTÀI là tập thực thể chủ. Ta tạo quan hệ CÔNGVIỆC gồm các thuộc tính của tập thực thể yếu CÔNGVIỆC, thêm vào đó là thuộc tính MÃĐT – là khóa của ĐỀTÀI, giữ vai trò là khóa ngoại của CÔNGVIỆC. Khóa chính của CÔNGVIỆC là {MÃĐT, STT}.

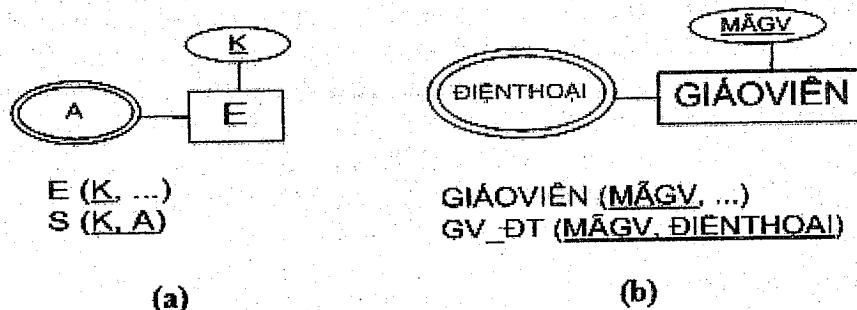


Hình III-5 Cách chuyển đổi tập thực thể yếu sang quan hệ. (a) Trường hợp tổng quát. (b) Ví dụ minh họa.

IV.4 Thuộc tính đa trị

Đối với từng thuộc tính đa trị A, tạo ra một quan hệ mới S gồm chính thuộc tính A và khóa chính K của quan hệ ứng với tập thực thể/ tập mối kết hợp mà A là thuộc tính của nó. Khóa chính của S là kết hợp A và K, K giữ vai trò là khóa ngoại trong S. Nếu thuộc tính đa trị A là thuộc tính kết hợp, ta chỉ lấy các thành phần đơn của nó.

Trong ví dụ, thuộc tính **ĐIỆNTHOẠI** của tập thực thể **GIÁOVIÊN** là thuộc tính đa trị (Hình III-6). Ta tạo quan hệ **GV_DT** gồm khóa chính **MÃGV** của **GIÁOVIÊN** và thuộc tính đa trị **ĐIỆNTHOẠI**. **MÃGV** là khóa ngoại của quan hệ **GV_DT**. Khóa chính của **GV_DT** là **{MÃGV, ĐIỆNTHOẠI}**.



Hình III-6 Cách chuyển đổi thuộc tính đa trị sang quan hệ. (a) Trường hợp tổng quát. (b) Ví dụ minh họa.

Chương 3 Mô hình dữ liệu quan hệ

Sau khi hoàn thành các bước chuyển đổi, ta có lược đồ CSDL như sau:

1. GIÁO VIÊN (MÃGV, HỌ TÊN, LƯƠNG, PHÁI, NGÀY SINH, SỐ NHÀ, ĐƯỜNG, QUẬN, THÀNH PHỐ, GVQLCM, MÃ BM)
2. GV ĐT (MÃGV, ĐIỆN THOẠI)
3. BỘ MÔN (MÃBM, TÊN BM, PHÒNG, ĐIỆN THOẠI, TRƯỞNG GBM, MÃ KHOA, NGÀY NHẬN CHỨC)
4. KHOA (MÃ KHOA, TÊN KHOA, NĂM TL, PHÒNG, ĐIỆN THOẠI, TRƯỞNG KHOA, NGÀY NHẬN CHỨC)
5. ĐỀ TÀI (MÃ ĐT, TÊN ĐT, KINH PHÍ, CÁP QL, NGÀY BĐ, NGÀY KKT, MÃ CĐ, GVCN ĐT)
6. CHỦ ĐỀ (MÃ CĐ, TÊN CĐ)
7. CÔNG VIỆC (MÃ ĐT, STT, TÊN CV, NGÀY BĐ, NGÀY KKT)
8. THAM GIA ĐT (MÃGV, MÃ ĐT, STT, PHỤC CẤP, KẾT QUẢ)

V. KẾT LUẬN

Chương 3 giới thiệu các khái niệm liên quan đến mô hình dữ liệu quan hệ gồm lược đồ quan hệ, quan hệ, thuộc tính của quan hệ, miền giá trị của thuộc tính, khóa, khóa chính, khóa ngoại, lược đồ CSDL, thể hiện CSDL và cách chuyển từ mô hình TT-KH sang mô hình quan hệ. Chương 4 sẽ giới thiệu khái niệm phụ thuộc hàm và phương pháp chuẩn hóa một lược đồ CSDL. Ngoài ra, chúng tôi cũng lưu ý rằng trong các chương sau, thuật ngữ “quan hệ” có thể được dùng theo nghĩa lược đồ quan hệ và được ký hiệu “R”; thuật ngữ “thể hiện của lược đồ” được ký hiệu “r”.

Câu hỏi ôn tập

1. Hãy định nghĩa các khái niệm sau: lược đồ quan hệ, quan hệ, bộ, thuộc tính của một quan hệ, miền giá trị của thuộc tính, bậc của một quan hệ, thể hiện quan hệ, khóa, khóa ứng viên, khóa chính, thuộc tính khóa, siêu khóa, khóa ngoại, lược đồ CSDL, thể hiện CSDL.

2. Cách lưu trữ dữ liệu dùng quan hệ có những đặc điểm gì khác so với cách lưu trữ dữ liệu dùng bảng hoặc tập tin thông thường?
3. Hãy cho biết khóa chính của quan hệ có vai trò gì?
4. Hãy phân biệt khóa và siêu khóa của một quan hệ?
5. Hãy cho biết khóa ngoại của quan hệ có vai trò gì? Nếu không khai báo khóa ngoại khi cài đặt một cơ sở dữ liệu thì gặp phải vấn đề gì?
6. Cho FK là khóa ngoại của quan hệ r2 tham chiếu đến PK là khóa chính của quan hệ r1. Hãy cho biết:
 - a. Số lượng thuộc tính của FK có nhất thiết bằng với số lượng thuộc tính của PK hay không?
 - b. Tên của các thuộc tính trong FK có nhất thiết trùng tên với các thuộc tính trong PK hay không?
 - c. Các thuộc tính trong FK có nhất thiết cùng miền giá trị với các thuộc tính trong PK hay không?
7. Hãy cho biết một thuộc tính của một quan hệ có thể mang giá trị rỗng (null) trong những trường hợp nào?
8. Giá trị của một bộ tại các thuộc tính tham gia làm khóa chính của một quan hệ có thể là rỗng (null) hay không? Vì sao?
9. Giá trị của một bộ tại các thuộc tính tham gia làm khóa ngoại của một quan hệ có thể là rỗng (null) hay không? Vì sao?
10. Khi chuyển mô hình ER sang mô hình dữ liệu quan hệ, trong những trường hợp nào thì phải tạo ra một quan hệ mới? Hãy cho ví dụ cụ thể cho những trường hợp nhận biết được.
11. Hãy trình bày cách thức chuyển đổi các dạng loại mối kết hợp từ mô hình ER sang mô hình quan hệ. Với mỗi dạng loại mối kết hợp, hãy cho ví dụ minh họa.

Bài tập

1. Hãy chuyển các mô hình ER trong bài tập chương 2 sang mô hình quan hệ. Hãy xác định khóa chính, khóa ngoại cho từng quan hệ.
2. Phòng giáo vụ tại một trường đại học muốn tin học hóa việc quản lý đăng ký học các chuyên đề của sinh viên. Sau đây là kết quả của việc phân tích thiết kế ứng dụng trên:
SINHVIEN (MÃSV, HỌTÊN, PHÁI, NGÀYSINH, ĐCHỈ, MÃNGÀNH)
Tân từ: Mỗi sinh viên có một mã số duy nhất, một họ tên, thuộc một phái, có một ngày sinh, một địa chỉ và theo học một ngành.

Chương 3 Mô hình dữ liệu quan hệ

NGÀNH (MÃNGÀNH, TÊNNGÀNH, SỐCĐ, TSSV)

Tân từ: Mỗi ngành có một mã ngành duy nhất, có một tên ngành duy nhất. SỐCĐ cho biết số lượng chuyên đề mà 1 sinh viên theo học ngành có mã là MÃNGÀNH phải học. TSSV cho biết tổng số sinh viên đã từng theo học ngành này từ trước đến nay.

CHUYÊNĐỀ (MÃCĐ, TÊNCĐ, SỐSVTĐ)

Tân từ: Mỗi chuyên đề có một mã duy nhất và có một tên duy nhất. SỐSVTĐ cho biết số sinh viên tối đa có thể chấp nhận được mỗi khi có 1 lớp mở cho chuyên đề có mã là MÃCĐ.

CĐ_NGANH (MÃCĐ, MÃNGÀNH)

Tân từ: Mỗi chuyên đề có thể được học bởi nhiều ngành và mỗi ngành phải học nhiều chuyên đề.

CĐ_MỞ (MÃCĐ, HOCKỲ, NĂM)

Tân từ: Mỗi bộ của quan hệ trên thể hiện một chuyên đề được mở ra vào một học kỳ của một năm học.

ĐĂNGKÝ (MÃSV, MÃCĐ, HOCKỲ, NĂM, ĐIỂM)

Tân từ: Mỗi bộ của quan hệ trên thể hiện việc đăng ký học một chuyên đề của một sinh viên vào một học kỳ của một năm học. ĐIỂM cho biết điểm số mà sinh viên đó đạt được khi học chuyên đề. Sinh viên chỉ được đăng ký học chuyên đề khi chuyên đề đó có mở. Sinh viên theo học mỗi ngành không được học quá 5 chuyên đề. Sinh viên không được đăng ký học quá 3 chuyên đề trong một học kỳ. Mỗi một học kỳ mở tối đa là 5 chuyên đề.

Hãy xác định khóa chính, khóa ngoại cho các lược đồ quan hệ trên. Chỉ rõ các thuộc tính mà từng khóa ngoại tham chiếu đến.

3. Cho lược đồ CSDL sau:

NHÀCC(MÃNCC, TÊNNCC, ĐỊACHỈ, ĐT)

Tân từ: Một nhà cung cấp có một mã nhà cung cấp, tên, địa chỉ và điện thoại của nhà cung cấp.

HÀNGHÓA(MÃHH, TÊNHH, ĐVT, QUYCÁCH, SLTỒN)

Tân từ: Càn lưu lại thông tin về tất cả các mặt hàng mà cửa hàng có mua bán: mã mặt hàng, tên hàngđơn vị tính, quy cách, số lượng tồn.

CUNGÚNG(MÃNCC, MÃHH)

Tân từ: Mỗi nhà cung cấp có thể cung ứng nhiều mặt hàng khác nhau và mỗi mặt hàng cũng có thể được cung cấp bởi nhiều nhà

cung cấp khác nhau, cần ghi nhận lại nhà cung cấp nào có thể cung ứng những mặt hàng gì.

ĐĐH(SỐĐĐH, NGÀYĐH, MÃNCC)

Tân từ: Mỗi đơn đặt hàng có một số đơn đặt hàng duy nhất, ngày đặt hàng, đặt tại nhà cung cấp nào.

CTĐĐH(SỐĐĐH, MÃHH, SỐLƯỢNG)

Tân từ: Mỗi đơn đặt hàng đặt nhiều mặt hàng khác nhau, mỗi mặt hàng ghi rõ số lượng đặt hàng. Đơn đặt hàng gửi đến một nhà cung cấp chỉ gồm các mặt hàng mà nhà cung cấp đó có thể cung ứng.

GIAOHÀNG(SỐGH, NGÀYGH, SỐĐĐH)

Tân từ: Mỗi phiếu giao hàng có một số phiếu giao duy nhất, có ngày giao, giao cho đơn đặt hàng nào

CTGH(SỐGH, MÃHH, SỐLƯỢNG)

Tân từ: Trong phiếu giao hàng cần ghi nhận mỗi mặt hàng có số lượng giao và đơn giá là bao nhiêu.

HÓAĐƠN(SỐHĐ, NGÀYHĐ, TÊNKH)

Tân từ: Mỗi hóa đơn có số hóa đơn duy nhất, ngày lập hóa đơn, tên khách hàng.

CTHĐ(SỐHĐ, MÃHH, SỐLƯỢNG, ĐÓNGIÁ)

Tân từ: Cần ghi nhận đối với từng hóa đơn khách hàng đã mua những mặt hàng nào với số lượng mua, đơn giá mua là bao nhiêu.

Ngoài ra, ứng với 1 lần đặt hàng, nhà cung cấp có thể giao hàng tối đa là 3 lần và không được trễ hơn 7 ngày so với ngày đặt. Nhà cung cấp chỉ được giao các mặt hàng mà nhà cung cấp có đặt với số lượng giao không lớn hơn số lượng đặt.

Hãy xác định khóa chính, khóa ngoại cho các lược đồ quan hệ trên.

4. Cho lược đồ CSDL sau:

ĐỀÁN(MÃĐA, NGÀYBD, NGÀYKT, TTRẠNG, MÃPB, CHIPHÍĐA)

Tân từ: Mỗi đề án có một mã số duy nhất để phân biệt với các đề án khác, có thời gian thực hiện đề án tính từ ngày bắt đầu (NGÀYBD) đến ngày kết thúc (NGÀYKT). Tình trạng cho biết đề án chưa thực thi, đang thực thi, thành công hay thất bại. Đề án do một phòng ban phụ trách và có một chi phí để thực hiện đề án.

CÔNGĐOẠN(MÃĐA, STTCĐ, NGÀYBD, NGÀYKT, MÃCV)

Tân từ: Mỗi công đoạn của một đề án có số thứ tự (STTCĐ) khác nhau dùng để phân biệt với các công đoạn khác nhau trong đề án đó, có ngày bắt đầu (NGÀYBD) và kết thúc (NGÀYKT) công đoạn. Mỗi công đoạn triển khai duy nhất cho một công việc (MÃCV).

THAMGIACĐ(MÃNV, MÃĐA, STTCĐ)

Tân từ: Mỗi bộ của quan hệ trên cho biết một nhân viên tham gia vào một công đoạn trong một đề án.

CÔNGVIỆC(MÃCV, TÊNCV, CHIPHÍCV)

Tân từ: Một công việc có một mã số duy nhất (MÃCV), có tên công việc (TÊNCV) và chi phí để triển khai công việc (CHIPHÍCV).

KHẢNĂNG(MÃNV, MÃCV)

Tân từ: Một nhân viên có khả năng thực hiện một số công việc nào đó mà thôi. Nhân viên chỉ được tham gia vào các công đoạn thuộc công việc mà nhân viên đó có khả năng thực hiện.

Hãy xác định khóa chính, khóa ngoại cho các lược đồ quan hệ trong lược đồ CSDL trên.

Chương 4

PHỤ THUỘC HÀM VÀ DẠNG CHUẨN

I. PHỤ THUỘC HÀM

Một khái niệm quan trọng trong lý thuyết thiết kế CSDL quan hệ là phụ thuộc hàm. Đây là một loại RBTV được sử dụng trong quá trình tạo nên một lược đồ CSDL được thiết kế tốt. Trong chương này chúng tôi sẽ đề cập đến các khái niệm liên quan đến phụ thuộc hàm và cách thức vận dụng chúng trong quá trình thiết kế CSDL.

I.1 Định nghĩa phụ thuộc hàm

Một phụ thuộc hàm là một ràng buộc giữa hai tập thuộc tính của một lược đồ quan hệ. Giả sử chúng ta có lược đồ quan hệ $R(A_1, A_2, \dots, A_n)$ và X, Y là tập con của $\{A_1, A_2, \dots, A_n\}$. Ta nói Y phụ thuộc hàm vào X , ký hiệu $X \rightarrow Y$ nếu mỗi giá trị tại X trong R xác định một giá trị duy nhất của Y trong R . Nghĩa là:

- Một thể hiện của R thỏa phụ thuộc hàm $X \rightarrow Y$ nếu:
 $\forall r_1, r_2 \in R$ sao cho $r_1.X = r_2.X$ thì $r_1.Y = r_2.Y$.
- Nếu $r_1.X = r_2.X$ mà $r_1.Y \neq r_2.Y$ thì thể hiện của lược đồ R vi phạm phụ thuộc hàm $X \rightarrow Y$.

Ví dụ xét lược đồ quan hệ:

PHIM (TÊNPHIM, NĂMSX, THỜI LƯỢNG, LOAIPHIM, XƯỞNG-SX, DIỄN VIÊN)

Chương 4 Phụ thuộc hàm và dạng chuẩn

và thể hiện của lược đồ ở

Hình IV-1:

Tên phim	Năm sx	Thời lượng	Loại phim	Xưởng sx	Diễn viên
Star Wars	1977	124	color	Fox	Carrie Fisher
Star Wars	1977	124	color	Fox	Mark Hamill
Star Wars	1977	124	color	Fox	Harrison Ford
Mighty Ducks	1991	104	color	Disney	Emilio Estevez
Wayne's World	1992	95	color	Paramount	Dana Carvey
Wayne's World	1992	95	color	Paramount	Mike Meyers

Hình IV-1 Thể hiện của lược đồ PHIM.

Thể hiện của lược đồ PHIM thỏa các phụ thuộc hàm:

TÊNPHIM → LOAIPHIM

TÊNPHIM → THỜI LƯỢNG

TÊNPHIM → XƯỞNGSX

I.2 Một số tính chất của phụ thuộc hàm

Cho lược đồ R , gọi R^+ là tập thuộc tính của R ; $X, Y \subseteq R^+$. Cho $X \rightarrow Y$ là phụ thuộc hàm thuộc R .

Phụ thuộc bắc cầu: Ta nói $X \rightarrow Y$ là phụ thuộc hàm bắc cầu, hay Y phụ thuộc bắc cầu vào X nếu $\exists Z \subseteq R^+$ sao cho $X \rightarrow Z$ và $Z \rightarrow Y$ đều đúng trên R .

Phụ thuộc đầy đủ: Ta nói $X \rightarrow Y$ là phụ thuộc hàm đầy đủ, hay Y phụ thuộc đầy đủ vào X , nếu $\forall A \in X, \{X - A\} \rightarrow Y$ không đúng trên R .

Ví dụ: Cho $R(A, B, C, D, E, I)$ và tập phụ thuộc hàm:

$$F = \{A \rightarrow BCD; BCD \rightarrow E; CD \rightarrow E\}$$

Phụ thuộc hàm $f: BCD \rightarrow E$ là phụ thuộc hàm không đầy đủ vì có phụ thuộc hàm $CD \rightarrow E$.

II. DẠNG CHUẨN

II.1 Thiết kế lược đồ CSDL tốt

II.1.1 Trùng lặp các giá trị trên các bộ

Một trong các mục tiêu của thiết kế lược đồ là làm giảm tối đa thông tin trùng lặp nhằm sử dụng hiệu quả không gian lưu trữ. Sự hiện diện của các thuộc tính trong một quan hệ đóng vai trò rất quan trọng trong việc hạn chế sự trùng lặp này.

Xét quan hệ PHIM và thể hiện của chúng ở Hình IV-2. Giá trị trên tập thuộc tính minh họa cho mỗi phim: TÊNPHIM, NĂMSX, THỜI LƯỢNG, LOAIPHIM, XUỐNGSX bị trùng lặp.

Tên phim	Năm sx	Thời lượng	Loại phim	Xưởng sx	Diễn viên
Star Wars	1977	124	color	Fox	Carrie Fisher
Star Wars	1977	124	color	Fox	Mark Hamill
Star Wars	1977	124	color	Fox	Harrison Ford
Mighty Ducks	1991	104	color	Disney	Emilio Esteves
Wayne's World	1992	95	color	Paramount	Dana Carvey
Wayne's World	1992	95	color	Paramount	Mike Meyers

Hình IV-2 Minh họa sự trùng lặp thông tin.

Việc lưu trữ thông tin trùng lặp còn gây ra một số弊端 thường khi thao tác trên dữ liệu như sau:

- *Thêm* một bộ mới chúng ta phải thêm chính xác tất cả các giá trị bị trùng lặp. Ví dụ với quan hệ PHIM trên khi thêm một diễn viên mới cho bộ phim Star Wars chúng ta phải thêm chính xác trên các thuộc tính tên phim, năm sản xuất, thời lượng, loại phim, sưởng sản xuất.

- Xóa có khả năng sẽ làm mất thông tin. Ví dụ với quan hệ PHIM trên, giả sử chúng ta muốn xóa diễn viên Emilio Esteves vô tình chúng ta xóa luôn thông tin bộ phim Mighty Ducks.
- Khi sửa giá trị trên một bộ, chúng ta cần kiểm tra thông tin trùng lặp có còn nhất quán không. Ví dụ ở quan hệ PHIM trên, chúng ta muốn sửa thời lượng của bộ phim Star Wars thành 120 phút, chúng ta phải sửa trên cả 3 bộ dữ liệu của phim Star Wars thì dữ liệu mới nhất quán được.

II.1.2 Giá trị rỗng trong các bộ

Nếu một thuộc tính không được áp dụng cho một bộ trong một quan hệ thì chúng ta đặt giá trị rỗng (*null*) tại thuộc tính đó. Điều này gây lãng phí không gian lưu trữ và có thể ảnh hưởng đến việc hiểu nghĩa của thuộc tính. Thực vậy, giá trị rỗng có thể có nhiều nghĩa. Sau đây là vài nghĩa phổ biến:

- Thuộc tính này không áp dụng cho bộ này.
- Giá trị của thuộc tính trong bộ này là không biết.
- Giá trị của thuộc tính trong bộ này được xác định nhưng chưa được nhập.

Tuy nhiên, trong một số thể hiện thì một số giá trị rỗng là hợp lệ theo nghĩa của chúng. Do vậy chúng ta cũng có thể cân nhắc để dùng nó.

II.2 Tiếp cận phân rã quan hệ

Theo quan điểm của cách tiếp cận này, các quan hệ con của cấu trúc CSDL ban đầu sẽ lần lượt được phân rã thành những quan hệ con với số lượng thuộc tính ít hơn, sao cho cấu trúc kết quả không còn trùng lặp thông tin và đạt được các tiêu chuẩn đề ra ở mức cao nhất. Quá trình phân rã là một quá trình được lặp lại đối với các quan hệ con nào được đánh giá còn có thể phân rã.

II.3 Dạng chuẩn 1

Định nghĩa: Một quan hệ đạt dạng chuẩn 1 (DC1) là một quan hệ mà các giá trị trên từng thuộc tính phải là giá trị nguyên tố và còn được gọi là cấu trúc phẳng.

Ví dụ: Cho quan hệ GIÁNGDẠY gồm có các thuộc tính và các thể hiện như

(mã số cán bộ GD)	(môn học)	(tên)	(học hàm)	(mức lương)	(tổng số SV)
MS_CBGD	MS_MH	T_CBGD	HH_CDGD	ML_TT	TSSV
3217	CA01	Ng.v.A	GVC	425	87
3217	CA02	Ng.v.A	GVC	425	32
4096	MA03	Trần.t.B	PGS	500	49
4371	CA11	Lê.v.C	GS	600	36
4371	CA21	Lê.v.C	GS	600	37
4371	CA41	Lê.v.C	GS	600	32
4589	MA11	Võ.v.X	GVC	425	47
4589	MA13	Võ.v.X	GVC	425	69
4672	CA22	Hoàng.t.Y	GVC	425	28
4672	CA01	Hoàng.t.Y	GVC	425	96

Hình IV-3. Ở quan hệ giảng dạy tất cả giá trị của thuộc tính đều là nguyên tố, nghĩa là quan hệ không chứa các trường lặp và trường kép. Do vậy quan hệ GIÁNG DẠY đạt DC1.

Chương 4 Phụ thuộc hàm và dạng chuẩn

(mã số cán bộ GD)	(môn học)	(tên)	(học hàm)	(mức lương)	(tổng số SV)
MS_CBGD	MS_MH	T_CBGD	HH_CDGD	ML_TT	TSSV
3217	CA01	Ng.v.A	GVC	425	87
3217	CA02	Ng.v.A	GVC	425	32
4096	MA03	Trần.t.B	PGS	500	49
4371	CA11	Lê.v.C	GS	600	36
4371	CA21	Lê.v.C	GS	600	37
4371	CA41	Lê.v.C	GS	600	32
4589	MA11	Võ.v.X	GVC	425	47
4589	MA13	Võ.v.X	GVC	425	69
4672	CA22	Hoàng.t.Y	GVC	425	28
4672	CA01	Hoàng.t.Y	GVC	425	96

Hình IV-3 Minh họa cho dạng chuẩn 1.

Nhận xét DC1: Ưu điểm của CSDL đạt dạng chuẩn này là cho phép đặt các câu hỏi chỉ dựa vào nghĩa của CSDL chứ không dựa vào hình thức lưu trữ thông tin. Tuy nhiên CSDL đạt DC1 không thể đáp ứng cho một số dạng câu hỏi và sự trùng lặp thông tin chưa được giải quyết. Sự trùng lặp thông tin sẽ dẫn đến một số dị thường trong quá trình khai thác dữ liệu. Các vấn đề này đã được nêu ở phần II.1.1 của chương.

III.4 Dạng chuẩn 2

Định nghĩa: Một quan hệ đạt ở dạng chuẩn 2 (DC2) nếu và chỉ nếu nó đạt DC1 và tất cả các thuộc tính không khóa phụ thuộc đầy đủ vào khóa.

Chú ý: Trước khi xác định dạng chuẩn của một quan hệ R, chúng ta phải xác định tất cả các khóa của R để xác định những thuộc tính nào của R là thuộc tính khóa/không khóa.

Ví dụ: Cho quan hệ GIÁNGDẠY ở

(mã số cán bộ GD)	(môn học)	(tên)	(học hàm)	(mức lương)	(tổng số SV)
MS_CBGD	MS_MH	T_CBGD	HH_CDGD	ML_TT	TSSV
3217	CA01	Ng.v.A	GVC	425	87
3217	CA02	Ng.v.A	GVC	425	32
4096	MA03	Trần.t.B	PGS	500	49
4371	CA11	Lê.v.C	GS	600	36
4371	CA21	Lê.v.C	GS	600	37
4371	CA41	Lê.v.C	GS	600	32
4589	MA11	Võ.v.X	GVC	425	47
4589	MA13	Võ.v.X	GVC	425	69
4672	CA22	Hoàng.t.Y	GVC	425	28
4672	CA01	Hoàng.t.Y	GVC	425	96

Hình IV-3, và tập phụ thuộc hàm:

$$F = \{ \begin{aligned} f1: & MS_CBGD \rightarrow T_CBGD, HH_CBGD, ML_TT \\ f2: & HH_CBGD \rightarrow ML_TT \\ f3: & MS_CBGD, MS_MH \rightarrow TSSV \end{aligned} \}$$

Ta dễ dàng nhận thấy khóa của GIÁNGDẠY là (MS_MH, MS_CBGD).

Quan hệ GIÁNG DẠY không đạt DC2, lý do phụ thuộc hàm f1 làm các thuộc tính không khóa (T_CBGD, HH_CBGD, ML_TT) không phụ thuộc đầy đủ vào khóa (MS_MH, MS_CBGD).

Để biến đổi từ DC1 thành DC2 ta phải xác định quan hệ đang ở DC1 và sau đó loại các phụ thuộc hàm không đầy đủ trên khóa chính ra khỏi quan hệ cũ bằng cách thay thế chúng bằng một quan hệ mới. Sau đó đổi tên các quan hệ (nếu cần) để phù hợp với thông tin lưu trữ bên trong nó.

Chương 4 Phụ thuộc hàm và dạng chuẩn

Với ví dụ trên quan hệ GIÁNGDẠY được tách ra làm hai quan hệ và lược đồ CSDL đạt DC2:

CÁNBỘ (MS_CBGD, T_CBGD, HH_CBGD, ML_TT)

GIÁNGDẠY (MS_CBGD, MS_MH, TSSV)

CBGD

MS_CBGD	T_CBGD	HH_CBGD	ML_TT
3217	Ng.v.A	GVC	425
4096	Trần.t.B	PGS	500
4371	Lê.v.C	GS	600
4589	Võ.v.X	GVC	425
4672	Hoàng.t.Y	GVC	425

Vẫn còn trùng
lặp thông tin

Hình IV-4 Thể hiện của quan hệ CÁNBỘ.

GIÁNGDẠY

MS_CBGD	MS_MH	TSSV
3217	CA01	87
3217	CA02	32
4096	MA03	49
4371	CA11	36
4371	CA21	37
4371	CA41	32
4589	MA11	47
4589	MA13	69
4672	CA22	28
4672	CA01	96

Hình IV-5 Thể hiện của quan hệ GIÁNGDẠY.

Nhận xét DC2: Thông tin còn trùng lặp, do vậy còn gây ra các tình trạng dị thường khi thao tác (thêm, xóa, sửa) dữ liệu.

II.5 Dạng chuẩn 3

Định nghĩa: Một quan hệ đạt dạng chuẩn 3 (DC3) khi nó đạt DC2 và tất cả các thuộc tính không khóa không phụ thuộc bắc cầu vào khóa.

Ví dụ: Xét quan hệ CÁNBỘ ở Hình IV-4, vì tồn tại phụ thuộc hàm f2: HH_CBGD → ML_TT, do vậy thuộc tính ML_TT là thuộc tính không khóa phụ thuộc bắc cầu vào khóa:

MS_CBGD → HH_CBGD (i)

HH_CBGD → ML_TT (ii)

Từ (i) và (ii) suy ra: MS_CBGD → ML_TT.

Để chuyển từ DC2 lên DC3 ta phải xác định quan hệ đang ở DC2, loại các phụ thuộc hàm gây ra các thuộc tính không khóa phụ thuộc bắc cầu vào khóa bằng cách đưa chúng ra quan hệ mới. Sau đó đổi tên các quan hệ (nếu cần) để phù hợp với thông tin lưu trữ bên trong nó.

Ví dụ xét quan hệ CÁNBỘ, phụ thuộc hàm HH_CBGD → ML_TT gây cho thuộc tính ML_TT phụ thuộc bắc cầu vào khóa. Do vậy ta xóa và đưa nó ra một quan hệ mới là HỌCHÀM. Như vậy CÁNBỘ được tách làm hai quan hệ mới và lược đồ đạt dạng DC3:

CÁNBỘ(MS_CBGD, T_CBGD, HH_CBGD)

HỌCHÀM(HH_CBGD, ML_TT)

Chương 4 Phụ thuộc hàm và dạng chuẩn

CÁNBỘ

MS_CBGD	T_CBGD	HH_CBGD
3217	Ng.v.A	GVC
4096	Trần.t.B	PGS
4371	Lê.v.C	GS
4589	Võ.v.X	GVC
4672	Hoàng.t.Y	GVC

Hình IV-6 Quan hệ CÁNBỘ được chuẩn hóa về DC3.

HỌCHÀM

HH_CBGD	ML_TT
GVC	425
PGS	500
GS	600

Hình IV-7 Thể hiện của quan hệ HỌCHÀM.

Nhận xét DC3: Mức độ trùng lặp thông tin có giảm bớt, tuy nhiên trong một số trường hợp khi xóa dữ liệu sẽ bị mất thông tin.

II.6 Dạng chuẩn BCK (Boyce Codd Kent)

Định nghĩa: Một quan hệ đạt DC BCK khi nó đạt DC3 và với mọi phụ thuộc hàm $X \rightarrow A$ không hiển nhiên định nghĩa trên quan hệ, $A \notin X$ thì X là một siêu khóa của quan hệ, nghĩa là X là một khóa hoặc chứa một khóa.

Ví dụ: Với lược đồ GIÁNGDẠY được phân rã thành lược đồ mới gồm ba quan hệ:

GIÁNGDẠY (MS_CBGD, MS_MH, TSSV)

CÁNBỘ (MS_CBGD, T_CBGD, HH_CBGD)

HỌCHÀM (HH_CBGD, ML_TT)

Lược đồ này đạt DC3 và cũng đạt được DC BCK, vì tất cả các phụ thuộc hàm xác định được đều dựa trên khóa.

Nhận xét: Quan hệ đạt DC BCK thì hiển nhiên đạt DC3, tuy nhiên quan hệ đạt DC3 chưa chắc đạt DC BCK.

Chúng ta xem xét quan hệ thông tin phòng vấn (TT_PV) gồm có các thuộc tính: mã ứng viên (MÃ_UV), ngày phỏng vấn (NGÀY_PV), giờ phỏng vấn (GIỜ_PV), mã nhân viên (MÃ_NV), phòng phỏng vấn (PHÒNG_PV). Khóa của TT_PV gồm 3 khóa: {MÃ_UV, NGÀY_PV}; {MÃ_NV, NGÀY_PV, GIỜ_PV}; {PHÒNG_PV, NGÀY_PV, GIỜ_PV} và thể hiện dữ liệu được thể hiện ở Hình IV-8:

MÃ_UV	NGÀY_PV	GIỜ_PV	MÃ_NV	PHÒNG_PV
CR76	13/05/2005	10:30	SG5	G101
CR56	13/05/2005	12:00	SG5	G101
CR74	13/05/2005	12:00	SG37	G102
CR56	1/6/2005	10:30	SG5	G102

Hình IV-8 Thể hiện của quan hệ TT_PV.

Giả sử trên quan hệ TT_PV có tập phụ thuộc hàm:

$$\begin{aligned}
 F = \{ & f_1: MÃ_UV, NGÀY_PV \rightarrow GIỜ_PV, MÃ_NV, PHÒNG_PV \\
 & f_2: MÃ_NV, NGÀY_PV \rightarrow PHÒNG_PV \\
 & f_3: MÃ_NV, NGÀY_PV, GIỜ_PV \rightarrow MÃ_UV \\
 & f_4: PHÒNG_PV, NGÀY_PV, GIỜ_PV \rightarrow MÃ_UV, MÃ_NV \}
 \end{aligned}$$

Quan hệ TT_PV đạt DC3 nhưng không đạt DC BCK do tồn tại f2 là phụ thuộc hàm không suy ra từ khóa.

Để biến đổi từ DC3 lên DC BCK ta xác định các phụ thuộc hàm $X \rightarrow A$ của Q, trong đó $A \notin X$ và X không là khóa. Xóa các phụ thuộc hàm này ra khỏi quan hệ gốc bằng cách đưa chúng ra các quan hệ mới. Đổi tên các quan hệ (nếu cần) để phù hợp với thông tin chúng lưu trữ.

Ví dụ với quan hệ TT_PV, do phụ thuộc hàm f2: MÃ_NV, NGÀY_PV \rightarrow PHÒNG_PV là phụ thuộc hàm không suy ra từ khóa, tách chúng ra một quan hệ mới, vậy lược đồ mới gồm 2 quan hệ:

Chương 4 Phụ thuộc hàm và dạng chuẩn

PHÒNGVÂN (MÃ_UV, NGÀY_PV, GIỜ_PV, MÃ_NV)

NV_PHÒNG (MÃ_NV, NGÀY_PV, PHÒNG_PV)

PHÒNGVÂN

MÃ_UV	NGÀY_PV	GIỜ_PV	MÃ_NV
CR76	13/05/2005	10:30	SG5
CR56	13/05/2005	12:00	SG5
CR74	13/05/2005	12:00	SG37
CR56	1/6/2005	10:30	SG5

NV_PHÒNG

MÃ_NV	NGÀY_PV	PHÒNG_PV
SG5	13/05/2005	G101
SG37	13/05/2005	G102
SG5	1/6/2005	G102

Hình IV-9 Quan hệ PHÒNGVÂN được chuẩn hóa thành DC BCK.

Nhận xét DC BCK: DC BCK quan tâm giải quyết vấn đề thông tin trùng lặp. Đó là mấu chốt cho mọi bất tiện thường xảy ra trong khai thác CSDL. Nhưng DC BCK lại coi nhẹ một tiêu chuẩn khác không kém quan trọng trong quá trình thiết kế CSDL đó là làm thế nào để kiểm tra phụ thuộc dữ liệu thuận lợi nhất.

II.7 Dạng chuẩn của lược đồ CSDL

Định nghĩa: Cho một cấu trúc CSDL $C = \{<R_i, F_i>\}$ với $1 \leq i \leq n$. Mỗi R_i sẽ được xác định dạng chuẩn của nó. Dạng chuẩn của C là dạng chuẩn thấp nhất trong các dạng chuẩn của các R_i .

Ví dụ: Cho lược đồ CSDL:

CÁNBỘ (MS_CBGD, T_CBGD, HH_CBGD, ML_TT)

GIÁNGDẠY (MS_CBGD, MS_MH, TSSV)

và tập phụ thuộc hàm:

$$F = \{ \begin{array}{l} f1: MS_CBGD \rightarrow T_CBGD, HH_CBGD, ML_TT \\ f2: HH_CBGD \rightarrow ML_TT \\ f3: MS_CBGD, MS_MH \rightarrow TSSV \end{array} \}$$

Dựa trên định nghĩa dạng chuẩn lược đồ ta thấy CÁNBỘ đạt DC2, lược đồ GIẢNGDAY đạt DC BCK. Như vậy dạng chuẩn của lược đồ CSDL là DC2.

II.8 Kết luận

Chương 4 trình bày khái niệm phụ thuộc hàm và các dạng chuẩn, cũng như việc vận dụng khái niệm phụ thuộc hàm trong quá trình chuẩn hóa lược đồ CSDL. Chương 5 trình bày về một ngôn ngữ thao tác trên cơ sở dữ liệu quan hệ, ngôn ngữ đại số quan hệ.

Câu hỏi ôn tập

1. Hãy cho biết thẻ nào là thêm, xóa, sửa dị thường? Hãy cho ví dụ minh họa.
2. Tại sao “hạn chế sử dụng giá trị rỗng càng nhiều trong quan hệ càng tốt”? Trình bày về vấn đề xuất hiện các bộ dữ liệu sai liên quan đến giá trị rỗng? Làm sao có thể ngăn chặn vấn đề này?
3. Phụ thuộc hàm là gì? Phụ thuộc hàm được xây dựng dựa trên ý nghĩa của lược đồ quan hệ hay dựa trên thể hiện của lược đồ quan hệ? Hãy giải thích lý do cho mỗi lựa chọn.
4. Thẻ nào là một lược đồ phi dạng chuẩn? Hãy cho ví dụ minh họa?
5. Hãy định nghĩa dạng chuẩn 1, dạng chuẩn 2, dạng chuẩn 3, dạng chuẩn BCK. Với mỗi dạng chuẩn, hãy cho một ví dụ minh họa?
6. Lược đồ đạt dạng chuẩn 2 không được chứa loại phụ thuộc hàm nào?
7. Lược đồ đạt dạng chuẩn 3 không được chứa loại phụ thuộc hàm nào?
8. Hãy cho biết sự khác biệt giữa dạng chuẩn 3 và dạng chuẩn BCK? Tại sao BCK được xem là dạng chuẩn mạnh hơn dạng chuẩn 3?

9. Trình bày ý nghĩa của dạng chuẩn trong quá trình thiết kế cơ sở dữ liệu?

Bài tập

1. Hãy xác định dạng chuẩn của từng Q_i, F_i , được cho như sau:
 - a. $Q_1(ABCDEFGH), F_1 = \{A \rightarrow H; AB \rightarrow C; BC \rightarrow D; G \rightarrow B\}$
 - b. $Q_2(ABCSXYZ), F_2 = \{S \rightarrow A; AX \rightarrow B; S \rightarrow B; BY \rightarrow C; CZ \rightarrow X\}$
 - c. $Q_3(ABCD), F_3 = \{A \rightarrow B; BC \rightarrow D; D \rightarrow A\}$
 - d. $Q_4(ABCSXYZ), F_4 = \{S \rightarrow A; AX \rightarrow B; BY \rightarrow C; Y \rightarrow Z; CZ \rightarrow X\}$
 - e. $Q_5(ABCDEG), F_5 = \{AB \rightarrow C; CD \rightarrow E; AG \rightarrow B; B \rightarrow D; A \rightarrow D\}$
 - f. $Q_6(ABCDE), F_6 = \{AC \rightarrow B; E \rightarrow B; BC \rightarrow A; D \rightarrow A; DE \rightarrow C\}$
 - g. $Q_7(ABCDEGHIJ), F_7 = \{BG \rightarrow D; G \rightarrow J; AI \rightarrow C; CE \rightarrow H; BD \rightarrow G; JH \rightarrow A; D \rightarrow I\}$
 - h. $Q_8(ABCDMNOP), F_8 = \{AM \rightarrow N; BN \rightarrow C; AM \rightarrow B; A \rightarrow P; D \rightarrow M; BN \rightarrow M; PC \rightarrow A; DO \rightarrow A\}$
 - i. $Q_9(MNOPRSTU), F_9 = \{M \rightarrow S; MR \rightarrow T; T \rightarrow R; OR \rightarrow T; M \rightarrow U; MT \rightarrow P; NP \rightarrow O; SU \rightarrow R\}$
 - j. $Q_{10}(ABCDEGHIJ), F_{10} = \{BH \rightarrow I; GC \rightarrow A; I \rightarrow J; AE \rightarrow G; D \rightarrow B; I \rightarrow H\}$
 - k. $Q_{11}(ABCDEHI), F_{11} = \{A \rightarrow BC; B \rightarrow DEI; EI \rightarrow AH\}$
 - l. $Q_{12}(ABCDHIGL), F_{12} = \{A \rightarrow BCD; CD \rightarrow HI; IG \rightarrow BL\}$
 - m. $Q_{13}(ABCDGH), F_{13} = \{GH \rightarrow A; AG \rightarrow B; CD \rightarrow G; HG \rightarrow D; BH \rightarrow C; CD \rightarrow H; C \rightarrow A\}$
 - n. $Q_{14}(ABCDGHI), F_{14} = \{G \rightarrow H; AC \rightarrow D; CI \rightarrow G; BC \rightarrow I; B \rightarrow C; A \rightarrow B\}$
 - o. $Q_{15}(MNLPQRS), F_{15} = \{M \rightarrow N; MR \rightarrow N; PN \rightarrow LR; L \rightarrow S; S \rightarrow R\}$
 - p. $Q_{16}(ABCDE), F_{16} = \{DE \rightarrow A; C \rightarrow DE; AD \rightarrow B; BE \rightarrow C\}$

2. Cho danh sách SINHVIÊN gồm có các thuộc tính TênSV, Email1, Email2, Điệnthoại1, Loại ĐT1, Điệnthoại2, LoạiĐT2, ChuyênNgành. Địa chỉ email của sinh viên được ưu tiên sử dụng là địa chỉ email 1, tương tự điện thoại của sinh viên ưu tiên sử dụng là điện thoại 1. Thuộc tính chuyên ngành dùng để lưu trữ chuyên ngành hoặc môn học sinh viên đó đã tham gia học, ví dụ như chuyên ngành kiến trúc sư, hội họa, triết học, ...)
- Giải thích tại sao danh sách SINHVIÊN không đạt dạng chuẩn 1.
 - Hãy thiết kế lại để được bảng SINHVIÊN đạt dạng chuẩn 1 và vẽ lược đồ quan hệ.
3. Cho bảng dữ liệu BÁNHÀNG bên dưới, mỗi cửa hàng sẽ lưu trữ các mặt hàng quan trọng nhất mà cửa hàng có bán:

Vị trí	Mặt hàng
Cửa hàng tạp hóa và thực phẩm	Sữa, trứng, chuối
Cửa hàng văn phòng phẩm	Giấy, bút chì, compa, bút bi
Bưu điện	Tem, bao thư
Cửa hàng máy tính	Đĩa từ, đĩa mềm, đĩa cứng, chuột

- Giải thích tại sao bảng dữ liệu trên không đạt dạng chuẩn 1.
 - Hãy thiết kế lại để được bảng dữ liệu mới đạt dạng chuẩn 1 (có xác định khóa chính cho bảng).
4. Với bảng dữ liệu đã được xây dựng ở câu 3 ở trên:
- Hãy giải thích tại sao bảng dữ liệu không đạt dạng chuẩn 2?
 - Hãy chuẩn hóa quan hệ về dạng chuẩn 2.
5. Xét bảng NHÂNVIÊN sau, với TÊNNV là khóa chính:

TÊNNV	DỰÁN	PHÒNGBAN
Anh	Phân công việc	Phòng thí nghiệm mạng

Chương 4 Phụ thuộc hàm và dạng chuẩn

Bình	Định tuyến mạng	Phòng thí nghiệm mạng
Duy	Thiết kế khảo sát	Phòng quản lý nhân sự
Hạ	Phân công việc	Phòng thí nghiệm mạng
Tâm	Thiết kế khảo sát	Phòng quản lý nhân sự
Minh	Định tuyến mạng	Phòng thí nghiệm mạng
Hùng	Dịch vụ phân tích tin tức	Phòng quản lý nhân sự

- a. Giải thích vì sao NHÂN VIÊN không đạt dạng chuẩn 3?
 - b. Hãy chuẩn hóa NHÂN VIÊN thành dạng chuẩn 3.
6. Một hệ thống thông tin quản lý sinh viên được mô tả như sau:
- Lưu trữ thông tin của sinh viên gồm có tên (TÊNSV), mã số sinh viên (MSSV), số chứng minh nhân dân (SỐCMND), địa chỉ (ĐIACHI), điện thoại (ĐIỆNTHOẠI), ngày sinh (NGÀYSINH), giới tính (GIỚITÍNH), học năm thứ mấy (NĂMHỌC), phòng ban quản lý (MÃPHG), chương trình học (CHƯƠNGTRÌNH). Mỗi sinh viên được xác định thông qua MSSV hoặc SỐCMND.
 - Mỗi phòng ban gồm có tên phòng (TÊNPHG), mã phòng ban (MÃPHG), số phòng (SỐPHG), điện thoại (ĐTPHG), tòa nhà (TÒANHÀ). Mỗi phòng ban được xác định duy nhất thông qua MÃPHG hoặc SỐPHG.
 - Mỗi môn học có tên môn học (TÊNMH), mô tả (MÔTẢ), mã môn (MÃMH), số tín chỉ (SỐTC), mức độ (MÚCĐỘ), phòng ban quản lý (PHGQL). Mỗi môn học được xác định duy nhất thông qua mã môn học.
 - Mỗi học phần mở ra học kỳ (HOCKỲ), năm học (NĂMHỌC) có một mã học phần (MÃHP), liên quan đến một môn học (MÃMH) và do một giáo viên giảng dạy (TÊNGV). Mỗi học phần có một mã phân biệt với các học phần khác trong cùng một khóa học được giảng dạy trong suốt học kỳ của năm. Mã học phần tăng dần

từ 1, 2, 3, ... đến tổng số học phần được dạy trong suốt mỗi học kỳ.

- Mỗi sinh viên (SỐCMND) tham gia học trên mỗi học phần sẽ có một điểm (ĐIỂMTHI) cụ thể.
 - a. Hãy thiết kế một lược đồ cơ sở dữ liệu quan hệ cho hệ thống quản lý sinh viên này.
 - b. Hãy xác định tất cả các phụ hàm nhận diện được.
 - c. Thiết kế lược đồ cơ sở dữ liệu cho ứng dụng này đạt dạng chuẩn 3 hoặc Boyce-Codd. Hãy chỉ ra thuộc tính khóa của mỗi quan hệ. Lưu ý: không đưa vào thêm bất kỳ yêu cầu hay giả định nào trong quá trình thực hiện bài tập này.

Chương 5

NGÔN NGỮ ĐẠI SỐ QUAN HỆ

I. GIỚI THIỆU

Một mô hình dữ liệu, ngoài việc cung cấp những khái niệm dùng để định nghĩa cấu trúc CSDL và các ràng buộc trên dữ liệu, cần phải định nghĩa tập hợp các toán tử để có thể thao tác trên CSDL. Chương 5 trình bày về ngôn ngữ đại số quan hệ bao gồm các phép toán cho phép thao tác trên CSDL quan hệ.

I.1 Khái niệm đại số quan hệ

Ngôn ngữ đại số quan hệ gồm một tập hợp các thao tác cơ bản trên mô hình dữ liệu quan hệ cho phép người dùng thực hiện các yêu cầu tìm kiếm trên CSDL.

Ngôn ngữ đại số quan hệ có vai trò quan trọng vì nó cung cấp nền tảng cho các thao tác trên quan hệ, cho việc xử lý và tối ưu hóa câu truy vấn trong các HQT CSDL quan hệ.

Một yêu cầu thao tác trên dữ liệu được thể hiện bằng một biểu thức đại số quan hệ. Đó là việc thực hiện một hoặc nhiều phép toán đại số quan hệ trên một hoặc nhiều quan hệ và cho ra kết quả là một quan hệ. Quan hệ kết quả này có thể được tiếp tục dùng cho một nhu cầu thao tác khác, cứ như thế tạo thành một chuỗi các toán tử đại số quan hệ.

I.2 Các phép toán đại số quan hệ

Các phép toán trên quan hệ gồm có:

- Phép합 \cup , phép giao \cap , phép hiệu (trừ) – và phép tích Δ các \times . Đây là các phép toán dựa trên lý thuyết tập hợp trong

toán học. Vì một quan hệ được định nghĩa như là một tập hợp các bộ nên ta hoàn toàn có thể áp dụng các phép toán tập hợp này lên quan hệ.

- Phép chọn σ , phép chiếu Π , phép kết \bowtie là các phép toán đặc trưng cho CSDL quan hệ.
- Phép gán \leftarrow và phép đổi tên ρ giúp thuận tiện trong việc thể hiện một biểu thức đại số quan hệ, đặc biệt là các trường hợp thực hiện yêu cầu truy vấn liên quan đến nhiều phép toán quan hệ.
- Phép chia \div và các hàm như tính tổng (SUM), đếm (COUNT), tìm giá trị lớn nhất (MAX), tìm giá trị nhỏ nhất (MIN), tính giá trị trung bình (AVG)... thường được dùng để đáp ứng cho nhu cầu truy vấn thực tế.
- Các phép toán cập nhật dữ liệu: thêm, xóa, sửa.

Các phép toán chỉ cần 1 toán hạng gọi là các phép toán một ngôi (ví dụ phép chọn và phép chiếu). Các phép toán cần 2 toán hạng là các phép toán hai ngôi (ví dụ phép hội, phép giao, phép hiệu, phép kết).

Sau đây là phần trình bày về các phép toán.

II. CÁC PHÉP TOÁN ĐẠI SỐ QUAN HỆ

II.1 Phép hội

Với các phép hội, phép giao và phép hiệu, hai quan hệ tham gia vào phép toán phải thỏa điều kiện là *khả hợp (union compatibility)*. Quan hệ là kết quả của ba phép toán này có cùng tập thuộc tính với quan hệ thứ nhất tham gia vào phép toán.

Cho hai lược đồ quan hệ $R(A_1, A_2, \dots, A_n)$, $S(B_1, B_2, \dots, B_n)$ và hai quan hệ $r(R)$, $s(S)$. Hai quan hệ r và s khả hợp nếu:

- a) Chúng có cùng bậc n (hay cùng số thuộc tính).
- b) Và các thuộc tính tương ứng cùng miền giá trị: $\text{dom}(A_i) = \text{dom}(B_i)$, $1 \leq i \leq n$.

Chương 5 Ngôn ngữ đại số quan hệ

Sau đây là định nghĩa phép hội.

Ký hiệu: \cup

$$r \cup s = \{t \mid (t \in r) \vee (t \in s)\}$$

Kết quả của phép hội r và s khả hợp là một quan hệ chứa các bộ thuộc r hoặc thuộc s . Các bộ trùng nhau (xuất hiện lần thứ 2) sẽ bị loại bỏ khỏi quan hệ kết quả. Xem ví dụ Hình V-1(a).

II.2 Phép giao

Ký hiệu: \cap

$$r \cap s = \{t \mid (t \in r) \wedge (t \in s)\}$$

Kết quả của phép giao r và s khả hợp là một quan hệ chứa các bộ vừa thuộc r vừa thuộc s . Xem ví dụ Hình V-1 (b).

Phép hội và phép giao có tính giao hoán:

$$r \cup s = s \cup r \text{ và } r \cap s = s \cap r$$

Phép hội và phép giao có thể được xem là phép toán n ngôi (tức là chúng có thể áp dụng cho n quan hệ) vì chúng có tính chất kết hợp:

$$r \cup (s \cup t) = (r \cup s) \cup t \text{ và } (r \cap s) \cap t = r \cap (s \cap t)$$

II.3 Phép hiệu

Ký hiệu: -

$$r - s = \{t \mid (t \in r) \wedge (t \notin s)\}$$

Kết quả của phép hiệu r và s khả hợp là một quan hệ gồm các bộ thuộc r mà không thuộc s . Xem ví dụ Hình V-1 (c).

Phép hiệu không có tính giao hoán:

$$r - s \neq s - r$$

r	s	
A	B	C
a1	b1	c1
a2	b2	c2
a3	b3	c3

r ∪ s	r ∩ s	r - s
A	B	C
a1	b1	c1
a2	b2	c2
a3	b3	c3
a4	b4	c4

(a)

(b)

(c)

Hình V-1 (a) Phép합. (b) Phép giao. (c) Phép hiệu.

II.4 Phép tích Đè-các

Phép tích Đè-các không đòi hỏi hai quan hệ tham gia phải khả hợp.

Ký hiệu: \times

Cho $R(A_1, A_2, \dots, A_n)$, $S(B_1, B_2, \dots, B_m)$ và $r(R)$, $s(S)$. Giả sử $r(R)$ có n_R bộ và $s(S)$ có n_S bộ. Tích Đè-các của r và s được định nghĩa như sau:

$$r \times s = q$$

với q là một quan hệ có $n + m$ thuộc tính, có lược đồ là $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ và có $n_R \times n_S$ bộ, mỗi bộ là một sự kết hợp một bộ của r và một bộ của s .

Ví dụ:

Chương 5 Ngôn ngữ đại số quan hệ

r	s	r × s																																																					
<table border="1"><tr><td>A</td><td>B</td><td>C</td></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr><tr><td>a3</td><td>b3</td><td>c3</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	a3	b3	c3	<table border="1"><tr><td>D</td><td>E</td></tr><tr><td>d1</td><td>e1</td></tr><tr><td>d2</td><td>e2</td></tr></table>	D	E	d1	e1	d2	e2	<table border="1"><tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td></tr><tr><td>a1</td><td>b1</td><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>a1</td><td>b1</td><td>c1</td><td>d2</td><td>e2</td></tr><tr><td>a2</td><td>b2</td><td>c2</td><td>d1</td><td>e1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td><td>d2</td><td>e2</td></tr><tr><td>a3</td><td>b3</td><td>c3</td><td>d1</td><td>e1</td></tr><tr><td>a3</td><td>b3</td><td>c3</td><td>d2</td><td>e2</td></tr></table>	A	B	C	D	E	a1	b1	c1	d1	e1	a1	b1	c1	d2	e2	a2	b2	c2	d1	e1	a2	b2	c2	d2	e2	a3	b3	c3	d1	e1	a3	b3	c3	d2	e2
A	B	C																																																					
a1	b1	c1																																																					
a2	b2	c2																																																					
a3	b3	c3																																																					
D	E																																																						
d1	e1																																																						
d2	e2																																																						
A	B	C	D	E																																																			
a1	b1	c1	d1	e1																																																			
a1	b1	c1	d2	e2																																																			
a2	b2	c2	d1	e1																																																			
a2	b2	c2	d2	e2																																																			
a3	b3	c3	d1	e1																																																			
a3	b3	c3	d2	e2																																																			

Hình V-2 Phép tích Đề - các.

Thông thường, không phải bất kỳ dòng nào trong kết quả của phép tích Đề-các đều có ý nghĩa thực tế. Do vậy, người ta thực hiện một phép chọn từ kết quả này để cho ra các bộ dữ liệu có nghĩa. Việc thực hiện phép tích Đề-các trước và sau đó thực hiện phép chọn tương đương với một phép toán đơn, đó là phép kết (sẽ được trình bày ở phần II.9).

II.5 Phép chọn

Phép chọn được dùng để chọn ra từ một quan hệ các bộ thỏa mãn điều kiện chọn.

Ký hiệu: σ

Cú pháp: $\sigma_{<\text{điều kiện chọn}>} (r)$

Trong đó $<\text{điều kiện chọn}>$ là một biểu thức luận lý được tạo thành dựa trên các thuộc tính của r. Một cách tổng quát, r là một biểu thức đại số quan hệ (có kết quả là một quan hệ). Trường hợp đơn giản nhất, r là tên của một quan hệ trong CSDL.

Cụ thể hơn, $<\text{điều kiện chọn}>$ gồm một hoặc nhiều mệnh đề có dạng:

$<\text{tên thuộc tính}> <\text{toán tử so sánh}> <\text{giá trị bằng}>$

hoặc

$<\text{tên thuộc tính}> <\text{toán tử so sánh}> <\text{tên thuộc tính}>$

Trong đó, \langle tên thuộc tính \rangle là tên của một thuộc tính của r, \langle toán tử so sánh \rangle là một trong các phép so sánh $\{=, <, \leq, >, \geq, \neq\}$, \langle giá trị hằng \rangle là một giá trị không đổi thuộc miền giá trị của thuộc tính tương ứng. Các mệnh đề trong \langle điều kiện chọn \rangle được liên kết với nhau bởi các toán tử logic như AND (\wedge), OR (\vee) hoặc NOT (\neg).

Kết quả của phép chọn trên quan hệ r là một quan hệ có cùng tập thuộc tính như quan hệ r và có số bộ ít hơn hoặc bằng số bộ của quan hệ r.

Ví dụ: Cho quan hệ KHOA (MÃKHOA, TÊNKHOA, NĂMTL, TRƯỜNGKHOA). Cho biết các khoa được thành lập từ năm 1980 trở về sau.

$$\sigma_{NAMTL \geq 1980} (KHOA)$$

Xem ví dụ minh họa Hình V-3 (a).

Ta có thể xem phép chọn như là một bộ lọc chỉ giữ lại những bộ dữ liệu thỏa mãn điều kiện chọn.

Phép chọn có tính giao hoán, nghĩa là:

$$\sigma_{\langle \text{điều kiện 1} \rangle} (\sigma_{\langle \text{điều kiện 2} \rangle} (r)) = \sigma_{\langle \text{điều kiện 2} \rangle} (\sigma_{\langle \text{điều kiện 1} \rangle} (r))$$

II.6 Phép chiếu

Phép chiếu được dùng để giữ lại một số thuộc tính của một quan hệ ban đầu và bỏ qua những thuộc tính khác không quan tâm.

Ký hiệu: Π

Cú pháp: $\Pi_{\langle \text{danh sách thuộc tính} \rangle} (r)$

Trong đó \langle danh sách thuộc tính \rangle là các thuộc tính cần giữ lại (trong các thuộc tính của r). r là một biểu thức đại số quan hệ (có kết quả là một quan hệ). Trường hợp đơn giản nhất, r là một quan hệ trong CSDL.

Kết quả của phép chiếu là một quan hệ có các thuộc tính được liệt kê trong \langle danh sách thuộc tính \rangle , thứ tự của các thuộc tính giống như thứ tự của chúng trong \langle danh sách thuộc tính \rangle .

Chương 5 Ngôn ngữ đại số quan hệ

Nếu **<đanh sách thuộc tính>** gồm các thuộc tính không phải là khóa của r thì hoàn toàn có thể xảy ra tình trạng các bộ dữ liệu trùng nhau. Phép chiếu hủy các bộ trùng và kết quả của phép chiếu là một quan hệ. Tóm lại, số bộ của quan hệ kết quả của phép chiếu nhỏ hơn hoặc bằng số bộ của quan hệ r ban đầu. Trường hợp **<đanh sách thuộc tính>** là một siêu khóa của r , nghĩa là có chứa khóa của r , thì quan hệ kết quả có số bộ bằng số bộ của r .

Ví dụ: Cho biết mã và tên của tất cả các khoa.

$\Pi_{MÃKHOA, TÊNKHOA}(KHOA)$

Phép toán này được minh họa ở Hình V- 3 (b).

MÃKHOA	TÊNKHOA	NĂMTL	PHÒNG	ĐIỆNTHOAI	TRƯỜNGKHOA	NGÀYNHẬMCHỨC
CNTT	Công nghệ thông tin	1995	B11	838123456	2	20/02/2005
SH	Sinh học	1980	B31	838454545	4	11/10/2000
HH	Hóa học	1980	B41	838456456	7	15/10/2001

(a)

MÃKHOA	TÊNKHOA
CNTT	Công nghệ thông tin
VL	Vật lý
SH	Sinh học
HH	Hóa học

(b)

Hình V- 3 Kết quả phép chọn và phép chiếu. (a) $\sigma_{NÄMTL \geq 1980}(KHOA)$. (b) $\Pi_{MÃKHOA, TÊNKHOA}(KHOA)$.

II.7 Phép gán

Với một yêu cầu truy xuất dữ liệu, biểu thức đại số quan hệ tương ứng có thể đơn giản chỉ sử dụng một phép toán, cũng có thể là việc thực hiện một chuỗi các phép toán, kết quả của phép toán trước là toán hạng cho phép toán sau. Trong trường hợp biểu thức đại số quan hệ phức tạp liên quan đến nhiều phép toán, ta hoàn toàn có thể làm đơn giản chúng bằng cách cho thực hiện từng phép toán riêng biệt và gán kết quả cho một quan hệ trung gian. Toán tử thực hiện sau sẽ sử dụng kết quả này như là một toán hạng.

Ký hiệu: \leftarrow

Cú pháp: $r \leftarrow E$

Khi đó kết quả của biểu thức đại số quan hệ E sẽ được gán cho quan hệ r. Sau đó có thể dùng r như là toán hạng cho một phép toán đại số quan hệ khác.

Ví dụ: Cho biết mã và tên của các khoa được thành lập sau năm 1980.

$KHOA_1 \leftarrow \sigma_{NAMTL > 1980} (KHOA)$

$KETQUA \leftarrow \Pi_{MAKHOA, TENKHOA} (KHOA_1)$

II.8 Phép đổi tên

Ta có thể dùng phép đổi tên để đổi tên của quan hệ hoặc tên của thuộc tính trong quan hệ hoặc cả hai.

Ký hiệu: ρ

Phép đổi tên tổng quát áp dụng cho quan hệ R bậc n được viết như sau:

- $\rho_{s(B_1, B_2, \dots, B_n)}(r)$ dùng để đổi tên quan hệ lẩn tên thuộc tính hoặc
- $\rho_s(r)$ dùng để đổi tên quan hệ hoặc
- $\rho_{(B_1, B_2, \dots, B_n)}(r)$ dùng để đổi tên thuộc tính.

Trong đó s là tên mới của quan hệ; B_1, B_2, \dots, B_n là tên mới của các thuộc tính. Nếu quan hệ r ban đầu có tên các thuộc tính là $(A_1, A_2, \dots,$

Chương 5 Ngôn ngữ đại số quan hệ

A_n) thì theo biểu thức đổi tên thuộc tính bên trên, A_i sẽ được đổi tên thành B_i .

Ví dụ: Cho quan hệ ĐỀTÀI(MÃĐT, TÊNĐT, CÁPQL, GVCNĐT)
Biểu thức sau:

$\rho(MAĐETÀI, TÊNĐETÀI, CÁPQUÀNLÝ, NGƯỜICHÙNHIỆM) (ĐỀTÀI)$

đổi tên các thuộc tính của quan hệ ĐỀTÀI tương ứng thành
MÃĐETÀI, TÊNĐETÀI, CÁPQUÀNLÝ và NGƯỜICHÙNHIỆM.

II.9 Phép kết

Phép kết có vai trò rất quan trọng trong CSDL quan hệ. Phép kết cho phép ta truy xuất dữ liệu trên nhiều quan hệ.

Ký hiệu: \bowtie

Cú pháp: $r \bowtie_{<\text{điều kiện kết}>} s$

Phép kết được dùng để kết hợp các bộ có liên quan từ hai quan hệ tham gia vào phép kết. Mỗi liên quan này được thể hiện trong $<\text{điều kiện kết}>$.

Cho $R(A_1, A_2, \dots, A_n)$, $S(B_1, B_2, \dots, B_m)$ và $r(R)$, $s(S)$. Giả sử $r(R)$ có n_R bộ và S có n_S bộ, kết quả của phép kết là một quan hệ q có $m + n$ thuộc tính, có lược đồ quan hệ là $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ với các thuộc tính xuất hiện theo đúng thứ tự này. Với mỗi bộ r_0 thuộc r , nếu có một bộ s_0 thuộc s thỏa mãn $<\text{điều kiện kết}>$ thì sẽ sinh ra một bộ trong quan hệ kết quả q . Đây là điểm khác biệt chủ yếu giữa phép kết và phép tích Đề-các.

$<\text{Điều kiện kết}>$ liên quan đến các thuộc tính của r và s , có dạng:

$<\text{điều kiện}_1> \wedge <\text{điều kiện}_2> \wedge \dots \wedge <\text{điều kiện}_n>$

Trong đó mỗi $<\text{điều kiện}>$ có dạng $A_i \theta B_j$ với A_i là thuộc tính của r và B_j là thuộc tính của s ; A_i và B_j có cùng miền giá trị, θ là một trong các toán tử so sánh $\{=, <, \leq, >, \geq, \neq\}$. Phép kết với điều kiện kết liên quan đến phép so sánh θ tổng quát được gọi là Theta-kết.

Các bộ có giá trị tại thuộc tính tham gia vào điều kiện kết là rỗng thì không xuất hiện trong quan hệ kết quả.

Kết bằng (equijoin)

Phép kết được sử dụng phổ biến nhất trong CSDL quan hệ chỉ dùng phép so sánh bằng trong điều kiện kết, gọi là phép kết bằng.

Ví dụ: Cho hai quan hệ ĐỀTÀI_1 và CHỦĐỀ, xem Hình V-5 (a). Cho biết thông tin về đề tài và tên chủ đề của từng đề tài.

ĐỀTÀI_1 $\bowtie_{MÁCĐ = MÁCD} CHỦĐỀ$

Kết tự nhiên (natural join)

Kết quả của phép kết bằng luôn xuất hiện ít nhất một cặp thuộc tính có giá trị trùng nhau trong từng bộ. Một trong hai thuộc tính trong một/nhiều cặp thuộc tính này là thừa.

Phép kết tự nhiên, ký hiệu là *, bỏ đi thuộc tính thứ 2 trong từng cặp thuộc tính có giá trị trùng nhau này. Định nghĩa chuẩn của phép kết tự nhiên yêu cầu từng cặp thuộc tính kết này phải trùng tên. Trong trường hợp các cặp thuộc tính kết không trùng tên và muốn sử dụng phép kết tự nhiên thì phải thực hiện phép toán đổi tên trước.

Hình V-5 (b) minh họa phép kết tự nhiên giữa hai quan hệ ĐỀTÀI_1 và CHỦĐỀ.

Ví dụ: Cho biết họ tên các giáo viên thuộc bộ môn có tên là 'Hệ thống thông tin'.

Vì hai thuộc tính liên quan đến điều kiện kết của quan hệ GIÁOVIÊN và BỘMÔN là cùng tên nên ta có thể vận dụng phép kết tự nhiên như sau:

$\Pi_{HOTEN}(\sigma_{TENBM = 'Hệ thống thông tin'} (GIÁOVIÊN * BỘMÔN))$

Kết ngoài (Outer join)

Là mở rộng của phép kết thông thường nhằm đáp ứng cho một số kiểu nhu cầu truy vấn. Trong phép kết thông thường, chỉ những bộ trong hai quan hệ tham gia vào phép kết thỏa mãn điều kiện kết thì mới xuất hiện trong quan hệ kết quả, còn được gọi là phép kết trong (inner

join). Vì vậy, những bộ không thỏa mãn điều kiện kết sẽ bị loại bỏ. Các bộ có giá trị bằng rỗng trên điều kiện kết cũng bị loại bỏ. Giả sử ta đặt ra một yêu cầu đối với phép kết là tạo ra thống kê về tất cả thông tin của hai quan hệ tham gia vào phép kết. Khi đó phép kết trong không đạt được yêu cầu này mà còn làm mất thông tin.

Phép kết ngoài được sử dụng khi ta muốn giữ lại ở quan hệ kết quả tất cả các bộ của quan hệ bên trái (*left outer join*), hoặc tất cả các bộ của quan hệ bên phải (*right outer join*), hoặc của cả hai quan hệ (*outer join*) tham gia vào phép kết cho dù các bộ này không có bộ nào trong quan hệ còn lại thỏa mãn điều kiện kết.

r	s	Equijoin																																		
<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="width: 50%;">A</th> <th style="width: 50%;">B</th> </tr> </thead> <tbody> <tr><td>1</td><td>x</td></tr> <tr><td>2</td><td>y</td></tr> <tr><td>3</td><td>z</td></tr> </tbody> </table>	A	B	1	x	2	y	3	z	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="width: 50%;">C</th> <th style="width: 50%;">D</th> </tr> </thead> <tbody> <tr><td>1</td><td>a</td></tr> <tr><td>1</td><td>b</td></tr> <tr><td>2</td><td>c</td></tr> <tr><td>4</td><td>d</td></tr> </tbody> </table>	C	D	1	a	1	b	2	c	4	d	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="width: 25%;">A</th> <th style="width: 25%;">B</th> <th style="width: 25%;">C</th> <th style="width: 25%;">D</th> </tr> </thead> <tbody> <tr><td>1</td><td>x</td><td>1</td><td>a</td></tr> <tr><td>1</td><td>x</td><td>1</td><td>b</td></tr> <tr><td>2</td><td>y</td><td>2</td><td>c</td></tr> </tbody> </table> <p style="text-align: center;">(a)</p>	A	B	C	D	1	x	1	a	1	x	1	b	2	y	2	c
A	B																																			
1	x																																			
2	y																																			
3	z																																			
C	D																																			
1	a																																			
1	b																																			
2	c																																			
4	d																																			
A	B	C	D																																	
1	x	1	a																																	
1	x	1	b																																	
2	y	2	c																																	

Left outer join	Right outer join	Outer join																																																																
<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="width: 25%;">A</th> <th style="width: 25%;">B</th> <th style="width: 25%;">C</th> <th style="width: 25%;">D</th> </tr> </thead> <tbody> <tr><td>1</td><td>x</td><td>1</td><td>a</td></tr> <tr><td>1</td><td>x</td><td>1</td><td>b</td></tr> <tr><td>2</td><td>y</td><td>2</td><td>c</td></tr> <tr><td>3</td><td>z</td><td>null</td><td>null</td></tr> </tbody> </table> <p style="text-align: center;">(b)</p>	A	B	C	D	1	x	1	a	1	x	1	b	2	y	2	c	3	z	null	null	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="width: 25%;">A</th> <th style="width: 25%;">B</th> <th style="width: 25%;">C</th> <th style="width: 25%;">D</th> </tr> </thead> <tbody> <tr><td>1</td><td>x</td><td>1</td><td>a</td></tr> <tr><td>1</td><td>x</td><td>1</td><td>b</td></tr> <tr><td>2</td><td>y</td><td>2</td><td>c</td></tr> <tr><td>null</td><td>null</td><td>4</td><td>d</td></tr> </tbody> </table> <p style="text-align: center;">(c)</p>	A	B	C	D	1	x	1	a	1	x	1	b	2	y	2	c	null	null	4	d	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="width: 25%;">A</th> <th style="width: 25%;">B</th> <th style="width: 25%;">C</th> <th style="width: 25%;">D</th> </tr> </thead> <tbody> <tr><td>1</td><td>x</td><td>1</td><td>a</td></tr> <tr><td>1</td><td>x</td><td>1</td><td>b</td></tr> <tr><td>2</td><td>y</td><td>2</td><td>c</td></tr> <tr><td>3</td><td>z</td><td>null</td><td>null</td></tr> <tr><td>null</td><td>null</td><td>4</td><td>d</td></tr> </tbody> </table> <p style="text-align: center;">(d)</p>	A	B	C	D	1	x	1	a	1	x	1	b	2	y	2	c	3	z	null	null	null	null	4	d
A	B	C	D																																																															
1	x	1	a																																																															
1	x	1	b																																																															
2	y	2	c																																																															
3	z	null	null																																																															
A	B	C	D																																																															
1	x	1	a																																																															
1	x	1	b																																																															
2	y	2	c																																																															
null	null	4	d																																																															
A	B	C	D																																																															
1	x	1	a																																																															
1	x	1	b																																																															
2	y	2	c																																																															
3	z	null	null																																																															
null	null	4	d																																																															

Hình V-4 Kết quả các phép kết r và s trên điều kiện kết là A = C. (a) Kết bằng (Equijoin). (b) Kết trái (Left outer join). (c) Kết phải (Right outer join). (d) Kết ngoài (Outer join).

ĐỀ TÀI_1

MÃ ĐT	TÊN ĐT	CÁPQL	MÃ CĐ
1	HTTT quản lý các trường ĐH	ĐHQG	QLGD
2	HTTT quản lý giáo vụ cho một Khoa	Trường	QLGD
3	Nghiên cứu chế tạo sợi Nanô Platin	ĐHQG	NCPT
4	Tạo vật liệu sinh học bằng màng ối người	Nhà nước	NCPT
5	Ứng dụng hóa học xanh	Trường	UDCN
6	Nghiên cứu tế bào gốc	Nhà nước	NCPT
7	HTTT quản lý thư viện ở các trường ĐH	Trường	QLGD

CHỦ ĐỀ

MÃ CĐ	TÊN CĐ
NCPT	Nghiên cứu phát triển
QLGD	Quản lý giáo dục
UDCN	Ứng dụng công nghệ

MÃ ĐT	TÊN ĐT	CÁPQL	MÃ CĐ	MÃ CĐ	TÊN CĐ
1	HTTT quản lý các trường ĐH	ĐHQG	QLGD	QLGD	Quản lý giáo dục
2	HTTT quản lý giáo vụ cho một Khoa	Trường	QLGD	QLGD	Quản lý giáo dục
3	Nghiên cứu chế tạo sợi Nanô Platin	ĐHQG	NCPT	NCPT	Nghiên cứu phát triển
4	Tạo vật liệu sinh học bằng màng ối người	Nhà nước	NCPT	NCPT	Nghiên cứu phát triển
5	Ứng dụng hóa học xanh	Trường	UDCN	UDCN	Ứng dụng công nghệ
6	Nghiên cứu tế bào gốc	Nhà nước	NCPT	NCPT	Nghiên cứu phát triển
7	HTTT quản lý thư viện ở các trường ĐH	Trường	QLGD	QLGD	Quản lý giáo dục

(a)

MÃ ĐT	TÊN ĐT	CÁPQL	MÃ CĐ	TÊN CĐ
1	HTTT quản lý các trường ĐH	ĐHQG	QLGD	Quản lý giáo dục
2	HTTT quản lý giáo vụ cho một Khoa	Trường	QLGD	Quản lý giáo dục
3	Nghiên cứu chế tạo sợi Nanô Platin	ĐHQG	NCPT	Nghiên cứu phát triển
4	Tạo vật liệu sinh học bằng màng ối người	Nhà nước	NCPT	Nghiên cứu phát triển
5	Ứng dụng hóa học xanh	Trường	UDCN	Ứng dụng công nghệ
6	Nghiên cứu tế bào gốc	Nhà nước	NCPT	Nghiên cứu phát triển
7	HTTT quản lý thư viện ở các trường ĐH	Trường	QLGD	Quản lý giáo dục

(b)

Hình V-5 Kết bằng và kết tự nhiên. (a) $\text{ĐỀ TÀI}_1 \bowtie_{\text{MÃ CĐ} = \text{MÃ CĐ}} \text{CHỦ ĐỀ}$. (b) $\text{ĐỀ TÀI}_1 * \text{CHỦ ĐỀ}$.

II.10 Phép chia

Ký hiệu: \div

Cho lược đồ quan hệ R có tập thuộc tính là Z , $R(Z)$ và quan hệ $r(R)$.

Cho lược đồ quan hệ S có tập thuộc tính là X , $S(X)$ và quan hệ $s(S)$.

Để áp dụng được phép chia trên hai quan hệ r và s thì $X \subseteq Z$. Đặt $Y = Z - X$, nghĩa là $Z = X \cup Y$. Y là tập thuộc tính của r mà không phải là tập thuộc tính của s . Phép chia được định nghĩa như sau:

$$r \div s = q$$

với q là quan hệ có tập thuộc tính là Y và gồm các bộ t nếu các bộ $r_0 \in r$ và $r_0[Y] = t$ thì $r_0[X] = s_0$ đối với mọi $s_0 \in s$. Nghĩa là, với một bộ t là kết quả của phép chia thì giá trị của t xuất hiện trong r dưới dạng kết hợp của t và mọi bộ của s . Xem ví dụ Hình V-6.

Ví dụ: $q = r \div s$

r(Z)	s(X)	q(Y)
A	A	B
a1	b1	b1
a2	b1	
a3	b1	
a4	b1	
a1	b2	
a3	b2	
a2	b3	
a3	b3	
a4	b3	
a1	b4	
a2	b4	
a3	b4	b4

$X = \{A\}$, $Y = \{B\}$, $Z = \{A, B\}$.
Hai bộ $b1$ và $b4$ là kết quả của
phép chia. Mỗi bộ này xuất hiện
trong r ở dạng kết hợp với tất cả
các bộ của s .

Hình V-6 Phép chia.

II.11 Tập hợp đầy đủ

Tập hợp các phép toán đại số quan hệ $\{\sigma, \Pi, \cup, -, \times\}$ là một tập hợp đầy đủ vì bất cứ phép toán đại số quan hệ nào khác cũng đều có thể được biểu diễn lại bằng một chuỗi các phép toán trong tập hợp này.

Ví dụ: $r \cap s \equiv (r \cup s) - ((r - s) \cup (s - r))$

$r \bowtie_{\text{điều kiện}} s \equiv \sigma_{\text{điều kiện}} (r \times s)$

Phép kết tự nhiên tương đương với việc thực hiện chuỗi các phép toán theo thứ tự: phép đổi tên, tích Đề-các, phép chọn và phép chiếu.

Tuy vậy, mỗi phép toán đại số quan hệ đều giữ một vai trò quan trọng trong các ứng dụng CSDL nên cần được xem như là một phép toán độc lập. Và ta nên đặt yêu cầu về sự tiện lợi khi thao tác trên CSDL dùng các phép toán này hơn là đặt ra vấn đề rằng có cần thiết chúng hay không.

III. CÁC HÀM TÍNH TỔNG HỢP VÀ GOM NHÓM

Một số yêu cầu truy vấn không thể thực hiện được với các phép toán đại số quan hệ cơ bản, chẳng hạn như yêu cầu tính lương trung bình của các giáo viên trong một khoa. Vì vậy, một số hàm tính tổng hợp (*aggregate function*) được định nghĩa.

Người ta thường có nhu cầu tính toán liên quan đến các hàm sau: SUM, AVG, MAX, MIN và COUNT.

Ngoài ra, việc tính toán trong thực tế không chỉ áp dụng trên tất cả các dòng của một quan hệ mà còn có thể áp dụng trên một số dòng của quan hệ sau khi gom nhóm theo từng giá trị của một thuộc tính nào đó. Ví dụ, khi muốn biết số nhân viên của từng bộ môn, ta sẽ gom nhóm các dòng dữ liệu dựa trên thuộc tính MÃBM, những dòng nào có cùng giá trị MÃBM sẽ thuộc cùng một nhóm. Sau đó, ta thực hiện đếm trên từng nhóm sẽ cho ra kết quả mong muốn.

Hàm tính tổng hợp và gom nhóm được định nghĩa như sau:

Ký hiệu: Σ

Cú pháp: $\langle \text{danh sách thuộc tính gom nhóm} \rangle \mathfrak{J} \langle \text{danh sách hàm} \rangle (r)$

Trong đó, $\langle \text{danh sách thuộc tính gom nhóm} \rangle$ là các thuộc tính của r mà dữ liệu sẽ được gom nhóm trên đó để phục vụ cho mục đích tính toán. $\langle \text{danh sách hàm} \rangle$ là những cặp (hàm, thuộc tính) với ‘hàm’ là một trong các hàm được đề cập ở trên và ‘thuộc tính’ là một trong các thuộc tính của r . Kết quả cho ra là một quan hệ có số thuộc tính là số thuộc tính gom nhóm, mỗi hàm tính tổng hợp cho ra thêm một thuộc tính nữa.

Ví dụ: Cho biết số giáo viên trong từng bộ môn.

$\rho_{(M\ddot{A}BM, S\ddot{O}GV)}(M\ddot{A}BM \mathfrak{J}_{COUNT(M\ddot{A}GV)}(GI\ddot{A}OVI\ddot{E}N))$

Xem ví dụ minh họa Hình V-7.

MÃBM	SỐGV
HPT	3
HTTT	2
MMT	2
VLĐT	1
VS	2

(a)

SỐGV
10

(b)

Hình V-7 Hàm tính tổng hợp.

(a) $\rho_{(M\ddot{A}BM, S\ddot{O}GV)}(M\ddot{A}BM \mathfrak{J}_{COUNT(M\ddot{A}GV)}(GI\ddot{A}OVI\ddot{E}N))$.

b) $\rho_{(S\ddot{O}GV)}(\mathfrak{J}_{COUNT(M\ddot{A}GV)}(GI\ddot{A}OVI\ddot{E}N))$.

IV. PHÉP TOÁN CẬP NHẬT DỮ LIỆU

IV.1 Thêm

Cú pháp:

$$r \leftarrow r \cup E$$

Chương 5 Ngôn ngữ đại số quan hệ

trong đó, r là một quan hệ và E là một biểu thức đại số quan hệ cho biết một/ nhiều bộ dữ liệu muốn chèn vào r .

Ví dụ: Thêm vào KHOA (MÃKHOA, TÊNKHOA, NĂMTL, TRƯỞNGKHOA) một bộ mới <'MT', 'Môi trường', 1998, Null>.

$KHOA \leftarrow KHOA \cup \{('MT', 'Môi trường', 1998, Null)\}$

IV.2 Xóa

Cú pháp:

$r \leftarrow r - E$

Trong đó, r là một quan hệ và E là một biểu thức đại số quan hệ cho biết một/ nhiều bộ dữ liệu muốn chèn vào r .

Ví dụ: Xóa tất cả những dòng dữ liệu liên quan đến giáo viên mã số là '003' trong quan hệ THAMGIAĐT.

$THAMGIAĐT \leftarrow THAMGIAĐT - \sigma_{MÃGV='003'} (THAMGIAĐT)$

IV.3 Sửa

Cú pháp:

$r \leftarrow \prod_{F1, F2, \dots, Fn} (r)$

Trong đó, F_i là thuộc tính thứ i của quan hệ r , hoặc là biểu thức cho biết giá trị mới của thuộc tính thứ i .

Ví dụ: Cho GIÁOVIÊN (MÃGV, HỌTÊN, LUONG, MÃBM) và BỘMÔN (MÃBM, TÊNBM). Tăng lương của các giáo viên thuộc bộ môn 'Hệ thống thông tin' lên 1.5 lần.

$GIÁOVIÊN \leftarrow \prod_{MÃGV, HỌTÊN, LUONG*1.5, MÃBM} (GIÁOVIÊN * (\sigma_{TÊNBM} = 'Hệ thống thông tin' (BỘMÔN)))$

V. KẾT LUẬN

Chương 5 trình bày về các phép toán đại số quan hệ. Các phép toán quan hệ một ngôi gồm có phép chọn, phép chiếu, phép đổi tên. Các phép toán quan hệ hai ngôi gồm có phép hội, phép giao, phép hiệu và phép tích Đề-các. Với phép hội, phép giao và phép hiệu, hai quan hệ

là toán hạng phải khả hợp. Phép kết trong đương với việc thực hiện phép tích Đè-các và sau đó thực hiện phép chọn. Có nhiều loại phép kết khác nhau: kết bằng, kết tự nhiên, kết trái, kết phải, kết ngoài. Hàm tính tổng hợp và gom nhóm cũng được định nghĩa nhằm phục vụ cho nhiều yêu cầu thực tế. Cuối cùng là phần trình bày các thao tác thêm, xóa, sửa trên CSDL.

Chương tiếp theo sẽ trình bày về ngôn ngữ truy vấn dữ liệu được dùng trong các HQT CSDL quan hệ.

Câu hỏi ôn tập

1. Hãy liệt kê các phép toán đại số quan hệ, cho biết ký hiệu và vai trò của từng phép toán.
2. Hãy phát biểu một cách tổng quát bằng ngôn ngữ tự nhiên về từng phép toán đại số quan hệ, điều kiện thực hiện phép toán và kết quả của chúng. Ví dụ: “Phép chọn từ quan hệ r với điều kiện chọn C cho ra kết quả là một quan hệ có các thuộc tính là các thuộc tính của quan hệ r ban đầu, số bộ là số bộ của quan hệ r ban đầu thỏa mãn điều kiện C ”.
3. Thế nào là hai lược đồ quan hệ khả hợp? Các phép toán đại số quan hệ nào đòi hỏi các quan hệ tham gia vào phép toán phải khả hợp? Kết quả của chúng có đặc điểm chung gì?
4. Khi nào thì ta thực hiện kết bằng? Khi nào thì ta có thể thực hiện phép kết tự nhiên thay vì thực hiện kết bằng? Kết quả phép kết bằng và phép kết tự nhiên giống nhau và khác nhau như thế nào?
5. Cho biết vai trò của phép đổi tên? Cho ví dụ minh họa. Có nhu cầu truy vấn nào mà nếu không dùng phép đổi tên thì sẽ không thể hiện được bằng đại số quan hệ hay không? Nếu có, hãy cho ví dụ minh họa.
6. Vai trò của phép gán là gì? Cho ví dụ minh họa.
7. Vai trò của phép kết ngoài (outer join)? Cho biết sự khác nhau của phép kết trong (inner join) và phép kết ngoài?
8. Thông thường, kết quả của phép tích Đè-các không phải luôn luôn có ý nghĩa thực tế. Hãy cho một ví dụ cho biết rằng vẫn có trường hợp tất cả các dòng trong kết quả của phép tích Đè-các là có ý nghĩa thực tế.

Chương 5 Ngôn ngữ đại số quan hệ

9. Cho hai bảng T1 và T2 như sau. Hãy cho biết kết quả của những phép toán sau :

P	Q	R
10	a	5
15	b	8
25	a	6

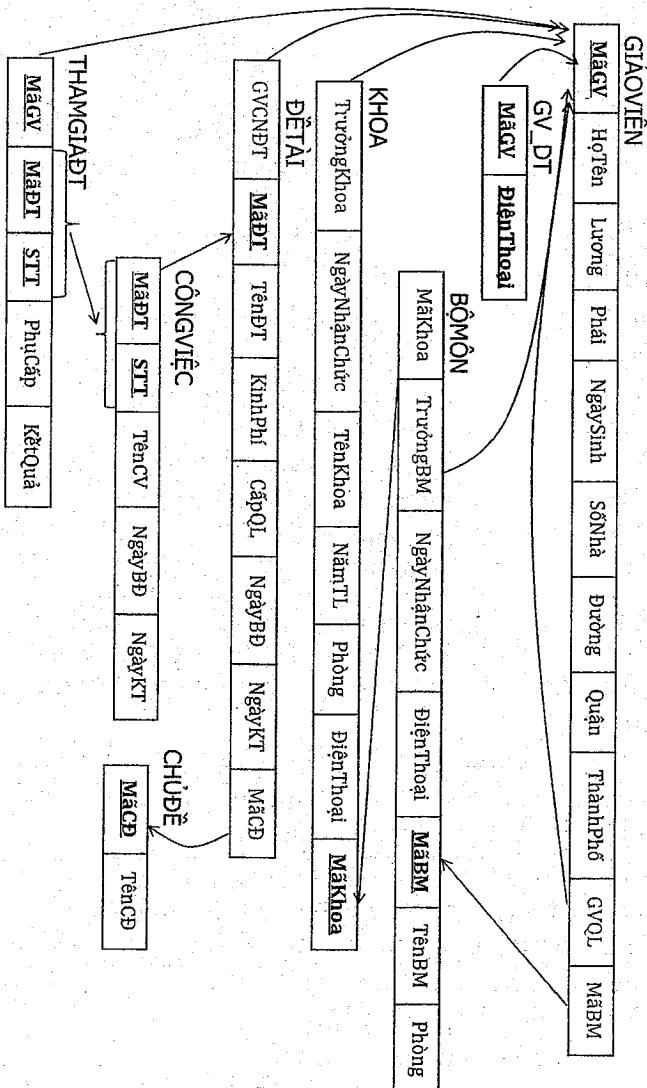
A	B	C
10	b	6
25	c	3
10	b	5

- a. T1 kết trong (inner join) T2 với điều kiện kết là $T1.P = T2.A$
- b. T1 kết trong T2 với điều kiện kết là $T1.Q = T2.B$
- c. T1 kết trái (left outer join) T2 với điều kiện kết là $T1.P = T2.A$
- d. T1 kết phải (right outer join) T2 với điều kiện kết là $T1.Q = T2.B$
- e. $T1 \cup T2$
- f. T1 kết trong (inner join) T2 với điều kiện kết là $(T1.P = T2.A \text{ AND } T1.R = T2.C)$

Bài tập

1. Cho lược đồ CSDL Quản lý Đề tài nghiên cứu khoa học như hình bên dưới, hãy thực hiện những nhu cầu truy vấn sau dùng ngôn ngữ đại số quan hệ:

Giáo trình CSDL
Khoa Công nghệ thông tin, Trường ĐHKHTN Tp.HCM



- a. Cho biết danh sách giáo viên gồm mã, họ tên, phái, ngày sinh.
- b. Cho danh sách đề tài gồm mã đề tài, tên đề tài, kinh phí.
- c. Cho biết danh sách giáo viên có lương lớn hơn 2000.
- d. Cho biết danh sách giáo viên thuộc bộ môn có mã là HTTT.
- e. Cho biết danh sách giáo viên thuộc bộ môn mã là HTTT có lương lớn hơn 2000.
- f. Cho biết những bộ môn chưa biết người làm trưởng bộ môn.
- g. Cho biết những bộ môn đã phân công giáo viên làm trưởng bộ môn.
- h. Cho biết danh sách gồm mã, họ tên, phái, ngày sinh của các giáo viên có lương lớn hơn 2000.
- i. Cho biết danh sách gồm mã các giáo viên có tham gia đề tài mã số 001 hoặc đề tài có mã là 002.
- j. Cho biết danh sách gồm mã các giáo viên vừa có tham gia đề tài mã số 001 vừa có tham gia đề tài mã số 002.
- k. Cho biết danh sách gồm mã các giáo viên có tham gia đề tài mã số 001 nhưng không có tham gia đề tài mã số 002.
- l. Cho biết các thể hiện cho biết tất cả các giáo viên thuộc bộ môn HTTT tham gia tất cả các công việc của các đề tài cấp trường. Danh sách kết xuất gồm mã giáo viên, mã đề tài, số thứ tự.
- m. Liệt kê danh sách các thể hiện cho biết các giáo viên thuộc bộ môn mã là MMT tham gia tất cả các công việc liên quan đến đề tài mã là 001.
- n. Liệt kê danh sách các thể hiện cho biết các giáo viên thuộc bộ môn tên là 'Mạng máy tính' tham gia tất cả các công việc liên quan đến đề tài tên là 'Ứng dụng hóa học xanh'.
- o. Liệt kê danh sách các thể hiện cho biết giáo viên mã là 003 tham gia tất cả các công việc liên quan đến đề tài mã là 001.
- p. Cho biết danh sách giáo viên và mã, tên bộ môn mà giáo viên trực thuộc. Danh sách kết xuất gồm MÃGV, HỌTÊN, PHÁI, NGÀYSINH, MÃBM, TÊNBM.
- q. Cho biết danh sách các trưởng khoa.
- r. Cho biết danh sách các bộ môn và tên của người làm trưởng bộ môn.
- s. Cho biết danh sách gồm thông tin các bộ môn và tên của người làm trưởng bộ môn, đối với những bộ môn chưa biết giáo viên

nào làm trưởng bộ môn thì tại các cột cho biết mã và tên của trưởng bộ môn mang giá trị rỗng (null).

- t. Cho biết danh sách gồm thông tin giáo viên và đề tài mà giáo viên đã tham gia, những giáo viên nào chưa có tham gia đề tài thì tại các cột cho biết thông tin đề tài hiện giá trị rỗng. Danh sách kết xuất gồm MÃGV, HỌTÊN, MÃĐT, STT, TÊNCV, TÊNĐT.
- u. Cho biết danh sách gồm mã, họ tên, phái, ngày sinh của các giáo viên thuộc bộ môn tên là 'Hệ thống thông tin'.
- v. Với những đề tài cấp trường và cấp Đại học quốc gia thuộc chủ đề là 'Quản lý giáo dục', cho biết mã và tên các giáo viên làm chủ nhiệm đề tài.
- w. Cho biết danh sách giáo viên và tên người quản lý chuyên môn với kết quả gồm các cột sau: MÃGV, HỌTÊN, NGÀYSINH, TÊN_GVQLCM. Chỉ xuất thông tin các giáo viên có người quản lý chuyên môn.
- x. Cho biết danh sách gồm mã và tên các giáo viên có tham gia đề tài tên là 'HTTT quản lý các trường ĐH' hoặc đề tài có tên là 'HTTT quản lý giáo vụ cho một Khoa'.
- y. Cho biết danh sách gồm mã và tên các giáo viên vừa có tham gia đề tài tên là 'Ứng dụng hóa học xanh' vừa có tham gia đề tài có tên là 'Nghiên cứu tế bào gốc'.
- z. Những giáo viên nào chưa từng tham gia đề tài (mã giáo viên, tên giáo viên).
 - aa. Cho biết danh sách các giáo viên có người quản lý chuyên môn không ở cùng một thành phố.
 - bb. Cho biết danh sách các giáo viên tham gia tất cả các công việc của đề tài mã là 001.
 - cc. Có tất cả bao nhiêu giáo viên.
 - dd. Mỗi bộ môn có bao nhiêu giáo viên (mã bộ môn, tên bộ môn, số giáo viên).
 - ee. Mỗi bộ môn có bao nhiêu giáo viên sinh trước năm 1975 (mã bộ môn, tên bộ môn, số giáo viên).
 - ff. Cho biết những bộ môn có số giáo viên nữ lớn hơn 5 (mã bộ môn, tên bộ môn, số giáo viên nữ).
 - gg. Có bao nhiêu đề tài được thực hiện từ năm 2007 đến năm 2010.

- hh. Thêm vào bảng THAMGIAĐT các bộ dữ liệu cho biết giáo viên mã là 003 tham gia tất cả các công việc của đề tài mã là 001.
 - ii. Xóa các dòng dữ liệu liên quan đến đề tài 002 trong bảng THAMGIAĐT.
 - jj. Cập nhật lương của những giáo viên thuộc bộ môn mã là HTTT tăng 1.5 lần.
 - kk. Sửa phụ cấp cho những giáo viên tham gia đề tài mã là 006 thành 2.
2. Cho lược đồ CSDL sau (mô tả ở bài tập 2 chương 3):
- SINHVIÊN (MÃSV, HỌTÊN, PHÁI, NGÀYSINH, ĐCHỈ, MÃNGÀNH)
NGÀNH (MÃNGÀNH, TÊNNGÀNH, SỐCĐ, TSSV)
CHUYÊNĐỀ (MÃCĐ, TÊNCĐ, SỐSVTD)
CĐ_NGÀNH (MÃCĐ, MÃNGÀNH)
CĐ_MỞ (MÃCĐ, HỌCKỲ, NĂM)
ĐĂNGKÝ (MÃSV, MÃCĐ, HỌCKỲ, NĂM, ĐIỂM)

Hãy thực hiện những nhu cầu truy vấn sau bằng ngôn ngữ đại số quan hệ:

- a. Liệt kê danh sách sinh viên gồm mã, họ tên, phái, ngày sinh.
- b. Liệt kê danh sách sinh viên thuộc ngành tên là 'Hệ thống thông tin' (MÃSV, HỌTÊN, PHÁI, NGÀYSINH).
- c. Cho biết các ngành có tổng số sinh viên theo học từ trước đến nay lớn hơn 2000 (MÃNGÀNH, TÊNNGÀNH).
- d. Những chuyên đề nào chỉ cho phép không quá 100 sinh viên đăng ký mỗi khi được mở (MÃCĐ, TÊNCĐ).
- e. Danh sách các chuyên đề bắt buộc đối với ngành tên là 'Mạng máy tính' (MÃCĐ, TÊNCĐ).
- f. Mỗi chuyên đề có tất cả bao nhiêu ngành phải học (MÃCĐ, TÊNCĐ, SỐ_NGÀNH).
- g. Danh sách các sinh viên đăng ký học một chuyên đề tên là 'Java' nhiều hơn 1 lần (MÃSV, HỌTÊN).
- h. Cho danh sách các sinh viên thuộc ngành tên là 'Hệ thống thông tin' đã đăng ký học chuyên đề "Oracle" (MÃSV, HỌTÊN).
- i. Danh sách các ngành phải học nhiều hơn 2 chuyên đề (MÃNGÀNH, TÊNNGÀNH).
- j. Cho danh sách các sinh viên đã đăng ký nhiều hơn 2 chuyên đề trong học kỳ 1 năm 2009 (MÃSV, HỌTÊN).

- k. Cho biết các ngành phải học chuyên đề 'Java' hoặc chuyên đề 'Oracle'.
- l. Cho biết các ngành vừa phải học chuyên đề 'Java' vừa phải học chuyên đề 'Oracle'.
- m. Cho biết các ngành phải học chuyên đề 'Java' nhưng không phải học chuyên đề 'Oracle'.
- n. Liệt kê tên các chuyên đề mà sinh viên có mã là "0012345" đã học.
- o. Danh sách các sinh viên đã đăng ký học 2 chuyên đề trong học kỳ 1 năm 2004.
- p. Danh sách các sinh viên đã đăng ký học 2 chuyên đề trong học kỳ 1 năm 2004 đều có điểm là "Đạt".
- q. Cho danh sách các sinh viên đã học tất cả các chuyên đề bắt buộc đối với ngành 'Hệ thống thông tin'.
- r. Danh sách các sinh viên đã đăng ký học nhiều hơn 1 chuyên đề trong năm học 2005.
- s. Danh sách các sinh viên thuộc ngành 'Hệ thống thông tin' đã học chuyên đề 'Oracle' mà không học chuyên đề 'CSDL phân tán' trong năm 2005.
- t. Cho đến hiện tại, cho biết mỗi chuyên ngành có bao nhiêu sinh viên theo học.
- u. Liệt kê các thể hiện dữ liệu cho biết tất cả các sinh viên thuộc ngành tên là 'Hệ thống thông tin' đăng ký học tất cả các chuyên đề bắt buộc đối với ngành 'Hệ thống thông tin' trong học kỳ 1 năm 2010 (MÃSV, MÃCD, HỌCKỲ, NĂM).
- v. Danh sách các sinh viên chưa học chuyên đề nào (MÃSV, HỌTÊN).
- w. Cho biết năm nào, học kỳ nào mở tất cả các chuyên đề bắt buộc cho ngành "Hệ thống thông tin".
- x. Cho biết mã, tên của các chuyên đề thuộc chuyên ngành của sinh viên có mã là "0012345" mà sinh viên này chưa đăng ký học.
- y. Danh sách các sinh viên thuộc ngành "Hệ thống thông tin" chỉ học duy nhất 1 chuyên đề trong học kỳ 1 năm 2005.
- z. Cho biết tên các chuyên đề mà mọi ngành đều phải học chung.
- aa. Danh sách các chuyên đề bắt buộc đối với chuyên ngành tên là "Mạng máy tính" đã được mở ra trong học kỳ 1 năm 2005.

Chương 5 Ngôn ngữ đại số quan hệ

- bb. Danh sách các chuyên đề vừa là chuyên đề bắt buộc cho chuyên ngành tên là “Hệ thống thông tin” vừa là chuyên đề bắt buộc cho chuyên ngành tên là “Công nghệ tri thức”.
- cc. Cho danh sách các sinh viên chưa từng học lại một chuyên đề nào.
3. Cho CSDL sau:
- THỦY THỦ(MÃTT, TÊN TT, BẬC, TUỔI)
TÀU (MÃ TÀU, TÊN TÀU, MÀU)
ĐĂNG KÝ (MÃTT, MÃ TÀU, NGÀY)
- Hãy viết bằng đại số quan hệ đối với những yêu cầu truy vấn sau đây:
- Tên của những thủy thủ đã đăng ký tàu mã là 103.
 - Tên của những thủy thủ có bậc lớn hơn 7.
 - Tên và tuổi của những thủy thủ có bậc lớn hơn 7.
 - Tìm tên thủy thủ, tên tàu và ngày đăng ký của tất cả những lần đăng ký.
 - Tên của những thủy thủ có đăng ký tàu màu đỏ.
 - Tìm màu của những con tàu mà thủy thủ tên là Hùng đã đăng ký.
 - Tên của những thủy thủ đã đăng ký ít nhất 1 con tàu.
 - Tên của những thủy thủ đã có lần đăng ký con tàu màu đỏ hoặc con tàu màu xanh.
 - Tên của những thủy thủ đã có lần đăng ký con tàu màu đỏ lẫn con tàu màu xanh.
 - Tên của những thủy thủ đã đăng ký tối thiểu 2 con tàu.
 - Tìm mã của những thủy thủ có tuổi lớn hơn 20 chưa từng đăng ký con tàu màu đỏ.
 - Tên của những thủy thủ đã đăng ký tất cả các con tàu.
 - Tên của những thủy thủ đã đăng ký tất cả những con tàu có tên là “Marine”.
 - Tên thủy thủ đã đăng ký tất cả những con tàu màu đỏ.

Chương 6

NGÔN NGỮ SQL

I. GIỚI THIỆU

Ngôn ngữ SQL tạo nên sự thành công của CSDL quan hệ trong giới thương mại. SQL (*Structured Query Language*), trước kia được gọi là SEQUEL (*Structured English QUERy Language*), được thiết kế và thử nghiệm tại phòng nghiên cứu của IBM trong một hệ thống CSDL quan hệ có tên gọi là System R. Hiện tại, SQL là ngôn ngữ chuẩn cho các HQT CSDL quan hệ thương mại. SQL có nhiều phiên bản được tạo bởi tổ chức ANSI (*the American National Standards Institute*) và tổ chức ISO (*the International Standards Organization*). Các phiên bản đó gồm SQL-86, SQL-92 và SQL-99.

Chương này gồm các phần sau: Phần II sẽ trình bày đến các lệnh DDL của SQL gồm có các lệnh tạo lược đồ, lệnh tạo bảng và các kiểu dữ liệu trong SQL. Phần III chúng tôi sẽ trình bày cách thể hiện các ràng buộc cơ bản như ràng buộc khóa chính, khóa ngoại. Phần IV sẽ trình bày các lệnh dùng để thay đổi lược đồ, bảng, RBT. Phần V sẽ trình bày các lệnh rút trích dữ liệu đơn giản. Phần VI sẽ trình bày về các lệnh rút trích dữ liệu phức tạp hơn như sử dụng hàm tính tổng hợp, gom nhóm. Phần cuối cùng trình bày về các lệnh thêm, xóa, và cập nhật dữ liệu. Toàn bộ nội dung các phần trong chương sẽ được trình bày theo chuẩn SQL-99 và sử dụng lược đồ CSDL *Quản lý để tài nghiên cứu khoa học* đã được trình bày ở Chương 3. Một số ví dụ đặc trưng được trình bày ở Chương 5 và một số ví dụ mới được thêm vào để minh họa rõ ràng hơn cho lệnh truy vấn thông tin.

II. NGÔN NGỮ ĐỊNH NGHĨA DỮ LIỆU VÀ KIỂU DỮ LIỆU

SQL sử dụng các khái niệm bảng, dòng, cột để thay thế cho khái niệm tương ứng là quan hệ, bộ và thuộc tính trong mô hình dữ liệu quan hệ. Câu lệnh định nghĩa dữ liệu trong SQL là lệnh *CREATE*. Nó được sử dụng để tạo lược đồ (*schema*), tạo bảng (*table*), tạo khung nhìn (*view*) và ràng buộc (*assertion, trigger*).

II.1 Lệnh CREATE TABLE

Lệnh CREATE TABLE dùng để tạo một bảng mới bằng cách đưa ra tên bảng, các cột của bảng và các RBTW trên bảng đó. Mỗi cột có tên cột, kiểu dữ liệu để quy định miền giá trị, và các ràng buộc trên cột, chẳng hạn như ràng buộc NULL/NOT NULL. Các ràng buộc khóa chính, ràng buộc tham chiếu, RBTW trên một cột có thể khai báo trong phạm vi lệnh CREATE TABLE, sau khi chúng ta khai báo các cột. Tuy nhiên, các loại ràng buộc này cũng có thể được khai báo bằng lệnh ALTER TABLE được trình bày ở phần sau.

Cú pháp:

```
CREATE TABLE <Tên_bảng> (
    <Tên_cột> <Kiểu_dữ_liệu> [<RBTW>],
    <Tên_cột> <Kiểu_dữ_liệu> [<RBTW>],
    ...
    [<RBTW>]
)
```

Ví dụ: Tạo bảng GIÁOVIÊN VÀ BỘMÔN được trình bày trong lược đồ CSDL quan hệ ở chương 3. Trong các lệnh tạo bảng này, chúng tôi chưa quan tâm đến việc tạo các RBTW cho các cột.

```
CREATE TABLE GIÁOVIÊN (
```

MÃGV	CHAR(10),
HỌTÊN	VARCHAR(30),
LUONG	INT,
PHÁI	CHAR(10),

NGÀY SINH DATETIME,
SỐ NHÀ INT,
ĐƯỜNG VARCHAR(30),
QUẬN CHAR(10),
THÀNH PHỐ VARCHAR(20),
GVQLCM CHAR(10),
MÃ BM CHAR(10)
)

CREATE TABLE BỘ MÔN(

MÃ BM	CHAR(10),
TÊN BM	VARCHAR(30),
PHÒNG	CHAR(10),
ĐIỆN THOẠI	VARCHAR(20),
TRƯỞNG GBM	CHAR(10)
MÃ KHOA	CHAR(10),
NGÀY NHẬN CHỨC	DATETIME

)

II.2 Kiểu dữ liệu

Các kiểu dữ liệu cơ bản cho một cột bao gồm:

- *Kiểu số (numeric)* bao gồm các kiểu số nguyên với kích thước khác nhau (INTEGER hoặc INT và SMALLINT), các kiểu số thực với độ chính xác khác nhau (FLOAT hoặc REAL, và DOUBLE PRECISION). Các kiểu số thực này có thể khai báo theo định dạng mong muốn bằng cách sử dụng DECIMAL(i, j) hoặc DEC(i, j) hoặc NUMERIC(i, j) với i là độ chính xác của phần thập phân, còn j là phần số lẻ.

- **Kiểu chuỗi (character string)** gồm có các kiểu chuỗi có chiều dài cố định(CHAR(n) hoặc CHARACTER(n)) với n là số ký tự của chuỗi, các kiểu chuỗi có chiều dài thay đổi (VARCHAR(n) hoặc CHAR VARYING(n) hoặc CHARACTER VARYING(n)) với n là số ký tự tối đa của chuỗi.
- **Kiểu chuỗi Bit (Bit string)** cũng gồm có kiểu chuỗi bit cố định (BIT(n)) hoặc kiểu chuỗi bit biến động (BIT VARYING (n)) với n là số bit tối đa.
- **Kiểu BOOLEAN:** kiểu này có giá trị là TRUE hoặc FALSE.
- Các kiểu dữ liệu liên quan đến NGÀY và GIỜ: kiểu DATE (chứa ngày, tháng, năm), kiểu TIME (chứa giờ, phút, giây), kiểu TIMESTAMP (chứa ngày và giờ).

II.3 Lệnh CREATE DOMAIN

Lệnh CREATE DOMAIN được gọi là lệnh tạo miền giá trị. Lệnh này được dùng để tạo ra một kiểu dữ liệu mới sử dụng những kiểu dữ liệu có sẵn.

Cú pháp:

CREATE DOMAIN <Tên_kdl_mới> AS <Kiểu_dữ_liệu>

Ví dụ:

CREATE DOMAIN KIỀU_CHUỖI AS VARCHAR(30)

III. KHAI BÁO RÀNG BUỘC CƠ BẢN TRONG SQL

Trong phần này chúng tôi sẽ trình bày một số ràng buộc cơ bản có thể được khai báo trong phạm vi của lệnh CREATE TABLE. Những ràng buộc này bao gồm: ràng buộc khóa chính, ràng buộc tham chiếu, ràng buộc trên cột và ràng buộc trên các bộ trong phạm vi một bảng.

III.1 Ràng buộc trên thuộc tính

Có những loại ràng buộc trên thuộc tính như sau:

- NOT NULL: ràng buộc này quy định các giá trị tại cột được đặt ràng buộc này không được phép rỗng. Ví dụ cột MÃGV của GIÁO VIÊN không được phép rỗng.
- NULL: là ràng buộc mặc định của các cột, nó cho phép giá trị tại cột có thể nhận giá trị rỗng. Ví dụ cột NGÀY SINH của GIÁO VIÊN được phép nhận giá trị rỗng.
- UNIQUE: ràng buộc này quy định các giá trị tại cột được đặt ràng buộc này là duy nhất. Ví dụ giá trị tại cột TÊN BM của BỘ MÔN là duy nhất.
- DEFAULT: ràng buộc này cho phép chúng ta đặt giá trị mặc định cho cột bất kỳ. Ví dụ cột LƯƠNG của GIÁO VIÊN được mặc định là 2 triệu.
- CHECK: ràng buộc này cho phép chúng ta kiểm tra các giá trị của một cột chỉ được phép nằm trong miền giá trị cho trước. Ví dụ: cột PHÁI của giáo viên chỉ được nhận giá trị {'Nam' hoặc 'Nữ'}.

Các ràng buộc trên được thể hiện cụ thể ở 2 lệnh tạo bảng GIÁO VIÊN và BỘ MÔN như sau:

CREATE TABLE GIÁO VIÊN (

MÃGV	CHAR(10)	NOT NULL,
HỌ TÊN	VARCHAR(30)	NOT NULL,
LƯƠNG	INT	DEFAULT(2000000),
PHÁI	CHAR(10)	CHECK(PHAI IN ('Nam', 'Nu')),
NGÀY SINH	DATETIME,	
SỐ NHÀ	INT,	
ĐƯỜNG	VARCHAR(30),	
QUẬN	CHAR(10),	
THÀNH PHỐ	VARCHAR(20),	
GVQLCM	CHAR(10),	

MÃBM CHAR(10)

)

CREATE TABLE BỘMÔN(

MÃBM	CHAR(10)	NOT NULL,
TÊNBM	VARCHAR(30)	UNIQUE,
PHÒNG	CHAR(10),	
ĐIỆNTHOẠI	VARCHAR(20),	
TRƯỞNGBM	CHAR(10),	
MÃKHOA	CHAR(10),	
NGÀYNHẬNCHỨC	DATETIME	

)

III.2 Ràng buộc khóa chính và ràng buộc khóa ngoại

- PRIMARY KEY: ràng buộc này dùng để khai báo khóa chính. Ví dụ khai báo khóa chính là cột MÃGV cho bảng GIÁOVIÊN, cột MÃBM cho bảng BỘMÔN.
- FOREIGN KEY / REFERENCES: ràng buộc này dùng để khai báo khóa ngoại cho một bảng hay còn được gọi là ràng buộc tham chiếu khóa ngoại/khóa chính. Ví dụ khai báo khóa ngoại trên cột MÃBM của bảng GIÁOVIÊN tham chiếu đến khóa chính là cột MÃBM trên bảng BỘMÔN.

CREATE TABLE GIÁOVIÊN (

MÃGV	CHAR(10)	PRIMARY KEY,
HOTÊN	VARCHAR(30)	NOT NULL,
LUONG	INT	DEFAULT(2000000),
PHÁI	CHAR(10)	CHECK(PHAI IN ('Nam', 'Nu'),
NGÀYSINH	DATETIME,	
SỐNHÀ	INT,	

```
ĐƯỜNG      VARCHAR(30),  
QUẬN       CHAR(10),  
THÀNHPHỐ  VARCHAR(20),  
GVQLCM     CHAR(10),  
MÃBM       CHAR(10)  FOREIGN KEY (MÃBM)  
                           REFERENCES BỘMÔN(MÃBM)  
)
```

III.3 Đặt tên cho ràng buộc

Chúng ta có thể đặt tên cho các RBTW trên cột theo cú pháp sau:

CONSTRAINT <Tên_RBTW> <RBTW>

Ví dụ:

```
CREATE TABLE GIÁOVIÊN (  
    MÃGV CHAR(10)  
        CONSTRAINT GV_MAGV_PK PRIMARY KEY,  
    HỌTÊN VARCHAR(30)  
        CONSTRAINT GV_HOTEN_NN NOT NULL,  
    LƯƠNG INT  
        CONSTRAINT GV_LUONG_DF DEFAULT(2000000),  
    PHÁI CHAR(10) CONSTRAINT GV_LUONG_CHECK  
        CHECK(PHAI IN ('Nam', 'Nu'),  
    NGÀYSINH DATETIME,  
    SỐNHÀ INT,  
    ĐƯỜNG VARCHAR(30),  
    QUẬN CHAR(10),
```

THÀNHPHỐ VARCHAR(20),
GVQLCM CHAR(10),
MÃBM CHAR(10) FOREIGN KEY (MÃBM)
REFERENCES BỘMÔN(MÃBM)

)

CREATE TABLE THAMGIAĐT(

MÃGV CHAR(10),
MÃĐT CHAR(10),
STT INT,
PHỤCÁP INT,
KẾTQUẢ CHAR(10),
CONSTRAINT TGĐT_PK PRIMARY KEY
(MÃGV,MÃ ĐT, STT),
CONSTRAINT TGĐT_FK FOREIGN KEY
(MÃĐT, STT)
REFERENCES CÔNGVIỆC(MÃĐT, STT)

)

IV. CÁC LỆNH CẬP NHẬT LUỢC ĐỒ

Các lệnh cập nhật lược đồ dùng để thay đổi cấu trúc hoặc ràng buộc của bảng. Các lệnh gồm có: thêm cột, xóa cột, mở rộng cột, thêm ràng buộc, xóa ràng buộc.

IV.1 Lệnh ALTER

- ❖ Lệnh thêm một cột:

ALTER TABLE <Tên_bảng> ADD <Tên_cột> <Kiểu_dữ_liệu>
[<RBTV>]

Ví dụ: Thêm cột HỌCVIỆN vào bảng GIÁOVIÊN

ALTER TABLE GIÁOVIÊN ADD HỌCVIỆN CHAR(20)

❖ Lệnh xóa cột:

ALTER TABLE <Tên_bảng> DROP COLUMN <Tên_cột>

Ví dụ: Xóa cột HỌCVIỆN ra khỏi bảng GIÁOVIÊN

ALTER TABLE GIÁOVIÊN DROP COLUMN HỌCVIỆN

❖ Lệnh mở rộng cột:

ALTER TABLE <Tên_bảng> ALTER COLUMN <Tên_cột>
<Kiểu_dữ_liệu_mới>

Ví dụ: Mở rộng chiều dài của cột HỌCVIỆN thành 50.

ALTER TABLE GIÁOVIÊN ALTER COLUMN HỌCVIỆN
CHAR(50)

❖ Lệnh thêm ràng buộc:

ALTER TABLE <Tên_bảng> ADD
CONSTRAINT <Ten_RBTV> <RBTV>,
CONSTRAINT <Ten_RBTV> <RBTV>,
...

Ví dụ: Chúng ta tạo bảng KHOA, sau đó dùng lệnh ALTER TABLE để thêm ràng buộc khóa chính và khóa ngoại cho bảng KHOA.

CREATE TABLE KHOA(

MÃKHOA **CHAR(10)** NOT NULL,

```

TÊNKHOA      VARCHAR(30),
NĂMTL        INT,
PHÒNG        CHAR(10),
ĐIỆNTHOẠI   VARCHAR(20),
TRƯỜNGKHOA  CHAR(10),
NGÀYNHẬNCHỨC DATETIME

```

ALTER TABLE KHOA ADD

CONSTRAINT KHOA PK PRIMARY KEY (MÃ KHOA).

CONSTRAINT KHOA_FK_TK FOREIGN KEY (TRƯỚNG KHOA)

REFERENCES GIÁO VIÊN (MÃGV)

❖ Lệnh xóa ràng buộc:

ALTER TABLE <Tên_bảng> DROP <Tên_RBTV>

Ví dụ: Xóa ràng buộc khóa ngoại TRƯỞNGKHOA ra khỏi bảng KHOA.

ALTER TABLE KHOA DROP KHOA FK TK

IV.2 Lệnh DROP

Lệnh **DROP** được sử dụng để xóa một bảng. Lệnh **DROP** có cú pháp:

DROP TABLE <Tên bảng>

Ví dụ:

Xóa bảng GIÁOVIÊN: **DROP TABLE GIÁOVIÊN**

Xóa bảng BỘMÔN: **DROP TABLE BỘMÔN**

V. TRUY VẤN CƠ BẢN TRONG SQL

SQL hỗ trợ lệnh cơ bản để rút trích thông tin từ một CSDL đó là lệnh SELECT. Trong phần này chúng tôi sẽ giới thiệu cú pháp cơ bản của lệnh SELECT và những tùy chọn của nó.

V.1 Cấu trúc SELECT-FROM-WHERE

Ngôn ngữ SQL được chúng tôi trình bày từ đơn giản đến phức tạp hơn. Câu lệnh truy vấn thường gồm có 3 mệnh đề SELECT, FROM và WHERE được thể hiện như sau:

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

Trong đó:

- <danh sách các cột>: là tên các cột cần được hiển thị trong kết quả truy vấn.
- <danh sách các bảng>: là tên các bảng liên quan đến câu truy vấn.
- <điều kiện>: là biểu thức boolean xác định dòng nào sẽ được rút trích.

Trong SQL các toán tử so sánh được sử dụng để so sánh giá trị của thuộc tính này với giá trị của thuộc tính khác hoặc giữa giá trị của thuộc tính và hằng số là <, >, ≤, ≥, ≠, =. Ngoài ra, để nối các biểu thức trong mệnh đề điều kiện SQL sử dụng AND, OR, NOT.

Query 0: Cho biết khoa được thành lập từ năm 1980 trở về sau.

Q0: **SELECT MÃKHOA, TÊNKHOA, NAMTL, PHÒNG,**

ĐIỆNTHOẠI, TRƯỞNGKHOA, NGÀYNHẠNCHỨC
FROM KHOA
WHERE NĂMTL >= 1980

Câu truy vấn **Q0** lấy dữ liệu từ bảng KHOA. Câu truy vấn này rút trích các bộ dữ liệu từ bảng KHOA thỏa điều kiện NĂMTL >= 1980. Kết quả của **Q0** được thể hiện ở Hình VI-1.

MÃKHOA	TÊNKHOA	NĂMTL	PHÒNG	ĐIỆNTHOẠI	TRƯỞNGKHOA	NGÀYNHẠNCHỨC
CNTT	Công nghệ thông tin	1995	B11	838123456	2	20/02/2005
SH	Sinh học	1980	B31	838454545	4	11/10/2000
HH	Hóa học	1980	B41	838456456	7	15/10/2001

Hình VI-1 Kết quả của truy vấn Q0.

Query 1: Cho biết mã giáo viên, họ tên giáo viên có lương lớn hơn 2000.

Q1: SELECT MÃGV, HỌTÊN

FROM GIÁOVIÊN

WHERE LUONG > 2000

Câu truy vấn **Q1** lấy dữ liệu từ bảng GIÁOVIÊN. Câu truy vấn này rút trích các bộ dữ liệu từ bảng GIÁOVIÊN thỏa điều kiện LUONG > 2000 và sau đó chiếu lên thuộc tính MÃGV, HỌTÊN. Kết quả của **Q1** được thể hiện ở Hình VI-2.

MÃGV	HỌ TÊN	LUONG
2	Trần Trà Hương	2500
3	Nguyễn Ngọc Ánh	2200
4	Trương Nam Sơn	2300
5	Lý Hoàng Hà	2500
7	Nguyễn An Trung	2100

Hình VI-2 Kết quả của truy vấn Q1.

Query 2: Cho biết mã khoa, tên khoa và tên giáo viên làm trưởng khoa đó.

Q2: SELECT MÃKHOA, TÊNKHOA, HỌTÊN
FROM KHOA, GIÁOVIÊN
WHERE TRƯỞNGKHOA = MÃGV

Câu truy vấn **Q2** lấy dữ liệu từ bảng KHOA và GIÁOVIÊN. Trong mệnh đề WHERE của câu truy vấn này chỉ có điều kiện kết TRƯỞNGKHOA = MÃGV. Điều kiện kết trong câu lệnh SELECT tương tự như điều kiện kết trong ngôn ngữ đại số quan hệ. Kết quả của **Q2** được thể hiện ở Hình VI-3.

MÃKHOA	TÊNKHOA	TRƯỞNG KHOA
CNTT	Công nghệ thông tin	Trần Trà Hương
HH	Hóa học	Nguyễn An Trung
SH	Sinh học	Trương Nam Sơn
VL	Vật lý	Lý Hoàng Hà

Hình VI-3 Kết quả của truy vấn Q2.

V.2 Thuộc tính trùng, cách đặt bí danh và các biến bộ

❖ Thuộc tính trùng

Trong CSDL đôi khi chúng ta đặt tên thuộc tính cho các quan hệ khác nhau là giống nhau. Để truy xuất đến các thuộc tính giống nhau này bắt buộc chúng ta phải chỉ định các thuộc tính này một cách rõ ràng bằng cách sử dụng tên bảng để truy xuất đến thuộc tính của nó.

Query 3: Cho biết mã giáo viên, họ tên giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q3: SELECT MÃGV, HỌTÊN

FROM GIÁOVIÊN, BỘMÔN
WHERE GIÁOVIÊN.MÃBM = BỘMÔN.MÃBM
AND TÊNBM = ‘Hệ thống thông tin’

Câu truy vấn **Q3** lấy dữ liệu từ bảng GIÁOVIÊN và BỘMÔN. Mệnh đề WHERE của câu truy vấn này có điều kiện chọn TÊNBM = ‘Hệ

thông thông tin' và điều kiện kết GIÁOVIÊN.MÃBM = BỘMÔN.MÃBM. Vì khóa chính và khóa ngoại trong phép kết có tên trùng nhau nên chúng ta phải thể hiện như sau: GIÁOVIÊN.MÃBM, BỘMÔN.MÃBM). Nếu không sử dụng cách này chúng ta có thể dùng cách đặt bí danh, phần này sẽ được trình bày ở phần kế tiếp. Kết quả của **Q3** được thể hiện ở Hình VI-4.

MÃGV	HỌ TÊN
2	Trần Trà Hương
3	Nguyễn Ngọc Ánh

Hình VI-4 Kết quả của truy vấn Q3.

❖ **Cách đặt bí danh và biến bộ**

SQL cho phép chúng ta đặt bí danh cho các bảng dữ liệu bằng từ khóa AS. Một bảng trong một lệnh truy vấn có thể có nhiều bí danh khác nhau, ví dụ như ở **Q4**. Sau khi đặt bí danh cho bảng, chúng ta sử dụng bí danh này để truy xuất đến các thuộc tính bên trong bảng. Lưu ý, trong SQL cũng cho phép chúng ta khai báo bí danh không tường minh bằng cách lược bỏ từ khóa AS, ví dụ như ở **Q4a**.

Query 3a: Cho biết mã giáo viên, họ tên giáo viên thuộc bộ môn 'Hệ thống thông tin'.

Q3a: `SELECT GV.MÃGV, GV.HOTÊN
FROM GIÁOVIÊN AS GV, BỘMÔN AS BM
WHERE GV.MÃBM = BM.MÃBM
AND TÊNBM = 'Hệ thống thông tin'`

Kết quả của **Q3a** trả về giống kết quả của **Q3** và thể hiện ở Hình VI-4.

Query 4: Cho biết mã giáo viên, họ tên giáo viên và họ tên người quản lý chuyên môn giáo viên đó.

Q4: `SELECT GV.MÃGV, GV.HOTÊN,
QLCM.HOTÊN AS GVQLCM
FROM GIÁOVIÊN AS GV, GIÁOVIÊN AS QLCM`

WHERE GV.GVQLCM = QLCM.MÃGV

Trong truy vấn **Q4**, bảng GIÁOVIÊN được đặt hai bí danh là GV và QLCM. GV đại diện bảng dữ liệu cần lấy thông tin của giáo viên và QLCM đại diện bảng dữ liệu cần lấy thông tin của giáo viên quản lý chuyên môn của một giáo viên bất kỳ.

MÃGV	HỌ TÊN	GVQLCM
3	Nguyễn Ngọc Ánh	Trần Trà Hương
6	Trần Bạch Tuyết	Trương Nam Sơn
8	Trần Trung Hiếu	Nguyễn An Trung
9	Trần Hoàng Nam	Nguyễn Hoài An
10	Phạm Nam Thành	Nguyễn An Trung

Hình VI-5 Kết quả của truy vấn Q4.

Query 4a: Cho biết mã giáo viên, họ tên giáo viên và họ tên người quản lý chuyên môn giáo viên đó.

Q4a: `SELECT GV.MÃGV, GV.HỌTÊN, QLCM.HỌTÊN
FROM GIÁOVIÊN GV, GIÁOVIÊN QLCM
WHERE GV.GVQLCM = QLCM.MÃGV`

Trong truy vấn **Q4a**, bảng GIÁOVIÊN được đặt hai bí danh là GV và QLCM nhưng không sử dụng từ khóa AS, AS được lược bỏ và thay thế bằng dấu cách. Kết quả của **Q4a** hoàn toàn tương tự như **Q4**.

V.3 Lệnh SELECT không sử dụng mệnh đề WHERE và sử dụng ký tự đại diện

❖ Lệnh SELECT không sử dụng mệnh đề WHERE

Trong câu lệnh truy vấn, nếu mệnh đề WHERE không được khai báo thì kết quả của câu truy vấn sẽ là:

- Đối với câu truy vấn mà mệnh đề FROM chỉ sử dụng một bảng dữ liệu thì tất cả các bộ của bảng dữ liệu đó sẽ được rút trích. Ví dụ cho trường hợp này được thể hiện ở **Q5**.

- Đối với câu truy vấn mà mệnh đề FROM sử dụng nhiều bảng dữ liệu thì việc rút trích được thực hiện trên kết quả phép kết giữa các bảng dữ liệu này. Ví dụ cho trường hợp này được thể hiện ở **Q6**.

Query 5: Cho biết mã giáo viên, họ tên giáo viên.

Q5: `SELECT MÃGV, HỌTÊN`

`FROM GIÁOVIÊN`

Q5 sẽ rút trích toàn bộ thông tin mã giáo viên, họ tên từ bảng **GIÁOVIÊN**. Kết quả của **Q5** được thể hiện ở **Hình VI-6**.

MÃGV	HỌTÊN
1	Nguyễn Hoài An
2	Trần Trà Hương
3	Nguyễn Ngọc Ánh
4	Trương Nam Sơn
5	Lý Hoàng Hà
6	Trần Bạch Tuyết
7	Nguyễn An Trung
8	Trần Trung Hiếu
9	Trần Hoàng Nam
10	Phạm Nam Thành

Hình VI-6 Kết quả của Q5.

Query 6: Cho biết kết quả phép tích Đè-các của hai quan hệ **GIÁOVIÊN** và **BỘMÔN**, chỉ xuất ra họ tên giáo viên và tên bộ môn.

Q6: `SELECT HỌTÊN, TÊNBỘMÔN`

`FROM GIÁOVIÊN, BỘMÔN`

Q6 sẽ rút trích toàn bộ dữ liệu họ tên từ bảng giáo viên, tên bộ môn từ bảng bộ môn. Sau đó thực hiện kết nối các bộ dữ liệu từ hai kết quả

Chương 6 Ngôn ngữ SQL

trung gian trên để cho ra kết quả sau cùng. Kết quả của **Q6** được thể hiện ở Hình VI-7.

MÃBM	TÊNBM	...	HỌTÊN	TÊNBM
HTTT	Hệ thống thông tin	...	Nguyễn Hoài An	Hệ thống thông tin
CNTT	Công nghệ tri thức	...	Trần Trà Hương	Hệ thống thông tin
MMT	Mạng máy tính	...	Nguyễn Ngọc Ánh	Hệ thống thông tin
VLDT	Vật lý điện tử	...	Trương Nam Sơn	Hệ thống thông tin
VLUĐ	Vật lý ứng dụng	...	Lý Hoàng Hà	Hệ thống thông tin
VS	Vi sinh	...	Trần Bạch Tuyết	Hệ thống thông tin
SH	Sinh hóa	...	Nguyễn An Trung	Hệ thống thông tin
HL	Hóa lý	...	Trần Trung Hiếu	Hệ thống thông tin
HPT	Hóa phân tích	...	Trần Hoàng Nam	Hệ thống thông tin
HHC	Hóa hữu cơ	...	Phạm Nam Thành	Hệ thống thông tin

(a)

MÃGV	HỌTÊN	...	HỌTÊN	TÊNBM
1	Nguyễn Hoài An	...	Nguyễn Hoài An	Mạng máy tính
2	Trần Trà Hương	...	Trần Trà Hương	Mạng máy tính
3	Nguyễn Ngọc Ánh	...	Nguyễn Ngọc Ánh	Mạng máy tính
4	Trương Nam Sơn	...	Trương Nam Sơn	Mạng máy tính
5	Lý Hoàng Hà	...	Lý Hoàng Hà	Mạng máy tính
6	Trần Bạch Tuyết	...	Trần Bạch Tuyết	Mạng máy tính
7	Nguyễn An Trung	...	Nguyễn An Trung	Mạng máy tính
8	Trần Trung Hiếu	...	Trần Trung Hiếu	Mạng máy tính
9	Trần Hoàng Nam	...	Trần Hoàng Nam	Mạng máy tính
10	Phạm Nam Thành	...	Phạm Nam Thành	Mạng máy tính

(b)

HỌTÊN	TÊNBM
Nguyễn Hoài An	Hệ thống thông tin
Trần Trà Hương	Hệ thống thông tin
Nguyễn Ngọc Ánh	Hệ thống thông tin
Trương Nam Sơn	Hệ thống thông tin
Lý Hoàng Hà	Hệ thống thông tin
Trần Bạch Tuyết	Hệ thống thông tin
Nguyễn An Trung	Hệ thống thông tin
Trần Trung Hiếu	Hệ thống thông tin
Trần Hoàng Nam	Hệ thống thông tin
Phạm Nam Thành	Hệ thống thông tin
Nguyễn Hoài An	Mạng máy tính
Trần Trà Hương	Mạng máy tính
Nguyễn Ngọc Ánh	Mạng máy tính
Trương Nam Sơn	Mạng máy tính
Lý Hoàng Hà	Mạng máy tính
Trần Bạch Tuyết	Mạng máy tính
Nguyễn An Trung	Mạng máy tính
Trần Trung Hiếu	Mạng máy tính
Trần Hoàng Nam	Mạng máy tính
Phạm Nam Thành	Mạng máy tính
Nguyễn Hoài An	Sinh hóa
...	...

(c)

Hình VI-7: (a) Danh sách họ tên tất cả bộ môn, (b) Danh sách tên tất cả giáo viên, (c) Kết quả tích Đề-các (a) và (b).

❖ Sử dụng ký tự đại diện *

Để rút trích tất cả các thuộc tính của một bảng dữ liệu chúng ta không cần liệt kê tường minh tất cả thuộc tính của bảng dữ liệu đó, trong SQL cho phép chúng ta sử dụng dấu sao (*) như một ký tự đại diện cho tất cả các thuộc tính trong một bảng dữ liệu.

Query 7: Cho biết thông tin các khoa.

Q7: SELECT *

FROM KHOA

Q7 sẽ trả về toàn bộ các bộ dữ liệu có trong KHOA và chiều lên tất cả các thuộc tính của KHOA.

MÃKHOA	TÊNKHOA	NĂMTL	PHÒNG	ĐIỆNTHOAI	TRƯỞNGKHOA	NGÀYNHẬNCHỨC
CNTT	Công nghệ thông tin	1995	B11	838123456	2	20/02/2005
VL	Vật lý	1976	B21	838223223	5	18/09/2003
SH	Sinh học	1980	B31	838454545	4	11/10/2000
HH	Hóa học	1980	B41	838456456	7	15/10/2001

Hình VI-8 Kết quả của Q7.

V.4 Xử lý kết quả trả về có bộ trùng

Kết quả của một câu truy vấn có thể có những bộ dữ liệu trùng nhau. SQL cho phép chúng ta có thể giữ lại hoặc loại bỏ các bộ trùng trong kết quả trả về. Nếu chúng ta muốn loại bỏ các bộ trùng trong kết quả trả về thì sử dụng từ khóa DISTINCT trong mệnh đề SELECT. Nếu chúng ta muốn hiển thị các bộ trùng thì theo cách tường minh chúng ta khai báo từ khóa ALL trong mệnh đề SELECT. Nếu chúng ta không khai báo từ khóa nào cả thì SQL mặc định là từ khóa ALL.

Query 8a: Cho biết thông tin lương của giáo viên (yêu cầu loại bỏ giá trị trùng).

Q8a: SELECT DISTINCT LUONG

FROM GIAOVIEN

Query 8b: Cho biết thông tin lương của giáo viên (yêu cầu không loại bỏ giá trị trùng).

Q8b: SELECT ALL LUONG

FROM GIAOVIEN

(a)
(b)

Hình VI-9: (a) Kết quả của Q8a. (b) Kết quả của Q8b.

V.5 So sánh chuỗi và các phép toán số học

Tiếp theo đây là phần trình bày về các thao tác trên dữ liệu kiểu chuỗi và kiểu số trong ngôn ngữ SQL.

- ❖ So sánh chuỗi: SQL cho phép chúng ta so sánh chuỗi bằng cách sử dụng toán tử LIKE hoặc NOT LIKE. Chúng ta có thể so sánh chính xác chuỗi, hoặc so sánh gần đúng bằng cách sử dụng các ký tự đại diện để thay thế cho ký tự ta muốn so sánh. Ví dụ như ký tự phần trăm (%) đại diện cho chuỗi bất kỳ, ký tự gạch ngang (_) đại diện cho ký tự bất kỳ, từ khóa ESCAPE để loại bỏ ký tự không muốn so sánh.

Query 9a: Cho biết các giáo viên trực thuộc thành phố HCM.

Q9a: SELECT *

```
FROM GIÁO VIÊN
WHERE THÀNH PHỐ LIKE 'HCM'
```

Query 9b: Cho biết các giáo viên không thuộc thành phố HCM.

Q9b: SELECT *

```
FROM GIÁOVIÊN  
WHERE THÀNHPHỐ NOT LIKE 'HCM'
```

Query 9c: Cho biết các giáo viên có họ là 'Nguyễn'.

Q9c: SELECT *

```
FROM GIÁOVIÊN  
WHERE HỌTÊN LIKE '%Nguyễn%'
```

Query 9d: Cho biết các giáo viên sinh vào những năm 1970s. (Giả sử rằng định dạng ngày sinh của giáo viên là dd/mm/yyyy (dd: ngày, mm: tháng, yyyy: năm)).

Q9d: SELECT *

```
FROM GIÁOVIÊN  
WHERE NGÀYSINH LIKE '_/_/197_'
```

Query 9d: Cho biết các giáo viên có họ là 'Trần' (sử dụng ESCAPE).

Q9d: SELECT *

```
FROM GIÁOVIÊN  
WHERE HỌTÊN LIKE '%Trần%' ESCAPE 's'
```

- ❖ Toán tử BETWEEN: SQL cho phép chúng ta sử dụng toán tử BETWEEN để kiểm tra một giá trị (của một thuộc tính) có thuộc một khoảng giá trị cụ thể hay không. Ngoài ra chúng ta có thể sử dụng NOT BETWEEN để với ý nghĩa phủ định.

Query 10a: Cho biết các giáo viên có lương không thuộc khoảng từ 2000 và 2500.

Q10a: SELECT *

```
FROM GIÁO VIÊN  
WHERE LUƠNG BETWEEN 2000 AND 2500
```

Query 10b: Cho biết các giáo viên không có lương nằm trong khoảng từ 2000 và 2500.

Q10b: SELECT *

```
FROM GIÁO VIÊN  
WHERE LUƠNG NOT BETWEEN 2000 AND 2500
```

- ❖ Sử dụng phép toán cộng (+), trừ (-), nhân (*), chia (/): SQL cho phép chúng ta sử dụng các phép toán này trên các giá trị số hoặc các thuộc tính có miền giá trị số.

Query 11: Cho biết mã giáo viên, họ tên và lương của họ sau khi tăng lên 10%.

Q11: SELECT MÃGV, HỌTÊN, LUƠNG*1.1

```
FROM GIÁO VIÊN
```

- ❖ Đặt bí danh cho thuộc tính: SQL cho phép chúng ta đặt bí danh cho thuộc tính bằng cách sử dụng từ khóa AS.

Query 11a: Cho biết mã giáo viên, họ tên và lương của họ sau khi tăng lên 10%.

Q11a: SELECT MÃGV AS MÃ_GV, HỌTÊN AS 'HỌ TÊN',

```
LUƠNG*1.1 AS 'LUƠNG TĂNG 10%'
```

```
FROM GIÁO VIÊN
```

Kết quả truy vấn **Q11a** là một bảng gồm các cột MÃGV, HỌTÊN, LUƠNG*1.1, có ý nghĩa tương ứng là mã giáo viên, họ tên giáo viên, lương tăng 10%.

- ❖ Phép cộng chuỗi: SQL cho phép chúng ta sử dụng toán tử cộng (+) để cộng giá trị các chuỗi trên mệnh đề SELECT.

Query 12: Cho biết mã giáo viên, họ tên, địa chỉ. Yêu cầu câu địa chỉ hiển thị thành một cột gồm: số nhà + đường + quận + thành phố.

Q12: `SELECT MÃGV, HỌTÊN, SỐNHÀ + ' ' + ĐƯỜNG + ' ' + QUẬN + ' ' + THÀNHPHỐ AS ĐỊACHỈ
FROM GIÁOVIÊN`

V.6 Sắp xếp kết quả trả về với mệnh đề ORDER BY

SQL cho phép chúng ta sử dụng mệnh đề ORDER BY để sắp xếp kết quả trả về của một câu truy vấn dựa trên các giá trị của một hoặc nhiều thuộc tính.

Cú pháp:

```
SELECT <danh sách các cột>  
FROM <danh sách các bảng>  
WHERE <điều kiện>  
ORDER BY <danh sách các cột> <ASC/DESC>
```

Trong đó, ASC dùng để sắp xếp tăng dần và đây cũng là cơ chế mặc định của SQL; DESC dùng để sắp xếp giảm dần.

Query 13: Cho biết mã giáo viên, họ tên. Kết quả trả về sắp xếp mã giáo viên giảm dần.

Q13: `SELECT MÃGV, HỌTÊN
FROM GIÁOVIÊN
ORDER BY MÃGV DESC`

MÃGV	HỌTÊN
9	Trần Hoàng Nam
8	Trần Trung Hiếu
7	Nguyễn An Trung
6	Trần Bạch Tuyết
5	Lý Hoàng Hà
4	Trương Nam Sơn
3	Nguyễn Ngọc Ánh
2	Trần Trà Hương
1	Nguyễn Hoài An

Hình VI-10 Kết quả của truy vấn Q13.

Query 14: Cho biết mã giáo viên, họ tên. Kết quả trả về sắp xếp tên họ tên giáo viên tăng dần.

Q14: SELECT MÃGV, HỌTÊN

```
FROM GIÁOVIÊN
ORDER BY HỌTÊN
```

MÃGV	HỌTÊN
5	Lý Hoàng Hà
7	Nguyễn An Trung
1	Nguyễn Hoài An
3	Nguyễn Ngọc Ánh
6	Trần Bạch Tuyết
9	Trần Hoàng Nam
2	Trần Trà Hương
8	Trần Trung Hiếu
4	Trương Nam Sơn

Hình VI-11 Kết quả của truy vấn Q14.

Query 15: Cho biết mã đề tài, số thứ tự công việc, tên công việc. Kết quả trả về sắp xếp mã đề tài tăng dần, số thứ tự công việc giảm dần.

Q15: SELECT MÃĐT, STT, TÊNCV

FROM CÔNGVIỆC
ORDER BY MÃĐT ASC, STT DESC

MÃĐT	STT	TÊNCV
1	5	Cài đặt thử nghiệm
1	4	Thiết kế hệ thống
1	3	Phân tích hệ thống
1	2	Xác định yêu cầu
1	1	Khởi tạo và Lập kế hoạch
2	5	Cài đặt thử nghiệm
2	4	Thiết kế hệ thống
2	3	Phân tích hệ thống
2	2	Xác định yêu cầu
2	1	Khởi tạo và Lập kế hoạch
6	2	Nuôi cây
6	1	Lấy mẫu

Hình VI-12 Kết quả của truy vấn Q15.

V.7 Phép toán tập hợp trong SQL

SQL có cài đặt các phép toán tập hợp trên hai quan hệ khả hợp, gồm: hội (UNION), giao (INTERSECT), trừ (EXCEPT). Hai quan hệ là khả hợp khi chúng có cùng số lượng thuộc tính và các thuộc tính tương ứng cùng miền giá trị. Kết quả của các phép toán này sẽ loại bỏ các bộ trùng nhau. Nếu chúng ta muốn giữ lại các bộ trùng thì sử dụng từ khóa ALL đi kèm như: UNION ALL, INTERSECT ALL, EXCEPT ALL.

Cú pháp:

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
UNION [ALL]

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

Query 16: Cho biết mã đề tài có giáo viên tham gia với họ là 'Nguyễn' hoặc người chủ nhiệm đề tài có họ là 'Nguyễn'.

Q16: SELECT MÃĐT

```
FROM GIÁOVIÊN GV, THAMGIAĐT TG
WHERE   GV.MÃGV=TG.MÃGV
        AND GV.HỌTÊN LIKE '%Nguyễn%'
UNION
SELECT MÃĐT
FROM GIÁOVIÊN GV1, ĐỀTÀI ĐT
WHERE   GV1.MÃGV= ĐT.GVCNĐT
        AND GV.HỌTÊN LIKE '%Nguyễn%'
```

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

INTERSECT [ALL]

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

Query 17: Tìm các mã giáo viên vừa làm trưởng bộ môn vừa tham gia chủ nhiệm đề tài.

Q17: SELECT TRUONGBM

```
FROM BỘMÔN
INTERSECT
SELECT GVCNĐT
```

FROM ĐỀ TÀI

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

EXCEPT [ALL]

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

Query 18: Cho biết mã giáo viên chưa tham gia đề tài nào cả.

Q18: SELECT MÃGV

FROM GIÁO VIÊN

EXCEPT

SELECT MÃGV

FROM THAMGIAĐT

VI. TRUY VẤN PHÚC TẠP TRONG SQL

Trong phần trước chúng tôi đã trình bày về truy vấn đơn giản trong SQL. Trong phần này chúng tôi sẽ trình bày thêm một số đặc tính khác của SQL để giúp người dùng xây dựng câu truy vấn phức tạp hơn.

VI.1 So sánh với giá trị NULL và UNKNOWN

Trong SQL, giá trị NULL được thể hiện dưới 3 nghĩa khác nhau:

- *Giá trị chưa biết (Unknown value)*: ví dụ một giáo viên thì có ngày sinh nhưng giá trị này chưa được biết, do vậy sẽ bị thể hiện thành giá trị NULL trong CSDL.
- *Giá trị bị che giấu (Unavailable/Withheld value)*: ví dụ một giáo viên có số điện thoại nhà, nhưng họ không muốn khai báo do vậy nó được thể hiện thành giá trị NULL trong CSDL.
- *Thuộc tính không áp dụng (Not applicable attribute)*: ví dụ như mỗi giáo viên có thuộc tính ngày nhận chức, tuy nhiên thuộc tính này không áp dụng cho tất cả các giáo viên, do vậy giá trị tại thuộc tính cho các giáo viên này sẽ là NULL.

Tuy nhiên, trong thực tế khó mà xác định được ý nghĩa rõ ràng của mỗi giá trị NULL. Ví dụ như với thuộc tính số điện thoại nhà chúng ta cũng có thể hiểu theo cả 3 nghĩa như trên và SQL cũng không phân biệt được sự khác biệt giữa các giá trị NULL được đề cập ở trên. Do vậy giá trị NULL khi được đưa vào phép so sánh thì đều cho ra giá trị UNKNOWN (có thể TRUE hoặc FALSE). SQL sử dụng 3 giá trị logic là TRUE, FALSE và UNKNOWN chứ không phải sử dụng 2 giá trị TRUE và FALSE. Vì thế chúng ta cần xem xét bảng kết nối logic cho 3 giá trị này với các phép nối AND, OR và NOT được thể hiện ở Hình VI-13.

AND	TRUE	FALSE	UNKNOWN
TRUE	TRUE	FALSE	UNKNOWN
FALSE	FALSE	FALSE	FALSE
UNKNOWN	UNKNOWN	FALSE	UNKNOWN

OR	TRUE	FALSE	UNKNOWN
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	UNKNOWN
UNKNOWN	TRUE	UNKNOWN	UNKNOWN

NOT	
TRUE	FALSE
FALSE	TRUE
UNKNOWN	UNKNOWN

Hình VI-13 Bảng kết nối logic giữa 3 giá trị TRUE, FALSE, UNKNOWN.

Trong câu truy vấn, chỉ bộ nào làm cho giá trị của biểu thức logic ở mệnh đề WHERE là TRUE mới xuất hiện ở quan hệ kết quả; các bộ làm giá trị của biểu thức logic ở mệnh đề WHERE là FALSE hoặc UNKNOWN thì chúng không thỏa mãn điều kiện truy vấn nên không

có trong kết quả trả về. Tuy nhiên, điều này không đúng cho truy vấn có phép kết ngoài (*OUTER JOIN*).

SQL cho phép chúng ta kiểm tra giá trị của một thuộc tính có bằng NULL hay không bằng cách sử dụng từ khóa IS hoặc IS NOT thay vì chỉ sử dụng phép so sánh = hoặc \neq , ví dụ được thể hiện ở **Q19** và **Q20**.

Query 19: Cho biết mã giáo viên, họ tên giáo viên chưa có người quản lý chuyên môn.

Q19: SELECT MÃGV, HỌTÊN

FROM GIÁOVIÊN
WHERE GVQLCM IS NULL

Query 20: Cho biết mã bộ môn, tên bộ môn của các bộ môn đã có giáo viên làm trưởng bộ môn.

Q20: SELECT MÃBM, TÊNBM

FROM BỘMÔN
WHERE TRƯỞNGBM IS NOT NULL

VI.2 Truy vấn lồng

Truy vấn lồng là chúng ta sử dụng một khối lệnh SELECT – FROM – WHERE trong mệnh đề WHERE của một câu truy vấn khác. Khối lệnh SELECT – FROM – WHERE này được gọi là truy vấn con. SQL cho phép các câu lệnh truy vấn lồng nhau ở nhiều mức và kết quả của truy vấn con trả về thường là một tập các giá trị.

Cú pháp:

SELECT <danh sách các cột>
FROM <danh sách các bảng>
WHERE <so sánh tập hợp>

Truy vấn cha

SELECT <danh sách các cột>
FROM <danh sách các bảng>
WHERE <điều kiện>

Truy vấn con

Trong mệnh đề WHERE của truy vấn cha: *WHERE <so sánh tập hợp> <truy vấn con>*, các toán tử thường được sử dụng trong so sánh tập hợp này là: IN, NOT IN, ALL, ANY, SOME, EXISTS và NOT EXISTS.

IN dùng để so sánh giá trị của thuộc tính với một tập hợp. Đối với IN, thuộc tính ở mệnh đề SELECT của truy vấn con phải có cùng kiểu với dữ liệu với thuộc tính ở mệnh đề WHERE của truy vấn cha.

EXISTS dùng để kiểm tra một câu truy vấn bất kỳ có trả về bộ dữ liệu nào hay không. Nếu câu truy vấn tham số của EXISTS có trả về ít nhất một bộ dữ liệu thì EXISTS trả về giá trị TRUE. Ngược lại, khi câu truy vấn không trả về bộ dữ liệu nào thì EXISTS sẽ trả về giá trị FALSE. Đối với EXISTS không có thuộc tính, hằng số hay biểu thức nào đứng ở vé trái (phía trước EXISTS), câu truy vấn tham số không cần phải liệt kê bất kỳ thuộc tính nào.

Những câu truy vấn có sử dụng “= ANY” hoặc IN đều có thể chuyển về câu truy vấn sử dụng EXISTS.

Truy vấn lồng được chia làm 2 loại: lồng phân cấp và lồng tương quan, các truy vấn này sẽ được trình bày chi tiết ở phần kế tiếp.

VI.2.1 Truy vấn lồng phân cấp

Truy vấn lồng phân cấp là truy vấn lồng mà mệnh đề WHERE của truy vấn con không tham chiếu thuộc tính nào cả của các quan hệ trong mệnh đề FROM của truy vấn cha. Đối với loại truy vấn này, khi thực hiện truy vấn con sẽ được thực hiện một lần và thực hiện trước truy vấn cha.

Query 3b: Cho biết mã giáo viên, họ tên giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q3b:

```
SELECT MÃGV, HỌTÊN  
FROM GIÁOVIÊN  
WHERE MÃBM IN (SELECT MÃBM  
                FROM BỘMÔN  
                WHERE TÊNBM = 'Hệ thống thông tin')
```

Query 18a: Cho biết mã giáo viên, họ tên giáo chưa tham gia đề tài nào.

Q18a:

```
SELECT MÃGV, HỌTÊN  
FROM GIÁOVIÊN  
WHERE MÃGV NOT IN (SELECT MÃGV  
                    FROM THAMGIAĐT)
```

Query 21: Tìm những giáo viên có lương lớn hơn lương của ít nhất một giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q21:

```
SELECT *  
FROM GIÁOVIÊN  
WHERE LUƠNG > ANY (SELECT LUƠNG  
                     FROM GIÁOVIÊN GV, BỘMÔN BM  
                     WHERE GV.MÃBM = BM.MÃBM AND  
                           BM.TÊNBM = 'Hệ thống thông tin')
```

Query 22: Tìm những giáo viên có lương lớn hơn lương của tất cả giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q22:

```
SELECT *  
FROM GIÁOVIÊN  
WHERE LUONG > ALL (SELECT LUONG  
FROM GIÁOVIÊN GV, BỘMÔN BM  
WHERE GV.MÃBM = BM.MÃBM AND  
BM.TÊNBM = 'Hệ thống thông tin')
```

Query 23: Tìm những trưởng bộ môn có tham gia ít nhất một đề tài.

Q23:

```
SELECT *  
FROM GIÁOVIÊN  
WHERE MÃGV IN (SELECT TRƯỞNGBM FROM BỘMÔN)  
AND MÃGV IN (SELECT MÃGV FROM THAMGIAĐT)
```

VI.2.2 Truy vấn lồng tương quan

Truy vấn lồng tương quan là truy vấn lồng mà mệnh đề WHERE của câu truy vấn con tham chiếu ít nhất một thuộc tính của các quan hệ trong mệnh đề FROM của truy vấn cha. Đối với loại truy vấn này khi thực hiện truy vấn con sẽ được thực hiện nhiều lần, mỗi lần tương ứng với một bộ của truy vấn cha.

Query 3c: Cho biết mã giáo viên, họ tên giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q3c:

```
SELECT MÃGV, HỌTÊN  
FROM GIÁOVIÊN GV  
WHERE EXISTS (SELECT *
```

```
FROM BỘMÔN BM
WHERE TÊNBM = 'Hệ thống thông tin'
AND BM.MÃBM = GV.MÃBM)
```

Query 18b: Cho biết mã giáo viên, họ tên giáo chưa tham gia đề tài nào.

Q18b:

```
SELECT MÃGV, HỌTÊN
FROM GIÁOVIÊN GV
WHERE NOT EXISTS (SELECT *
                   FROM THAMGIAĐT TG
                   WHERE GV.MÃGV = TG.MÃGV)
```

Query 21a: Tìm những giáo viên có lương lớn hơn lương của ít nhất một giáo viên thuộc bộ môn 'Hệ thống thông tin'.

Q21a:

```
SELECT *
FROM GIÁOVIÊN GV1
WHERE EXISTS (SELECT *
              FROM GIÁOVIÊN GV2, BỘMÔN BM
              WHERE GV2.MÃBM = BM.MÃBM
              AND BM.TÊNBM = 'Hệ thống thông tin'
              AND GV2.LƯƠNG < GV1.LƯƠNG)
```

Query 23a: Tìm những trưởng bộ môn có tham gia ít nhất một đề tài.

Q23a:

```
SELECT *
FROM GIÁOVIÊN GV
WHERE EXISTS (SELECT *
```

```
FROM BỘMÔN BM
WHERE BM.TRUỐNGBM = GV.MÃGV)
AND EXISTS (SELECT *
FROM THAMGIAĐT TG
WHERE TG.MÃGV = GV.MÃGV)
```

Trong SQL chúng ta sử dụng truy vấn lòng trong quan để biểu diễn cho phép chia được trình bày trong chương ngôn ngữ đại số quan hệ. Chúng ta có thể sử dụng *EXCEPT* hay *NOT EXISTS* để biểu diễn cho các phép chia.

Ví dụ : Tìm họ tên giáo viên tham gia tất cả các đề tài.

```
SELECT G.HOTÊN
FROM GIÁOVIÊN G
WHERE NOT EXISTS ((SELECT D.MÃĐT
FROM ĐỀTÀI D)
EXCEPT
(SELECT T.MÃĐT
FROM THAMGIAĐT T
WHERE T.MÃGV = G.MÃGV))
```

Với ví dụ trên chúng ta có thể sử dụng *NOT EXISTS* thay vì sử dụng *EXCEPT*.

```
SELECT G.HOTÊN
FROM GIÁOVIÊN G
WHERE NOT EXISTS (SELECT D.MÃĐT
FROM ĐỀTÀI D
WHERE NOT EXISTS
(SELECT T.MÃĐT
FROM THAMGIAĐT T
```

WHERE T.MÃGV = G.MÃGV

AND T.MÃĐT = D.MÃĐT))

VI.3 Sử dụng phép kết – JOIN ở mệnh đề FROM

SQL cho phép chúng ta khai báo phép kết giữa các bảng ở mệnh đề FROM thay vì chỉ khai báo ở mệnh đề WHERE. Phép kết ở mệnh đề FROM gồm có kết bằng (*INNER JOIN*), kết trái (*LEFT OUTER JOIN*), kết phải (*RIGHT OUTER JOIN*) và kết ngoài (*FULL OUTER JOIN*). SQL cho phép chúng ta có thể lược bỏ từ khóa *INNER* và *OUTER* khi khai báo.

Cú pháp:

SELECT <danh sách các cột>
FROM R1 [**INNER**] **JOIN** R2 **ON** <biểu thức>
WHERE <điều kiện>

SELECT <danh sách các cột>
FROM R1 **LEFT|RIGHT|OUTER JOIN** R2 **ON** <biểu thức>
WHERE <điều kiện>

Query 3d: Cho biết mã giáo viên, họ tên giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q3d:

SELECT MÃGV, HỌTÊN
FROM GIÁOVIÊN GV INNER JOIN BỘMÔN BM
ON BM.MÃBM = GV.MÃBM
WHERE BM. TÊNBM = ‘Hệ thống thông tin’

Query 4b: Cho biết mã giáo viên, họ tên giáo viên và họ tên người quản lý chuyên môn giáo viên đó *nếu có*.

Q4b:

```
SELECT GV.MÃGV, GV.HỌTÊN, QLCM.HỌTÊN
FROM GIÁOVIÊN GV LEFT OUTER JOIN GIÁOVIÊN QLCM
ON GV.GVQLCM = QLCM.MÃGV
```

Lưu ý: Kết quả của truy vấn sẽ khác **Q4b** kết quả của truy vấn **Q4** và **Q4a**. Giáo viên không có người quản lý chuyên môn (giá trị tại thuộc tính GVQLCM của GIÁOVIÊN là NULL) vẫn được trả về và họ tên người quản lý chuyên môn của họ (QLCM.HỌTÊN) là rỗng (*NULL*).

Query 24: Cho biết mã giáo viên, họ tên giáo viên và tên đê tài của họ tham gia *nếu có*.

Q24:

```
SELECT GV.MÃGV, GV.HỌTÊN, ĐT.TÊNĐT
FROM GIÁOVIÊN GV LEFT JOIN (THAMGIAĐT TG JOIN
ĐỀTÀI ĐT ON TG.MÃĐT = ĐT.MÃĐT) ON GV.MÃGV =
TG.MÃĐT
```

VI.4 Hàm tính tổng hợp trong SQL

Trong chương 5, chúng tôi đã giới thiệu khái niệm về hàm tính tổng hợp thông qua một phép toán quan hệ. Vì gom nhóm và tính tổng hợp được dùng trong nhiều ứng dụng CSDL, nên SQL xây dựng các hàm để phục vụ cho nhu cầu tính toán này. Một số hàm được xây dựng bao gồm: COUNT, SUM, MAX, MIN, AVG. Hàm COUNT trả về số lượng bộ hoặc số lượng giá trị thỏa điều kiện truy vấn. Hàm SUM, MIN, MAX, AVG được áp dụng trên trường kiểu số và kết quả trả về tương ứng là tính tổng, lấy giá trị nhỏ nhất, lấy giá trị lớn nhất và tính giá trị trung bình. Các hàm này có thể được sử dụng trong mệnh đề SELECT hoặc mệnh đề HAVING (được giới thiệu ở phần sau). Sau đây chúng tôi trình bày một số ví dụ cho các hàm này.

Query 25: Tìm tổng lương, lương lớn nhất, lương nhỏ nhất và lương trung bình của toàn bộ giáo viên.

Q25:

```
SELECT SUM(LƯƠNG), MAX(LƯƠNG), MIN(LƯƠNG),  
       AVG(LƯƠNG)  
FROM GIÁOVIÊN
```

Query 26: Tìm tổng lương, lương lớn nhất, lương nhỏ nhất và lương trung bình của giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

Q26:

```
SELECT SUM(GV.LƯƠNG), MAX(GV.LƯƠNG),  
       MIN(GV.LƯƠNG), AVG(GV.LƯƠNG)  
FROM GIÁOVIÊN GV, BỘMÔN BM  
WHERE GV.MÃBM AND BM.MÃBM AND BM.TÊNBM = 'Hệ  
thống thông tin'
```

Query 27: Cho biết số lượng giáo viên.

Q27:

```
SELECT COUNT(*)  
FROM GIÁOVIÊN
```

Query 28: Cho biết số lượng giáo viên ở bộ môn ‘Hệ thống thông tin’.

Q28:

```
SELECT COUNT(*)  
FROM GIÁOVIÊN GV, BỘMÔN BM  
WHERE GV.MÃBM AND BM.MÃBM AND BM.TÊNBM = 'Hệ  
thống thông tin'
```

Query 29: Cho biết số lượng các mức lương của giáo viên (không tính giá trị lương bị trùng).

Q29:

```
SELECT COUNT (DISTINCT LUƠNG)  
FROM GIÁOVIÊN GV
```

VI.5 Mệnh đề GROUP BY và HAVING

Trong một số trường hợp, chúng ta cần áp dụng hàm tính tổng hợp cho các nhóm nhỏ các bộ trong quan hệ. Các nhóm nhỏ các bộ này được hình thành dựa trên một/ một số thuộc tính dùng để gom nhóm. Ví dụ khi muốn tìm lương trung bình của các giáo viên ở mỗi bộ môn, chúng ta cần chia quan hệ GIÁO VIÊN thành các nhóm nhỏ, mỗi nhóm gồm các bộ có giá trị giống nhau trên thuộc tính MÃBM, sau đó áp dụng trên mỗi nhóm hàm AVG trên thuộc tính LUONG. Thuộc tính mà ta căn cứ vào đó để tạo nhóm được gọi là thuộc tính gom nhóm. Trong ví dụ trên, MÃBM được gọi là thuộc tính gom nhóm. SQL đưa ra mệnh đề GROUP BY để giải quyết vấn đề này. Trong mệnh đề GROUP BY, chúng ta phải chỉ ra các thuộc tính gom nhóm. Trong một câu SQL, những thuộc tính xuất hiện sau SELECT phải xuất hiện sau mệnh đề GROUP BY (trừ các hàm tính toán).

Query 30: Cho biết lương trung bình của giáo viên ở từng bộ môn.

Q30:

```
SELECT BM.MÃBM, BM.TÊNBM,  
       AVG (LUONG) AS LUONGTB  
  FROM GIÁO VIÊN GV, BỘ MÔN BM  
 WHERE GV.MÃBM = BM.MÃBM  
 GROUP BY BM.MÃBM, BM.TÊNBM
```

Query 31: Cho biết số lượng giáo viên tham gia cho mỗi đề tài.

Q31:

```
SELECT ĐT.MÃĐT, ĐT.TÊNĐT, COUNT (*) AS SLGV  
  FROM ĐỀ TÀI ĐT, THAM GIA ĐT TG  
 WHERE ĐT.MÃĐT = TG.MÃĐT  
 GROUP BY ĐT.MÃĐT, ĐT.TÊNĐT
```

Thực tế, đôi khi chúng ta cần rút trích những bộ dữ liệu thỏa một số điều kiện mà các biểu thức điều kiện này được thành lập dựa trên kết

quả của các hàm tính toán trên CSDL. Ví dụ cho biết bộ môn nào có lương trung bình lớn hơn 2 triệu. Để tìm được dữ liệu thỏa điều kiện trên, trước tiên, ta phải tính lương trung bình của từng bộ môn, sau đó chọn ra các bộ môn có lương trung bình lớn hơn 2 triệu. Chúng ta phải dùng mệnh đề GROUP BY để gom nhóm dữ liệu của quan hệ GIÁO VIÊN với thuộc tính gom nhóm là MÃ BM, và áp dụng hàm tính trung bình (AVG) trên thuộc tính LUONG đối với từng nhóm. Kế đến ta dùng mệnh đề HAVING để thể hiện điều kiện chọn dựa trên giá trị lương trung bình vừa tính được.

Query 32: Cho biết lương bộ môn nào có lương trung bình hơn 2 triệu.

Q32:

```
SELECT BM.MÃBM, BM.TÊNBM, AVG (LUONG) AS  
LUONGTB  
FROM GIÁO VIÊN GV, BỘ MÔN BM  
WHERE GV.MÃBM = BM.MÃBM  
GROUP BY BM.MÃBM, BM.TÊNBM  
HAVING AVG(LUONG) > 2000000
```

Query 33: Cho biết đề tài nào có số lượng giáo viên tham gia trên 7 giáo viên.

Q33:

```
SELECT ĐT.MÃĐT, ĐT.TÊNĐT, COUNT (*) AS SLGV  
FROM ĐỀ TÀI ĐT, THAM GIA ĐT TG  
WHERE ĐT.MÃĐT = TG.MÃĐT  
GROUP BY ĐT.MÃĐT, ĐT.TÊNĐT  
HAVING COUNT(*) > 7
```

VI.6 Kết luận về truy vấn trong SQL

Một câu truy vấn viết bằng SQL có thể có 6 mệnh đề được thể hiện như cú pháp bên dưới, nhưng chỉ có 2 mệnh đề SELECT và FROM là bắt buộc. Các mệnh đề còn lại là tùy chọn dựa vào nhu cầu truy vấn

của người dùng. Thứ tự thực hiện của câu truy vấn như sau: FROM → WHERE → GROUP BY → HAVING → SELECT → ORDER BY.

Cú pháp của câu truy vấn đầy đủ:

SELECT <danh sách các cột>

FROM <danh sách các bảng>

[**WHERE** <điều kiện>]

[**GROUP BY** <các thuộc tính gom nhóm>]

[**HAVING** <điều kiện trên nhóm>]

[**ORDER BY** <các thuộc tính sắp thứ tự>]

VII. CÁC LỆNH CẬP NHẬT DỮ LIỆU

SQL cung cấp cho chúng ta các lệnh cập nhật bao gồm: INSERT, DELETE, và UPDATE. Lệnh INSERT được dùng để thêm các dòng dữ liệu vào một bảng. Lệnh DELETE được dùng để xóa các dòng dữ liệu ra khỏi bảng. Lệnh UPDATE được dùng để sửa các dòng dữ liệu trong bảng. Trong quá trình thêm, xóa, sửa dữ liệu chúng ta cần phải cẩn thận để tránh trường hợp vi phạm ràng buộc toàn vẹn.

VII.1 Lệnh thêm dữ liệu - INSERT

Lệnh INSERT được dùng để thêm các dòng dữ liệu vào bảng. Nó cho phép chúng ta thêm một hoặc nhiều dòng dữ liệu vào bảng dữ liệu tùy theo cú pháp được sử dụng.

- Lệnh thêm một bộ dữ liệu vào bảng:

INSERT INTO <tên bảng>(<danh sách các thuộc tính>)

VALUES (<danh sách các giá trị>)

Trong cú pháp trên, để thêm dòng dữ liệu vào bảng chúng ta cần phải chỉ định bảng dữ liệu được thêm vào, danh sách thuộc tính cần thêm dữ liệu và danh sách các giá trị tương ứng cho các thuộc tính.

U1: Thêm bộ dữ liệu vào bảng KHOA.

```
INSERT INTO KHOA(MÃKHOA, TÊNKHOA, NĂMTL, PHÒNG,  
ĐIỆNTHOAI, TRƯỞNGKHOA, NGÀYNHÂNCHỨC) VALUES  
(‘ĐC’, ‘Địa chất’, 1980, ‘B51’, ‘0838324467’, ‘010’, ‘20/07/1990’)
```

Lưu ý: Nếu chúng ta thêm toàn bộ giá trị cho toàn bộ thuộc tính của bảng KHOA thì chúng ta không cần phải liệt kê tất cả thuộc tính như sau..

```
INSERT INTO KHOA VALUES (‘ĐC’, ‘Địa chất’, 1980, ‘B51’,  
‘0838324467’, ‘010’, ‘20/07/1990’)
```

U2: Thêm vào bảng giáo viên dòng dữ liệu {‘011’, ‘Nguyễn Thị Nhơn’, 2000, ‘Nữ’, NULL} tương ứng với các thuộc tính { MÃGV, HỌTÊN, LUONG, PHÁI, NGÀYSINH }

```
INSERT INTO GIÁOVIÊN (MÃGV, HỌTÊN, LUONG, PHÁI,  
NGÀYSINH) VALUES (‘011’, ‘Nguyễn Thị Nhơn’, 2000, ‘Nữ’,  
NULL)
```

- Lệnh thêm vào nhiều dòng dữ liệu:

INSERT INTO <tên bảng>(<danh sách các thuộc tính>)

<câu truy vấn con>

Trong cú pháp trên, câu truy vấn con sẽ thực hiện lấy dữ liệu và số dòng dữ liệu được trả về cho câu truy vấn con sẽ được đưa vào bảng dữ liệu.

U3: Tạo bảng dữ liệu THỐNGKÊ gồm có thuộc tính: tên bộ môn (TÊNBM), số lượng giáo viên (SLGV), lương trung bình (LUONGTB). Sau đó đưa dữ liệu vào bảng THỐNG KÊ bằng cách dùng truy vấn con để tính toán từ bảng dữ liệu GIÁO VIÊN và BỘ MÔN.

```
CREATE TABLE THỐNGKÊ (
```

```
    TÊNBM      VARCHAR(30),
```

```
    SLGV      INT,  
    LUONGTB  INT  
)
```

```
INSERT INTO THÔNGKÊ(TÊNBM, SLGV, LUONGTB)  
SELECT BM.TÊNBM, COUNT(MÃGV), AVG(LƯƠNG)  
FROM GIÁOVIÊN GV, BỘMÔN BM  
WHERE GV.MÃBM= BM.MÃBM  
GROUP BY BM.TÊNBM
```

VII.2 Lệnh xóa dữ liệu – DELETE

SQL cho phép chúng ta dùng lệnh DELETE để xóa các dòng dữ liệu trong một bảng dữ liệu.

Cú pháp:

DELETE FROM <tên bảng>

[WHERE <điều kiện>]

U4: Xóa các giáo viên có họ là ‘Nguyễn’.

DELETE FROM GIÁOVIÊN

WHERE HỌTÊN LIKE ‘%Nguyễn%’

U5: Xóa toàn bộ giáo viên có trong bảng giáo viên.

DELETE FROM GIÁOVIÊN

U6: Xóa toàn bộ giáo viên thuộc bộ môn ‘Hệ thống thông tin’.

DELETE FROM GIÁOVIÊN

```
WHERE MÃBM IN  ( SELECT MÃBM  
                  FROM BỘMÔN  
                  WHERE TÊNBM = ‘Hệ thống thông tin’)
```

VII.3 Lệnh sửa dữ liệu - UPDATE

SQL cho phép sử dụng lệnh UPDATE để sửa các dòng dữ liệu trong một bảng dữ liệu.

Cú pháp

UPDATE <tên bảng>

SET <tên thuộc tính>=<giá trị mới>,

<tên thuộc tính>=<giá trị mới>,

...

[WHERE <điều kiện>]

U7: Tăng toàn bộ lương của giáo viên lên 10%.

UPDATE GIÁO VIÊN

SET LUONG = LUONG*1.1

U7: Cập nhật ngày sinh của giáo viên ‘Nguyễn Hoài An’ là ‘15/02/1975’ và lương là 2500.

UPDATE GIÁO VIÊN

SET NGAYSINH = ‘15/02/1975’, LUONG = 2500

WHERE HOTEN = ‘Nguyễn Hoài An’

VIII. KẾT LUẬN

Trong chương 6, chúng tôi đã trình bày ngôn ngữ SQL, gồm có: lệnh tạo lược đồ, các lệnh tạo ràng buộc, các lệnh rút trích dữ liệu từ cơ bản đến phức tạp cũng như các đặc tính bổ sung mới được bổ sung cho các lệnh truy vấn, các lệnh cập nhật dữ liệu. Trong chương kế tiếp chúng tôi sẽ trình bày về một ngôn ngữ khác dùng để thể hiện câu truy vấn trên CSDL quan hệ, ngôn ngữ phép tính quan hệ.

Câu hỏi ôn tập

1. Sự khác biệt giữa khái niệm bảng dữ liệu trong SQL so với định nghĩa về quan hệ trong lược đồ cơ sở dữ liệu quan hệ là gì? Tại sao

SQL cho phép các bộ trong một bảng hay kết quả của câu truy vấn có thể trùng nhau?

2. Cho biết các loại kiểu dữ liệu được sử dụng để khai báo cho các thuộc tính trong ngôn ngữ SQL.
3. SQL cung cấp các hàm tính toán nào để thực hiện trên một tập các bộ dữ liệu? Cách sử dụng chúng như thế nào trong quá trình viết câu truy vấn?
4. Truy vấn lồng là gì? Thế nào là lồng phân cấp và lồng tương quan? So sánh cách thực thi của hai loại truy vấn này thông qua ví dụ minh họa.
5. Các toán tử IN, EXISTS, UNIQUE, ANY, và ALL được sử dụng như thế nào trong quá trình viết truy vấn lồng? Tại sao chúng lại hữu ích? Minh họa câu trả lời của bạn thông qua việc thực hiện phép chia trong SQL.
6. Hãy phân biệt sự khác nhau khi sử dụng mệnh đề WHERE và HAVING trong một câu truy vấn SQL. Hãy cho ví dụ minh họa để giải thích cho câu trả lời.
7. Giá trị rỗng (NULL) là gì? Chúng ảnh hưởng như thế nào đến ngữ nghĩa của câu truy vấn? Có cho phép các thuộc tính tham gia làm khóa chính chứa giá trị rỗng không?
8. Hãy cho biết các từ khóa nào được sử dụng, cách khai báo của chúng trong cú pháp lệnh của ngôn ngữ SQL (cho ví dụ minh họa) khi khai báo cho các loại ràng buộc sau:
 - a. Khóa chính.
 - b. Khóa ngoại.
 - c. Ràng buộc giá trị cho một thuộc tính.
 - d. Ràng buộc duy nhất cho một hoặc một tập thuộc tính.
 - e. Ràng buộc giá trị mặc định cho một thuộc tính.
9. Trigger là gì? Cấu trúc khi khai báo một trigger gồm có những phần chính nào? Mục đích của trigger là để làm gì? Hãy giải thích sự khác biệt khi khai báo một ràng buộc toàn vẹn bắt buộc phải dùng kỹ thuật trigger so với các kỹ thuật khác của SQL?

Bài tập

1. Hãy dùng ngôn ngữ SQL để thể hiện những biểu thức đại số quan hệ sau:

- a. $\Pi_{A,B} (R(A, B, C))$
 - b. $\sigma_B (R(A, B))$
 - c. $R(A, B) \times S (C, D)$
 - d. $R(A, B) * S (B, C)$
 - e. $R(A, B) \bowtie S(B, C)$
 - f. $R(A, B) \div S (B)$
 - g. $\rho_{C, D} (R(A, B))$
 - h. $\exists \text{COUNT}^* (R(A, B))$
 - i. $A \exists \text{COUNT}^* (R(A, B))$
 - j. $\sigma \text{COUNT}^* \theta c (A \exists \text{COUNT}^* (R(A, B)))$ với θ là một phép so sánh, c là hằng số.
2. Hãy thực hiện các nhu cầu truy vấn được mô tả ở bài tập 1, 2, 3 của chương 5 bằng ngôn ngữ SQL.
 3. Cho lược đồ Ứng dụng Quản lý đề tài nghiên cứu khoa học như trong bài tập 1 chương 5. Hãy trả lời các câu hỏi sau bằng ngôn ngữ SQL:
 - a. Hãy cho biết thông tin giáo viên (MAGV, HOTEN) làm trưởng bộ môn hoặc trưởng khoa.
 - b. Hãy cho biết thông tin giáo viên nào trùng ngày sinh với người quản lý chuyên môn của mình. Kết quả cho ra thông tin: mã giáo viên, tên giáo viên, mã giáo viên quản lý, tên giáo viên quản lý.
 - c. Cho biết mã, tên các giáo viên thuộc bộ môn Hệ thống thông tin và chưa từng tham gia đề tài nào.
 - d. Cho biết khoa nào (tên khoa) có nhiều bộ môn trực thuộc nhất.
 - e. Với mỗi đề tài, cho biết giáo viên nào tham gia đề tài đó với tổng phụ cấp lớn nhất. Thông tin xuất ra gồm có tên đề tài, tên giáo viên, tổng tiền phụ cấp.
 - f. Cho biết trưởng khoa (tên trưởng khoa) có tuổi nhỏ nhất trong tất cả các trưởng khoa.
 - g. Cho biết giáo viên nào (mã giáo viên, tên giáo viên) tham gia đã tham gia tất cả các công việc thuộc một đề tài bất kỳ.
 - h. Cho biết đề tài nào (mã đề tài, tên đề tài) thuộc cấp quản lý 'Quốc gia' được tất cả các giáo viên thuộc bộ môn hệ thống thông tin tham gia thực hiện.
 - i. Hãy cho biết giáo viên (mã giáo viên, tên giáo viên) nào tham gia thực hiện nhiều đề tài nhất trong từng bộ môn của họ.

7

Chương 7

PHÉP TÍNH QUAN HỆ

I. GIỚI THIỆU

Mặc dù đại số quan hệ rất có ích trong việc biểu diễn cách thức thực hiện một câu truy vấn nhưng SQL lại dựa trên một ngôn ngữ khác, đó là phép tính quan hệ.

Phép tính quan hệ (relational calculus) là một ngôn ngữ hình thức do Codd đề nghị năm 1972 dựa trên lý thuyết logic.

Phép tính quan hệ là ngôn ngữ phi thủ tục, chỉ nêu kết quả cần tìm chứ không thể hiện cách thức tìm kiếm dữ liệu. Có hai loại phép tính quan hệ: *phép tính quan hệ có biến là bộ (tuple relational calculus)* và *phép tính quan hệ có biến là miền (domain relational calculus)*. Phép tính quan hệ có biến là bộ thể hiện một câu truy vấn dưới dạng một biểu thức dựa trên biến là dòng (bộ) của quan hệ. Phép tính quan hệ có biến là miền thể hiện một câu truy vấn dưới dạng một biểu thức dựa trên biến là cột (thuộc tính) của quan hệ.

Khả năng thể hiện nhu cầu truy vấn của ngôn ngữ phép tính quan hệ tương đương như đại số quan hệ, nghĩa là bất cứ nhu cầu truy vấn nào nếu thể hiện được qua một biểu thức đại số quan hệ thì cũng thể hiện được bằng phép tính quan hệ.

Ngôn ngữ SQL và QBE (*Query By Example*) là hai ngôn ngữ rất thân thiện với người dùng trong các HQT CSDL quan hệ. SQL dựa trên phép tính quan hệ có biến là bộ trong khi QBE dựa trên phép tính quan hệ có biến là miền.

II. PHÉP TÍNH QUAN HỆ CÓ BIẾN LÀ BỘ

II.1 Biến bộ và quan hệ miền

Phép tính quan hệ có biến là bộ còn được gọi là *phép tính bộ (tuple calculus)*, dựa trên các *biến bộ (tuple variable)*, mỗi biến bộ có miền giá trị là một quan hệ của CSDL, nghĩa là có thể mang giá trị là bất cứ dòng nào trong quan hệ đó. Một phép tính quan hệ đơn giản có dạng:

$$\{t \mid E(t)\}$$

Trong đó t là một biến bộ và $E(t)$ là một biểu thức điều kiện dựa trên t . Kết quả của câu truy vấn trên là tập hợp các bộ t thỏa mãn $E(t)$.

Với một biến bộ t , và *quan hệ miền (range relation)* r của t , được thể hiện bởi một điều kiện có dạng $r(t)$.

Mỗi điều kiện là một sự kết hợp giữa các biến bộ. Ta biết rằng các biến bộ có thể nhận lấy giá trị là các dòng của quan hệ miền tương ứng. Kết quả trả về của một biểu thức phép tính quan hệ là các dòng làm cho điều kiện đạt giá trị TRUE.

Với mỗi sự kết hợp làm cho điều kiện mang giá trị TRUE, tất cả các thuộc tính hoặc chỉ có giá trị các thuộc tính mà người dùng cần tìm được trả về.

Ví dụ: Cho danh sách các giáo viên nữ.

$$\{t \mid \text{GIÁO VIÊN}(t) \wedge t.\text{PHÁI} = \text{'Nữ'}\}$$

Kết quả trả về từ câu truy vấn trên là một quan hệ có số thuộc tính bằng số thuộc tính của quan hệ GIÁO VIÊN ban đầu, có số dòng là các dòng trong quan hệ GIÁO VIÊN thỏa điều kiện PHÁI là nữ.

Ví dụ: Danh biết họ tên các giáo viên nữ.

$$\{t.\text{HỌ TÊN} \mid \text{GIÁO VIÊN}(t) \wedge t.\text{PHÁI} = \text{'Nữ'}\}$$

Kết quả của câu truy vấn trên là một quan hệ có một thuộc tính là HỌ TÊN, có số dòng là số dòng của quan hệ GIÁO VIÊN thỏa điều kiện PHÁI là nữ.

II.2 Biểu thức và công thức

Biểu thức tổng quát của phép tính quan hệ có dạng:

$$\{t_1.A_j, t_2.A_k, \dots, t_n.A_m \mid E (t_1, t_2, \dots, t_n, t_{n+1}, \dots, t_{n+m})\}$$

Trong đó $t_1, t_2, \dots, t_n, t_{n+1}, \dots, t_{n+m}$ là các biến bộ, từng A_i là các thuộc tính của các quan hệ miền của t_i , E là một điều kiện hay công thức có biến là bộ.

Một công thức được hình thành từ các công thức nguyên tố. Một công thức nguyên tố có dạng:

1. $r(t_i)$ trong đó r là tên của quan hệ và t_i là biến bộ. Công thức nguyên tố này cho biết miền của biến bộ t_i là quan hệ r .
2. $t_i.A \text{ op } t_j.B$ trong đó op là một trong các toán tử so sánh thuộc tập hợp $\{=, <, \leq, >, \geq, \neq\}$; t_i, t_j là các biến bộ; A, B là các thuộc tính tương ứng của quan hệ miền của t_i, t_j .
3. $t_i.A \text{ op } c$ hoặc $c \text{ op } t_j.B$ trong đó op là một trong các toán tử so sánh thuộc tập hợp $\{=, <, \leq, >, \geq, \neq\}$; t_i, t_j là các biến bộ; A, B là các thuộc tính tương ứng của quan hệ miền của t_i, t_j ; c là một giá trị hằng.

Mỗi công thức nguyên tố có thể mang giá trị là TRUE hay FALSE tùy thuộc vào giá trị mà các biến bộ nhận lấy, và được gọi là giá trị chân lý của một công thức nguyên tố.

Với công thức nguyên tố $r(t)$, nếu t được gán giá trị là một bộ của quan hệ r , thì giá trị của $r(t)$ là TRUE, ngược lại là FALSE. Công thức nguyên tố loại 2 hoặc 3 cũng mang giá trị TRUE nếu các bộ dữ liệu mà các biến bộ nhận lấy làm thỏa mãn điều kiện tương ứng.

Một công thức (hay điều kiện) hình thành từ một hoặc nhiều công thức nguyên tố nối với nhau bởi một/ nhiều toán tử AND (\wedge), OR (\vee), và NOT (\neg), và được định nghĩa như sau:

1. Mọi công thức nguyên tố đều là công thức.
2. Nếu F_1 và F_2 là công thức thì $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$, $\neg(F_1)$, $\neg(F_2)$ cũng là công thức. Giá trị chân lý của những công thức này phụ thuộc vào giá trị chân lý của các công thức thành phần:

- $(F1 \wedge F2)$ mang giá trị là TRUE nếu cả $F1$ và $F2$ đều mang giá trị TRUE, ngược lại, mang giá trị FALSE.
- $(F1 \vee F2)$ mang giá trị FALSE nếu cả $F1$ và $F2$ đều mang giá trị FALSE, ngược lại, mang giá trị TRUE.
- $\neg(F1)$ mang giá trị TRUE nếu $F1$ mang giá trị FALSE, mang giá trị FALSE nếu $F1$ mang giá trị TRUE.
- $\neg(F2)$ mang giá trị TRUE nếu $F2$ mang giá trị FALSE, mang giá trị FALSE nếu $F2$ mang giá trị TRUE.

II.3 Lượng từ *với mọi* (\forall) và *tồn tại* (\exists)

Có hai lượng từ đặc biệt có thể sử dụng trong một công thức là lượng từ *với mọi* (\forall) và *tồn tại* (\exists).

Nếu F là một công thức thì $(\exists t)(F)$ cũng là một công thức, t là một biến bộ. Công thức $(\exists t)(F)$ mang giá trị là TRUE nếu công thức F mang giá trị TRUE tại *ít nhất một* bộ nào đó mà biến tự do t có thể nhận lấy, ngược lại $(\exists t)(F)$ mang giá trị FALSE.

Nếu F là một công thức thì $(\forall t)(F)$ cũng là một công thức, t là một biến bộ. Công thức $(\forall t)(F)$ mang giá trị là TRUE nếu công thức F mang giá trị TRUE tại *mọi* bộ mà biến tự do t có thể nhận lấy, ngược lại $(\forall t)(F)$ mang giá trị FALSE.

Nhận xét:

- Lượng từ \exists được gọi là lượng từ “*tồn tại*” vì công thức $(\exists t)(F)$ mang giá trị TRUE nếu tồn tại một bộ nào đó làm cho F mang giá trị TRUE.
- Lượng từ \forall được gọi là lượng từ “*với mọi*” vì công thức $(\forall t)(F)$ mang giá trị TRUE nếu mọi bộ trong quan hệ miền của t đều làm cho F mang giá trị TRUE.

Biến đổi tương đương

Sau đây là một số biến đổi thường dùng giữa hai lượng từ *với mọi* và *tồn tại*. Ký hiệu \equiv nghĩa là tương đương với nhau.

- a) $(\forall x)(P(x)) \equiv \neg(\exists x)(\neg(P(x)))$
- b) $(\exists x)(P(x)) \equiv \neg(\forall x)(\neg(P(x)))$
- c) $(\forall x)(P(x) \wedge Q(x)) \equiv \neg(\exists x)(\neg(P(x)) \vee \neg(Q(x)))$
- d) $(\forall x)(P(x) \vee Q(x)) \equiv \neg(\exists x)(\neg(P(x)) \wedge \neg(Q(x)))$
- e) $(\exists x)(P(x) \vee Q(x)) \equiv \neg(\forall x)(\neg(P(x)) \wedge \neg(Q(x)))$
- f) $(\exists x)(P(x) \wedge Q(x)) \equiv \neg(\forall x)(\neg(P(x)) \vee \neg(Q(x)))$
- g) $(\forall x)(P(x)) \Rightarrow (\exists x)(P(x))$
- h) $\neg(\exists x)(P(x)) \Rightarrow \neg(\forall x)(P(x))$

II.4 Biến tự do và biến kết buộc

Có hai loại biến được dùng trong một công thức, đó là *biến tự do (free variable)* và *biến kết buộc (bound variable)*.

Một biến bộ t là biến kết buộc nếu t xuất hiện ở dạng $(\exists t)$ hoặc $(\forall t)$. Ngược lại, t là biến tự do. Ngoài ra, ta có thể dựa vào các nguyên tắc sau để định nghĩa một biến bộ là tự do hay kết buộc:

Nếu công thức F là một công thức nguyên tố thì biến bộ t xuất hiện trong F là một biến tự do.

Biến bộ t là tự do hay kết buộc trong một biểu thức logic có dạng $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$, $\neg(F_1)$ hoặc $\neg(F_2)$ là tùy thuộc vào biến t là tự do hay kết buộc trong F_1 hoặc F_2 . Tuy nhiên, trong công thức $F = (F_1 \wedge F_2)$ hoặc $F = (F_1 \vee F_2)$ một biến bộ có thể là tự do trong F_1 và kết buộc trong F_2 hoặc ngược lại.

Tất cả các biến tự do t trong F là biến kết buộc trong công thức $F' = (\exists t)(F)$ hoặc $F' = (\forall t)(F)$.

Ví dụ: Cho 3 công thức:

$F_1: k.TÊNKHOA = \text{'Công nghệ thông tin'}$

$F_2: (\exists b) (k.MÃKHOA = b.MÃKHOA)$

$F_3: (\forall k) (k.TRUÖNGKHOA = \text{'002'})$

Biến k là biến tự do trong hai công thức F_1 và F_2 , nhưng k là biến kết buộc trong F_3 . Biến b là biến kết buộc trong F_2 .

II.5 Ví dụ

Ví dụ 1: Cho biết địa chỉ và ngày sinh của giáo viên tên là 'Trần Trung Hiếu'.

{gv.SƠN HÀ, gv.ĐUỐNG, gv.QUẬN, gv.THÀNH PHÔ, gv.NGÀY SINH | GIÁO VIÊN(gv) \wedge gv.HỌ TÊN = 'Trần Trung Hiếu'}

Ví dụ 2: Tìm mã, họ tên, ngày sinh của các giáo viên thuộc bộ môn 'Hệ thống thông tin'.

{gv.MÃ GV, gv.HỌ TÊN, gv.NGÀY SINH | GIÁO VIÊN(gv) \wedge ($\exists b$ (BỘ MÔN(b) \wedge b.TÊN BM = 'Hệ thống thông tin' \wedge gv.MÃ BM = b.MÃ BM)}

Ví dụ 3: Đối với từng bộ môn thuộc khoa 'Công nghệ thông tin', cho biết mã bộ môn, tên bộ môn, tên của giáo viên làm trưởng bộ môn.

{bm.MÃ BM, bm.TÊN BM, gv.HỌ TÊN | BỘ MÔN(bm) \wedge GIÁO VIÊN(gv) \wedge bm.TRƯỞNG KHOA = gv.MÃ GV \wedge ($\exists k$ (KHOA(k) and k.TÊN KHOA = 'Công nghệ thông tin' \wedge k.MÃ KHOA = bm.MÃ KHOA)}

Ví dụ 4: Cho biết mã, họ tên giáo viên và họ tên người quản lý chuyên môn của từng giáo viên.

{gv1.MÃ GV, gv1.HỌ TÊN, gv2.HỌ TÊN | GIÁO VIÊN(gv1) \wedge GIÁO VIÊN(gv2) \wedge gv1.GVQL CM = gv2.MÃ GV}

Ví dụ 5: Cho danh sách giáo viên (MÃ GV, HỌ TÊN, NGÀY SINH) có tham gia đợt tài do Trần Trà Hương làm chủ nhiệm.

{gv.MÃ GV, gv.HỌ TÊN, gv.NGÀY SINH | GIÁO VIÊN(gv) \wedge ($\exists t$ ($\exists d$ ($\exists cn$ (THAM GIA ĐT(t) \wedge ĐỀ TÀI(d) \wedge GIÁO VIÊN(cn) \wedge gv.MÃ GV = t.MÃ GV \wedge t.MÃ ĐT = d.MÃ ĐT \wedge d.GVCN ĐT = cn.MÃ GV \wedge cn.TÊN GV = 'Trần Trà Hương')))}

Ví dụ 6: Cho danh sách giáo viên có tham gia tất cả các đề tài do giáo viên mã là 002 làm chủ nhiệm.

$\{gv.MÃGV, gv.HOTEN \mid GIÁOVIÊN(gv) \wedge ((\forall d) ĐỀTÀI(d) \\ (d.GVCNĐT = '002') \Rightarrow ((\exists t) THAMGIAĐT(t) \wedge t.MÃĐT = \\ d.MÃĐT \wedge t.MÃGV = gv.MÃGV))\}$

Ví dụ 7: Danh sách các giáo viên không có tham gia đề tài.

$\{gv.MÃGV, gv.HOTEN \mid GIÁOVIÊN(gv) \wedge (\text{NOT}(\exists t) \\ (THAMGIAĐT(t) \wedge t.MÃGV = gv.MÃGV))\}$

II.6 Biểu thức an toàn

Trong phép tính quan hệ, biểu thức an toàn là biểu thức có kết quả là một số xác định các dòng dữ liệu, ngược lại là biểu thức không an toàn.

Ví dụ: $\{gv \mid \neg(GIÁOVIÊN(gv))\}$ là một biểu thức không an toàn vì số bộ của quan hệ kết quả là không xác định.

Để hiểu rõ hơn về biểu thức an toàn, ta định nghĩa khái niệm miền của biểu thức.

Miền của biểu thức phép tính quan hệ có biến là bộ (domain of a tuple relational calculus expression) là tập hợp gồm giá trị hằng hoặc là các giá trị liên quan các bộ được sử dụng trong biểu thức đó.

Ví dụ: Miền của biểu thức $\{gv \mid GIÁOVIÊN(gv)\}$ là tập hợp giá trị các thuộc tính tại các dòng của quan hệ GIÁOVIÊN.

Một biểu thức là an toàn nếu tất cả các giá trị trong kết quả của biểu thức đều thuộc miền của biểu thức.

$\{gv \mid \neg(GIÁOVIÊN(gv))\}$ là một biểu thức không an toàn vì kết quả là các bộ không thuộc quan hệ GIÁOVIÊN nên các giá trị liên quan đến từng bộ này cũng không thuộc miền của biểu thức.

III. PHÉP TÍNH QUAN HỆ CÓ BIẾN LÀ MIỀN

III.1 Giới thiệu

Phép tính quan hệ biến là miền hay còn gọi là *phép tính miền (domain calculus)* là một dạng khác của phép tính quan hệ. Sự khác nhau của hai phép tính này là kiểu của biến được dùng trong một công thức. Thay vì cho các biến bộ biến thiên trên tập hợp các bộ thì biến miền biến thiên trên miền giá trị của thuộc tính tương ứng.

III.2 Biểu thức

Để tạo ra một quan hệ kết quả bậc n, ta phải có n biến miền, mỗi biến miền ứng với một thuộc tính. Biểu thức tổng quát của phép tính miền có dạng:

$$\{x_1, x_2, \dots, x_n \mid E(x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{n+m})\}$$

Trong đó $x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{n+m}$ là các biến miền biến thiên trên các miền của các thuộc tính tương ứng. E là một điều kiện hoặc công thức của phép tính miền.

Một công thức hình thành từ công thức nguyên tố. Công thức nguyên tố có dạng:

1. $R(x_1, x_2, \dots, x_j)$ trong đó R là tên của một quan hệ bậc j và từng $x_i, 1 \leq i \leq j$ là một biến miền. Công thức nguyên tố này cho biết rằng tập hợp các giá trị $\langle x_1, x_2, \dots, x_j \rangle$ là một bộ của quan hệ R , x_i là giá trị của thuộc tính thứ i của bộ này. Trong biểu thức của phép tính miền, người ta thường bỏ đi dấu phẩy giữa các biến miền, và viết là:

$$\{x_1, x_2, \dots, x_n \mid R(x_1 x_2 x_3) \wedge \dots\}$$

thay vì viết:

$$\{x_1, x_2, \dots, x_n \mid R(x_1, x_2, x_3) \wedge \dots\}$$

2. $x_i op x_j$, trong đó op là một phép so sánh thuộc tập hợp $\{=, <, \leq, >, \geq, \neq\}$ và x_i, x_j là các biến miền.
3. $x_i op c$ hoặc $c op x_j$ trong đó op là một phép so sánh thuộc tập hợp $\{=, <, \leq, >, \geq, \neq\}$; x_i, x_j là các biến miền; c là giá trị hằng.

Giống như trong phép tính bộ, các công thức nguyên tố có thể mang giá trị là TRUE hay FALSE tùy thuộc vào giá trị của các biến miền, và được gọi là giá trị chân lý của công thức nguyên tố.

Ví dụ: Trong công thức nguyên tố 1, nếu các biến miền được gán giá trị tạo thành 1 bộ trong quan hệ R thì công thức có giá trị TRUE. Công thức nguyên tố 2, 3 sẽ mang giá trị TRUE nếu các biến miền được gán giá trị thỏa điều kiện E.

Tương tự như phép tính bộ, phép tính miền có công thức cũng được thành lập từ các công thức nguyên tố, các biến miền và các lượng tử.

III.3 Ví dụ

Ví dụ 1: Cho biết địa chỉ và ngày sinh của giáo viên tên là 'Trần Trung Hiếu'.

$\{u, v, w, x, t \mid (\exists p)(\exists q)(\exists r)(\exists s)(\exists y)(\exists z)$

$\text{GIÁO VIÊN}(p, q, r, s, t, u, v, w, x, y, z) \wedge r = \text{'Trần Trung Hiếu'}$

Ví dụ 2: Tìm mã, họ tên, ngày sinh của các giáo viên thuộc bộ môn Hệ thống thông tin.

$\{p, q, t \mid (\exists j)(\exists z)(\exists h) \text{ GIÁO VIÊN}(p, q, r, s, t, u, v, w, x, y, z) \wedge$

$\text{BỘ MÔN}(i, j, k, l, m, n, o) \wedge j = \text{'Hệ thống thông tin'} \wedge z = i\}$

Ví dụ 3: Đối với từng bộ môn thuộc khoa 'Công nghệ thông tin', cho biết mã bộ môn, tên bộ môn, tên của giáo viên làm trưởng bộ môn.

$\{i, j, q \mid (\exists a)(\exists b)(\exists m)(\exists n)(\exists p) \text{ BỘ MÔN}(i, j, k, l, m, n, o) \wedge$

$\text{KHOA}(a, b, c, d, e, f, g) \wedge \text{GIÁO VIÊN}(p, q, r, s, t, u, v, w, x, y, z) \wedge b = \text{'Công nghệ thông tin'} \text{ and } n = a \text{ and } m = p\}$

Ví dụ 4: Cho biết mã, họ tên giáo viên và họ tên người quản lý chuyên môn của từng giáo viên.

$\{p, q, b \mid (\exists a)(\exists y) \text{ GIÁO VIÊN}(p, q, r, s, t, u, v, w, x, y, z) \wedge$

$\text{GIÁO VIÊN}(a, b, c, d, e, f, g, h, i, j, k) \text{ and } y = a\}$

Ví dụ 5: Danh sách các giáo viên không có tham gia để tài.

$\{p, q \mid \text{GIÁO VIÊN}(p, q, r, s, t, u, v, w, x, y, z) \wedge (\neg(\exists a)$

$(\text{THAM GIA ĐT}(a, b, c, d, e) \wedge a = p))\}$

IV. KẾT LUẬN

Chương 7 giới thiệu hai ngôn ngữ hình thức áp dụng trên mô hình dữ liệu quan hệ là ngôn ngữ phép tính bộ và ngôn ngữ phép tính miền. Trong khi ngôn ngữ đại số quan hệ thể hiện cách thức thực hiện một câu truy vấn thì ngôn ngữ phép tính quan hệ chỉ thể hiện điều kiện phải thoả mãn trên dữ liệu cần truy vấn. Chương tiếp theo sẽ trình bày về các nguyên tắc để đảm bảo dữ liệu trong một CSDL là đúng đắn.

Câu hỏi ôn tập

1. Đối với phép tính quan hệ có biến là bộ, hãy định nghĩa các khái niệm: biến bộ, quan hệ miền, công thức nguyên tố, công thức.
2. Kết quả của một biểu thức phép tính quan hệ biến bộ là gì?
3. Đối với phép tính quan hệ có biến là miền, hãy định nghĩa các khái niệm: biến miền, quan hệ miền, công thức nguyên tố, công thức.
4. Kết quả của một biểu thức phép tính quan hệ biến miền là gì?
5. Phép tính quan hệ có biến là bộ khác phép tính quan hệ có biến là miền như thế nào?
6. Hãy dùng ngôn ngữ phép tính quan hệ có biến là bộ và ngôn ngữ phép tính quan hệ có biến là miền để thể hiện những biểu thức đại số quan hệ sau:
 - a. $\Pi_{A,B}(R(A, B, C))$
 - b. $\sigma_{A=C}(R(A, B, C))$
 - c. $R(A, B, C) \cup S(A, B, C)$
 - d. $R(A, B, C) \cap S(A, B, C)$
 - e. $R(A, B, C) - S(A, B, C)$
 - f. $R(A, B, C) \times S(D, E, F)$
 - g. $R(A, B, C) * S(C, D, E)$
 - h. $R(A, B) \bowtie S(B, C)$
 - i. $R(A, B) \div S(B)$

Bài tập

1. Hãy dùng ngôn ngữ phép tính quan hệ biến bộ và biến miền để thực hiện các nhu cầu truy vấn trong bài tập 1 chương 5. Có những nhu cầu truy vấn nào ngoài khả năng thể hiện của ngôn ngữ phép tính quan hệ không ? Vì sao ?

2. Hãy dùng ngôn ngữ phép tính quan hệ biến bộ và biến miền để thực hiện các nhu cầu truy vấn trong bài tập 2 chương 5. Có những nhu cầu truy vấn nào ngoài khả năng thể hiện của ngôn ngữ phép tính quan hệ không ? Vì sao ?
3. Hãy dùng ngôn ngữ phép tính quan hệ biến bộ và biến miền để thực hiện các nhu cầu truy vấn trong bài tập 3 chương 5. Có những nhu cầu truy vấn nào ngoài khả năng thể hiện của ngôn ngữ phép tính quan hệ không ? Vì sao ?

Chương 8

RÀNG BUỘC TOÀN VẸN

I. GIỚI THIỆU

Một trong những vấn đề quan trọng mà người viết ứng dụng phải đối mặt đó chính là quá trình cập nhật CSDL (thêm, xóa, sửa) làm cho thông tin bị sai lệch theo nhiều cách khác nhau. Một cách phổ biến để đảm bảo việc không cho phép thêm vào các bộ dữ liệu sai là khi thêm, xóa, sửa dữ liệu, các thao tác này cần phải được kết hợp với những kiểm tra cần thiết để đảm bảo tính đúng đắn của dữ liệu. SQL cho phép chúng ta sử dụng các kỹ thuật khác nhau để khai báo các ràng buộc toàn vẹn (RBTW) như một phần của việc khai báo lược đồ CSDL. Trong chương này chúng tôi sẽ đề cập đến các ràng buộc liên quan đến các thuộc tính, các bộ và các quan hệ của CSDL.

II. KHÁI NIỆM VỀ RBTW

Trong một CSDL luôn luôn tồn tại nhiều mối liên hệ giữa các thuộc tính và giữa các bộ. Mỗi liên hệ này có thể xảy ra trong cùng một lược đồ quan hệ hoặc trong các lược đồ quan hệ của một CSDL. Sự liên hệ này là điều kiện bắt buộc mà mọi thể hiện của quan hệ điều phải thỏa ở bất kỳ thời điểm nào. RBTW chính là các điều kiện bắt buộc đó, và chúng được định nghĩa trên một hay nhiều quan hệ khác nhau.

Các RBTW này xuất phát từ những quy định hoặc điều kiện trong thực tế hoặc trong quá trình mô hình hóa dữ liệu. Chúng được kiểm tra vào thời điểm cập nhật dữ liệu bằng một hàm chuẩn hoặc một đoạn chương trình cụ thể.

III. TẠI SAO PHẢI CÓ RBTV

RBTV đảm bảo kết quả của sự tác động, thay đổi của người dùng lên CSDL không làm mất đi tính nhất quán của dữ liệu. Vì vậy nó bảo vệ CSDL trước những lỗi hoặc mâu thuẫn không lường trước trong quá trình khai thác dữ liệu. Ngoài ra, RBTV cũng được xem là một công cụ để diễn đạt ngữ nghĩa của CSDL. Vì vậy RBTV đảm bảo CSDL luôn biểu diễn đúng ngữ nghĩa thực tế.

IV. CÁC ĐẶC TRƯNG CỦA RBTV

Mỗi RBTV có ba đặc trưng chính: bối cảnh, nội dung và bảng tầm ảnh hưởng.

IV.1 Bối cảnh

Bối cảnh của một RBTV là những quan hệ mà việc thay đổi dữ liệu (thêm, xóa, sửa) trên các quan hệ đó có khả năng gây ra vi phạm RBTV.

Ví dụ:

Ràng buộc R1: Giáo viên quản lý chuyên môn phải là một giáo viên trong khoa.

Ta phải kiểm tra R1 khi:

- Cập nhật người quản lý chuyên môn cho một giáo viên.
- Thêm một giáo viên.

Vì vậy, bối cảnh cho R1 là quan hệ **GIÁO VIÊN**.

Ràng buộc R2: Ngày nhận chức của trưởng bộ môn phải lớn hơn ngày sinh.

Ta phải kiểm tra R2 khi:

- Cập nhật ngày sinh của một giáo viên.
- Thêm mới một bộ môn.
- Cập nhật trưởng bộ môn hay ngày nhận chức.

Vì vậy, bối cảnh cho R2 là quan hệ **GIÁO VIÊN** và **BỘ MÔN**.

IV.2 Nội dung của RBTV

Nội dung của RBTV có thể được biểu diễn bằng ngôn ngữ tự nhiên hoặc ngôn ngữ hình thức, và đôi khi cũng có thể biểu diễn bằng phụ thuộc hàm. Nội dung của RBTV nếu được biểu diễn bằng ngôn ngữ tự nhiên thì sẽ dễ hiểu, nhưng lại thiếu tính chặt chẽ. Ngược lại, nội dung của RBTV được biểu diễn bằng ngôn ngữ hình thức thì chặt chẽ nhưng đôi khi lại khó hiểu. Chúng ta có thể sử dụng các ngôn ngữ hình thức như ngôn ngữ đại số quan hệ, phép tính quan hệ, mã giả để biểu diễn nội dung của RBTV.

Ví dụ:

Ràng buộc R1:

- Ngôn ngữ tự nhiên: Giáo viên quản lý chuyên môn phải là một giáo viên trong khoa.
- Ngôn ngữ hình thức:

$\forall t \in \text{GIÁO VIÊN} (t.\text{GVQLCM} \neq \text{null} \Rightarrow$

$\exists s \in \text{GIÁO VIÊN} (t.\text{GVQLCM} = s.\text{MÃ GV}))$

Ràng buộc R2:

- Ngôn ngữ tự nhiên: Ngày nhận chức của trưởng bộ môn phải lớn hơn ngày sinh.
- Ngôn ngữ hình thức:

$\forall t \in \text{BỘ MÔN} (\exists s \in \text{GIÁO VIÊN} (s.\text{MÃ GV} = t.\text{TRƯỞNG GBM} \wedge$

$t.\text{NGÀY NHẬN CHỨC} > s.\text{NGÀY SINH}))$

IV.3 Bảng tầm ảnh hưởng

Bảng tầm ảnh hưởng thể hiện thao tác thay đổi dữ liệu nào (thêm, xóa, sửa) khi được thực hiện trên các quan hệ bối cảnh có thể gây ra vi phạm RBTV. Có hai loại bảng RBTV: bảng tầm ảnh hưởng cho một RBTV và bảng tầm ảnh hưởng tổng hợp cho các RBTV. Trong bảng tầm ảnh hưởng chúng ta sẽ sử dụng hai ký hiệu:

- Dấu cộng (+): mang ngữ nghĩa là hành động cập nhật tại quan hệ bối cảnh làm vi phạm RBTv và cần phải kiểm tra.
- Dấu trừ (-): mang ngữ nghĩa là hành động cập nhật tại quan hệ bối cảnh không vi phạm RBTv và không cần phải kiểm tra.
- Bảng tầm ảnh hưởng cho một RBTv:

Tên_RB	Thêm	Xóa	Sửa
Quan hệ 1	+	-	+ (Thuộc tính)
Quan hệ 2	-	+	-
...			
Quan hệ n	-	+	-

- Bảng tầm ảnh hưởng tổng hợp:

	Ràng buộc 1			Ràng buộc 2			...			Ràng buộc m		
	T	X	S	T	X	S	T	X	S
Quan hệ 1	+	-	+	+	-	+				+	-	+
Quan hệ 2	-	+	-									
Quan hệ 3	-	-	+							-	+	-
...												
Quan hệ n					-	+	-			-	-	+

V. PHÂN LOẠI RBTv

V.1 RBTv trên một quan hệ

V.1.1 RBTv miền giá trị

RBTv miền giá trị là ràng buộc quy định các giá trị cho một thuộc tính. Miền giá trị của một thuộc tính có thể liên tục hoặc rời rạc.

Ví dụ:

Ràng buộc R3: Phái của giáo viên phải là 'Nam' hoặc 'Nữ'.

Chương 8 Ràng buộc toàn vẹn

- Bối cảnh: GIÁO VIÊN

- Biểu diễn:

$\forall t \in \text{GIÁO VIÊN} \ (t.\text{PHÁI} \in \{ \text{'Nam'}, \text{'Nữ'} \})$

Hoặc $\text{DOM}(\text{PHÁI}) = \{ \text{'Nam'}, \text{'Nữ'} \}$

- Bảng tầm ảnh hưởng:

R3	Thêm	Xóa	Sửa
GIÁO VIÊN	+	-	+(PHÁI)

Ràng buộc R4: Kinh phí của mỗi đề tài không vượt quá 700 triệu.

- Bối cảnh: ĐỀ TÀI

- Biểu diễn:

$\forall t \in \text{ĐỀ TÀI} \ (t.\text{KINH PHÍ} \leq 700)$

- Bảng tầm ảnh hưởng:

R4	Thêm	Xóa	Sửa
ĐỀ TÀI	+	-	+(KINH PHÍ)

V.1.2 RBTV liên bộ trên một quan hệ

RBTV liên bộ trên một quan hệ là ràng buộc giữa các bộ trong cùng một quan hệ, nghĩa là sự tồn tại của một hay nhiều bộ phụ thuộc vào sự tồn tại của một hay nhiều bộ khác trong một quan hệ đó. Trường hợp đặc biệt của RBTV liên bộ trên một quan hệ là ràng buộc khóa chính và ràng buộc duy nhất.

Ví dụ:

Ràng buộc R5: Tên của mỗi bộ môn là duy nhất.

- Bối cảnh: BỘ MÔN

- Biểu diễn:

$\forall t1, t2 \in BỘMÔN (t1 \neq t2 \wedge t1.TÊNBM \neq t2.TÊNBM)$

- Bảng tầm ảnh hưởng:

R5	Thêm	Xóa	Sửa
BỘMÔN	+	-	+(TÊNBM)

Ràng buộc R6: Mỗi giáo viên không được tham gia quá 5 công việc trong cùng một đề tài.

- Bối cảnh: THAMGIAĐT

- Biểu diễn:

$\forall t \in THAMGIAĐT (Card(\{s \in THAMGIAĐT | s.MÃGV = t.MÃGV \wedge s.MÃĐT = t.MÃĐT\}) \leq 5)$

- Bảng tầm ảnh hưởng:

R6	Thêm	Xóa	Sửa
THAMGIAĐT	+	-	+(MÃGV, MÃĐT)

V.1.3 RBTV liên thuộc tính trên một quan hệ

RBTV liên thuộc tính trên một quan hệ là sự ràng buộc giữa các thuộc tính trong cùng một quan hệ.

Ví dụ:

Ràng buộc R7: Ngày bắt đầu luôn luôn nhỏ hơn ngày kết thúc của đề tài.

- Bối cảnh: ĐỀTÀI

- Biểu diễn:

$\forall t \in ĐỀTÀI (t.NGÀYBD < t.NGÀYKT)$

Chương 8 Ràng buộc toàn vẹn

- Bảng tầm ảnh hưởng:

R7	Thêm	Xóa	Sửa
ĐỀ TÀI	+	-	+(NGÀYBD, NGÀYKT)

Ràng buộc R8: Mỗi công việc của đề tài có ngày bắt đầu luôn nhỏ hơn ngày kết thúc.

- Bối cảnh: CÔNGVIỆC
- Biểu diễn:
 $\forall t \in \text{CÔNGVIỆC} (t.\text{NGÀYBD} < t.\text{NGÀYKT})$
- Bảng tầm ảnh hưởng:

R8	Thêm	Xóa	Sửa
CÔNGVIỆC	+	-	+(NGÀYBD, NGÀYKT)

V.2 RBTV trên nhiều quan hệ

V.2.1 RBTV tham chiếu

RBTV tham chiếu là ràng buộc trên giá trị các thuộc tính trong một quan hệ nào đó, các giá trị này phải xuất hiện (ở bộ nào đó) trên tập thuộc tính là khóa chính của một quan hệ khác. RBTV tham chiếu còn được gọi là ràng buộc khóa ngoại hay ràng buộc tồn tại. Ràng buộc này rất phổ biến trong CSDL quan hệ. Thông thường bối cảnh của ràng buộc này là hai quan hệ. Tuy nhiên, cũng có trường hợp bối cảnh của RBTV loại này chỉ là một quan hệ, ví dụ như ràng buộc R1.

Ví dụ:

Ràng buộc R1: Giáo viên quản lý chuyên môn phải là một giáo viên trong khoa.

- Bối cảnh: GIÁOVIÊN

- Biểu diễn:

$\forall t \in \text{GIÁOVIÊN} (t.\text{GVQLCM} \neq \text{null} \Rightarrow$

$\exists s \in \text{GIÁOVIÊN} (t.\text{GVQLCM} = s.\text{MÃGV})$

- Bảng tầm ảnh hưởng:

R1	Thêm	Xóa	Sửa
GIÁOVIÊN	+	+	+(MÃGV, GVQLCM)

Ràng buộc R9: Trưởng bộ môn phải là một giáo viên trong khoa.

- Bối cảnh: BỘMÔN, GIÁOVIÊN

- Biểu diễn:

$\forall t \in \text{BỘMÔN} (\exists s \in \text{GIÁOVIÊN} (t.\text{TRƯỞNGBM} = s.\text{MÃGV}))$

- Bảng tầm ảnh hưởng:

R9	Thêm	Xóa	Sửa
GIÁOVIÊN	-	+	+(MÃGV)
BỘMÔN	+	-	+(TRƯỞNGBM)

V.2.2 RBTV liên bộ trên nhiều quan hệ

RBTV liên bộ trên nhiều quan hệ là ràng buộc giữa các bộ trên các quan hệ khác nhau, nghĩa là sự tồn tại của bộ dữ liệu trên quan hệ này quyết định sự tồn tại của bộ dữ liệu trên quan hệ khác.

Ví dụ:

Ràng buộc R10: Mỗi đề tài phải có ít nhất một công việc.

- Bối cảnh: ĐỀTÀI, CÔNGVIỆC

- Biểu diễn:

Chương 8 Ràng buộc toàn vẹn

$\forall t \in \text{ĐỀTÀI} (\exists s \in \text{CÔNGVIỆC} (t.\text{MÃĐT} = s.\text{MÃĐT}))$

- Bảng tầm ảnh hưởng:

R10	Thêm	Xóa	Sửa
ĐỀTÀI	+	-	+(MÃĐT)
CÔNGVIỆC	-	+	+(MÃĐT)

Ràng buộc R11: Mỗi công việc phải có ít nhất một giáo viên tham gia.

- Bối cảnh: CÔNGVIỆC, THAMGIAĐT
- Biểu diễn:

$\forall t \in \text{CÔNGVIỆC} (\exists s \in \text{THAMGIAĐT} (t.\text{MÃĐT} = s.\text{MÃĐT} \wedge t.\text{STT} = s.\text{STT}))$

- Bảng tầm ảnh hưởng:

R10	Thêm	Xóa	Sửa
CÔNGVIỆC	+	-	+(MÃĐT,STT)
THAMGIAĐT	-	+	+(MÃĐT,STT)

V.2.3 RBTV liên thuộc tính liên quan hệ

RBTV liên thuộc tính liên quan hệ là ràng buộc xảy ra trên nhiều thuộc tính của nhiều quan hệ khác nhau.

Ví dụ:

Ràng buộc R2: Ngày nhận chức của trưởng bộ môn phải lớn hơn ngày sinh.

- Bối cảnh: BỘMÔN, GIÁOVIÊN
- Biểu diễn:

$\forall t \in \text{BỘMÔN} (\exists s \in \text{GIÁOVIÊN} (s.\text{MÃGV} = t.\text{TRƯỞNGGBM})$

$\wedge t.\text{NGÀY NHẬN CHỨC} > s.\text{NGÀY SINH})$

- Bảng tầm ảnh hưởng:

R2	Thêm	Xóa	Sửa
BỘ MÔN	+	-	+(TRƯỚNG BM, NGÀY NHẬN CHỨC)
GIÁO VIÊN	-	-	+(NGÀY SINH)

Ràng buộc R11: Ngày kết thúc của một công việc trong đề tài luôn luôn nhỏ hơn hay bằng ngày kết thúc của đề tài đó.

- Bối cảnh: CÔNG VIỆC, ĐỀ TÀI
- Biểu diễn:

$$\forall t \in \text{CÔNG VIỆC}, s \in \text{ĐỀ TÀI} \ (s.\text{MÃ ĐT} = t.\text{MÃ ĐT}) \\ \Rightarrow s.\text{NGÀY KẾT THÚC} > t.\text{NGÀY KẾT THÚC})$$

- Bảng tầm ảnh hưởng:

R11	Thêm	Xóa	Sửa
CÔNG VIỆC	+	-	+(MÃ ĐT, NGÀY KẾT THÚC)
ĐỀ TÀI	-	-	+(NGÀY KẾT THÚC)

V.2.4 RBTV trên thuộc tính tổng hợp

Thuộc tính tổng hợp là thuộc tính mà giá trị của nó được tính toán từ các thuộc tính khác.

Khi lược đồ CSDL có thuộc tính tổng hợp thì RBTV trên thuộc tính tổng hợp đảm bảo sự đúng đắn về mối quan hệ giữa thuộc tính tổng hợp và các thuộc tính nguồn.

Ví dụ:

Chương 8 Ràng buộc toàn vẹn

Giả sử trong quan hệ BỘ MÔN ta thêm thuộc tính số giáo viên (SỐGV) như sau:

BỘ MÔN (MÃBM, TÊNBM, PHÒNG, ĐIỆNTHOAI, TRƯỞNGBM, MÃKHOA, NGÀY NHẬN CHỨC, SỐGV)

Khi đó, trên CSDL xuất hiện một ràng buộc tổng hợp như sau:

Ràng buộc R12: Số giáo viên (SỐGV) của một bộ môn phải thực sự bằng số lượng giáo viên thuộc về bộ môn đó.

- Bối cảnh: GIÁO VIÊN, BỘ MÔN

- Biểu diễn:

$\forall t \in BỘ MÔN ($

$t.SỐGV = \text{card} \{ s \in GIÁO VIÊN \mid s.MÃBM = t.MÃBM \})$

- Bảng tầm ảnh hưởng:

R12	Thêm	Xóa	Sửa
GIÁO VIÊN	+	+	+ (MÃBM)
BỘ MÔN	-	-	+ (SỐGV, MÃBM)

VI. CÀI ĐẶT RBTV VỚI SQL

Chúng ta có thể cài đặt một số ràng buộc đơn giản bằng các kỹ thuật sẵn có của HQT CSDL như *primary key*, *foreign key*, *check constraint*. Một số ràng buộc phức tạp chúng ta cần phải khai báo thông qua một đoạn lệnh SQL, đoạn lệnh này có thể khai báo thông qua kỹ thuật *assertion*, *trigger*, và *stored procedure*. Phần kê tiếp chúng tôi giới thiệu một vài ví dụ về cài đặt RBTV với *assertion*, *trigger* và *stored procedure*.

VI.1 Cài đặt RBTV với Assertion

Assertion là một biểu thức SQL luôn mang giá trị TRUE tại mọi thời điểm. Do vậy khi sử dụng *assertion* để khai báo RBTV người dùng

phải cho biết điều kiện nào (được thể hiện bằng một biểu thức logic) cần phải đúng (có giá trị là TRUE).

Cú pháp tạo ràng buộc:

CREATE ASSERTION <Tên_ASSERTION> CHECK (<Điều_kiện>)

Cú pháp xóa ràng buộc:

DROP ASSERTION <Tên_ASSERTION>

Ví dụ: Ngày nhận chức của trưởng bộ môn phải lớn hơn ngày sinh.

CREATE ASSERTION R2 CHECK (

 NOT EXISTS (

 SELECT *

 FROM GIÁOVIÊN, BỘMÔN

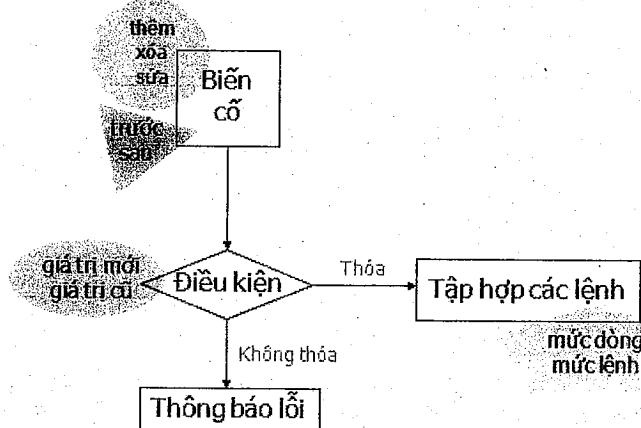
 WHERE MÃGV=TRƯỞNGBM

 AND NGÀYSINH > NGÀYNHẬNCHỨC))

VI.2 Cài đặt RBTv với trigger

Trigger là tập hợp các lệnh được thực hiện tự động khi xuất hiện một biến cố nào đó trên CSDL. Biến cố ở đây được xem là hành động thêm, xóa, sửa dữ liệu. Tập hợp các lệnh đó kiểm tra việc thêm, xóa, sửa có làm vi phạm RBTv không. Nếu không vi phạm thì việc thêm, xóa, sửa dữ liệu thành công, còn vi phạm thì hành động thêm, xóa, sửa dữ liệu sẽ bị báo lỗi, và không được thực hiện. Ngoài ra tập hợp các lệnh đó có thể được kiểm tra trước hoặc sau biến cố xảy ra. Nguyên lý hoạt động được minh họa ở sơ đồ Hình VIII - 1.

Chương 8 Ràng buộc toàn vẹn



Hình VIII -1 Hoạt động của trigger.

Cú pháp tạo trigger:

CREATE TRIGGER <Tên_trigger>

AFTER|BEFORE INSERT|UPDATE|DELETE ON <Tên_bảng>

REFERENCING

NEW ROW|TABLE AS <Tên_1>

OLD ROW|TABLE AS <Tên_2>

FOR EACH ROW | FOR EACH STATEMENT

WHEN (<Điều kiện>)

<Tập lệnh_SQL>

Cú pháp xóa trigger:

DROP TRIGGER <Tên_trigger>

Ví dụ: Lương của trưởng bộ môn phải lớn hơn 2000000.

```
CREATE TRIGGER TR_BM_UPD
AFTER UPDATE OF TRƯỜNGBM ON BỘMÔN
REFERENCING
    NEW ROW AS NewTuple
FOR EACH ROW
WHEN (2000000 >= (SELECT LUONG FROM GIÁOVIÊN
                      WHERE MÃGV=NewTuple.TRƯỜNGBM))
    <Phản gọi hàm/lệnh thông báo lỗi cho người dùng>
```

Trong SQL chúng ta có thể sử dụng lệnh *PRINT* hoặc hàm *RAISERROR* để thông báo lỗi cho người dùng.

VI.3 Cài đặt RBTv với Stored Procedure

Tập các lệnh được sử dụng để thực hiện một xử lý nào đó trong một ứng dụng CSDL được gọi là *giao tác (transaction)*. Trong quá trình thực thi tập các lệnh này cần phải đảm bảo:

- hoặc là các lệnh đều được thực hiện thành công.
- hoặc là không có lệnh nào được thực hiện.

Giao tác phải đảm bảo tính nguyên tố và tính nhất quán của CSDL. Tính nhất quán được đảm bảo nghĩa là các RBTv không được vi phạm hoặc trước, hoặc trong khi, hoặc sau khi thực hiện giao tác.

Ví dụ:

Giao tác Chuyển_tiền

- Kiểm tra tài khoản và tiền trong tài khoản người gửi.
- Giảm tiền trong tài khoản người gửi.
- Tăng tiền trong tài khoản người nhận.
- Nếu tất cả đều thành công thì hoàn tất giao tác.
- Ngược lại quay lui giao tác.

Cuối giao tác

Stored procedure là thủ tục do người dùng định nghĩa được lưu trữ trong DBMS và nó thực thi khi được gọi. Các RBTV được khai báo bằng trigger sẽ được tự động kiểm tra khi có biến cố xảy ra. Do vậy nếu hệ thống kiểm tra RBTV bằng trigger quá nhiều thì có thể làm cho tốc độ xử lý chậm. Để kiểm tra RBTV hiệu quả hơn chúng ta có thể dùng stored procedure.

Cú pháp:

CREATE PROCEDURE <Tên_thủ_tục> <DS_tham_số>

AS

Khai báo biến cục bộ

Thân chương trình

GO

EXEC <Tên_thủ_tục> <DS_tham_số>

Ví dụ: Số giáo viên của một bộ môn phải bằng tổng số lượng giáo viên thuộc về bộ môn đó.

CREATE PROCEDURE Thêm_giáo_viện t GIÁOVIÊN

AS

begin tran

 Thêm t vào GIÁO VIÊN

 If @@ERROR <>0 rollback tran

 commit tran

GO

EXEC Thêm_giáo_viện x

VII. KẾT LUẬN

Trong chương này chúng tôi đã trình bày ý nghĩa của RBTV trong một ứng dụng CSDL. Các RBTV được phân loại chi tiết và có ví dụ minh họa cụ thể để người đọc dễ hiểu. Ngoài ra chúng tôi cũng trình

bày cách thức cài đặt RBTB bằng *Assertion*, *Trigger*, và *Transaction* của SQL. Trong thực tế chúng ta cần phải cân nhắc nơi khai báo RBTB để cho hệ thống hoạt động hiệu quả. Ta có thể khai báo RBTB ở phía HQT CSDL (máy chủ) hoặc là ở phía ứng dụng (máy khách/máy trạm). Trong chương tiếp theo chúng tôi sẽ trình bày về tối ưu truy vấn.

Câu hỏi ôn tập

1. Tại sao ràng buộc toàn vẹn là quan trọng đối với mỗi cơ sở dữ liệu? Ràng buộc toàn vẹn đảm bảo được những vấn đề gì khi khai thác một cơ sở dữ liệu?
2. Bối cảnh của một RBTB là gì?
3. Có thể biểu diễn nội dung của RBTB bằng những ngôn ngữ hình thức nào?
4. Bảng tầm ảnh hưởng của một RBTB gồm có những thành phần nào? Hãy mô tả ý nghĩa từng thành phần trong bảng tầm ảnh hưởng.
5. Ràng buộc toàn vẹn được phân ra những loại nào? Hãy đưa ra một cơ sở dữ liệu để minh họa cho mỗi loại ràng buộc đã trình bày (mỗi ràng buộc toàn vẹn cần trình bày 3 nội dung: bối cảnh, nội dung và bảng tầm ảnh hưởng).

Bài tập

1. Với lược đồ *Ứng dụng Quản lý để tài nghiên cứu khoa học* trong Giáo trình– trang 57, hãy xác định một cách có phân loại toàn bộ các ràng buộc toàn vẹn nhận diện được .
2. Với tất cả các mô hình ER xây dựng được từ bài tập chương 2, hãy chuyển sang mô hình dữ liệu quan hệ. Với mỗi lược đồ CSDL , hãy xác định một cách có phân loại các ràng buộc toàn vẹn có thể có.
3. Hãy liệt kê có phân loại toàn bộ các ràng buộc toàn vẹn trong lược đồ CSDL bài tập 2, 3, 4 chương 3.
4. Cho lược đồ cơ sở dữ liệu quản lý hàng hóa của một cửa hàng như sau:

MẶTHÀNG(MÃMH, TÊNMH, ĐVT, ĐONGIÁ)

Tân từ: Thông tin mặt hàng được lưu trữ trong quan hệ MẶTHÀNG. Mỗi mặt hàng gồm có các thông tin: mã mặt hàng (MÃMH), tên mặt hàng (TÊNMH), đơn vị tính (ĐVT) và đơn giá (ĐONGIÁ). Mỗi mặt hàng có một mã mặt hàng để phân biệt với các mặt hàng khác.

NHÀ CUNG CẤP (MÃ NCC, TÊN NCC, ĐỊA CHỈ)

Tân từ: Nhà cung cấp là đơn vị cung ứng mặt hàng cho cửa hàng, thông tin nhà cung cấp được lưu trữ trong quan hệ NHÀ CUNG CẤP. Mỗi nhà cung cấp gồm có thông tin mã nhà cung cấp (MÃ NCC), tên nhà cung cấp (TÊN NCC), địa chỉ (ĐỊA CHỈ). Mỗi nhà cung cấp có một mã nhà cung cấp để phân biệt với nhà cung cấp khác.

CUNG ỨNG (MÃ NCC, MÃ MH)

Tân từ: Mỗi nhà cung ứng có thể cung ứng nhiều mặt hàng và mỗi mặt hàng có thể do nhiều nhà cung ứng cung ứng.

HÓA ĐƠN (MÃ HD, NGÀY HD, MÃ NCC, TỔNG TIỀN, TỔNG SỐ MH)

Tân từ: Hóa đơn dùng để ghi nhận lại thông tin đặt đã đặt hàng từ nhà cung cấp. Mỗi hóa đơn gồm có mã hóa đơn (MÃ HD), ngày lập hóa đơn (NGÀY HD), hóa đơn gởi cho nhà cung cấp nào (MÃ NCC), tổng tiền hóa đơn (TỔNG TIỀN) và tổng số mặt hàng đã đặt (TỔNG SỐ MH). Mỗi hóa đơn có một mã hóa đơn để phân biệt với các hóa đơn khác.

CTHD (MÃ HD, MÃ MH, SỐ LƯỢNG)

Tân từ: Chi tiết hóa đơn biết hóa đơn đã đặt mua những mặt hàng nào và số lượng (SỐ LƯỢNG) là bao nhiêu.

GIAO HÀNG (MÃ GH, NGÀY GH, SỐ HD, TỔNG TIỀN, TỔNG SỐ MH)

Tân từ: Quan hệ giao hàng dùng để lưu trữ lại hóa đơn cho mỗi lần giao hàng. Mỗi hóa đơn giao hàng gồm có mã giao hàng (MÃ GH), ngày giao (NGÀY GH), giao cho hóa đơn nào (SỐ HD), tổng tiền thanh toán cho lần giao hàng đó (TỔNG TIỀN), và tổng số mặt hàng đã giao (TỔNG SỐ MH). Mỗi hóa đơn giao hàng được phân biệt bởi mã giao hàng duy nhất.

CTGH (MÃ GH, MÃ MH, SỐ LƯỢNG)

Tân từ: Chi tiết giao hàng dùng để ghi nhận lại chi tiết cho mỗi lần giao hàng gồm đã giao những mặt hàng nào và số lượng bao nhiêu (SỐ LƯỢNG).

Yêu cầu: Biểu diễn tất cả các loại ràng buộc toàn vẹn sau:

- Tất cả các thuộc tính mang tính chất kiểu số có giá trị luôn lớn hơn 0.
- Nhà cung cấp chỉ được cung cấp các mặt hàng có khả năng

- c) Tổng số tiền của hóa đơn phải bằng tổng tiền của chi tiết hóa đơn đó.
- d) Tổng số mặt hàng của hóa đơn phải bằng tổng số mặt hàng của chi tiết hóa đơn đó.
- e) Tổng tiền của hóa đơn phải bằng tổng tiền của tất cả các lần giao hàng
- f) Tổng số mặt hàng của hóa đơn phải bằng tổng số mặt hàng của tất cả các lần giao hàng.
- g) Tổng tiền của giao hàng phải bằng với tổng tiền của chi tiết các lần giao
- h) Tổng số mặt hàng của mỗi lần giao hàng phải bằng với tổng tiền của chi tiết các lần giao.
- i) Chỉ được phép giao những mặt hàng mà có đặt hàng.
- j) Mỗi hóa đơn chỉ có tối đa 3 lần giao hàng
- k) Khoảng cách của ngày giao hàng và ngày đặt hàng không vượt quá 1 tuần.
- l) Ngày giao hàng phải luôn lớn hơn ngày đặt hàng.

Chương 9

TỐI ƯU TRUY VẤN

I. GIỚI THIỆU

I.1 Quá trình xử lý truy vấn

Một câu truy vấn viết bằng ngôn ngữ truy vấn cấp cao (ví dụ SQL) sẽ được xử lý qua các bước sau:

Bước 1: Phân tích từ.

Hệ thống duyệt qua câu truy vấn để xác định từ khóa của ngôn ngữ SQL, tên thuộc tính, tên quan hệ được dùng trong câu truy vấn.

Bước 2: Kiểm tra cú pháp.

Hệ thống kiểm tra xem câu truy vấn có tuân thủ đúng cú pháp của ngôn ngữ truy vấn đang dùng hay không.

Bước 3: Kiểm tra hợp lệ.

Hệ thống kiểm tra các tên thuộc tính và tên quan hệ có hợp lệ và có đúng như đã khai báo chúng trong lược đồ CSDL đang được dùng đến hay không. Ngoài ra, hệ thống còn kiểm tra sự nhập nhằng về tên đối tượng, kiểm tra sự hợp lệ về kiểu dữ liệu khi so sánh, ...

Kế đến câu truy vấn được hệ thống thể hiện lại dưới dạng *cây truy vấn* (*query tree*) hoặc *đồ thị truy vấn* (*query graph*). Căn cứ vào đó, hệ thống đề nghị chiến lược thực thi tốt nhất có thể được và phát sinh mã tương ứng để thi hành và cho ra kết quả cuối cùng.

1.2 Tối ưu hóa câu truy vấn

Một câu truy vấn thường có nhiều chiến lược thực thi khác nhau và quá trình chọn ra một chiến lược phù hợp để thực hiện câu truy vấn gọi là *tối ưu truy vấn* (*query optimization*).

Ta biết rằng khi truy vấn trên dữ liệu dùng ngôn ngữ truy vấn cấp cao, ta chỉ việc nêu điều kiện cho kết quả cần tìm mà không phải chỉ ra cách thức để đạt được kết quả đó. Vì vậy, tối ưu hóa truy vấn là quá trình thực sự cần thiết trong những hệ thống hỗ trợ truy vấn dùng ngôn ngữ truy vấn cấp cao.

Quá trình tối ưu truy vấn không phải luôn luôn cho ra kết quả là một chiến lược thực thi tối ưu nhất, mà đôi khi chỉ là một chiến lược thực thi có hiệu quả tốt chấp nhận được. Trừ những câu truy vấn đơn giản, việc tìm ra một chiến lược thực thi truy vấn tối ưu tốn nhiều thời gian và cần nhiều dữ liệu liên quan. Số lượng chiến lược thực thi truy vấn cần xem xét thường phụ thuộc vào: câu truy vấn cụ thể, thuật toán truy xuất dữ liệu của HQT CSDL, cách thức lưu trữ dữ liệu ở mức vật lý do người thiết kế CSDL quy định.

Có hai phương pháp tối ưu truy vấn:

- Phương pháp dựa trên các luật *heuristic*: dùng các luật *heuristic* để thay đổi thứ tự các phép toán trong một cây truy vấn để cho ra một cây truy vấn tương đương với cây truy vấn ban đầu nhưng chi phí thực hiện thấp hơn. Mỗi luật heuristic có thể chỉ cho ra kết quả tốt trong hầu hết các trường hợp nhưng không đảm bảo sẽ cho kết quả tốt trong mọi trường hợp.
- Phương pháp ước lượng chi phí: tính chi phí cho nhiều chiến lược thực thi của một câu truy vấn và chọn ra một chiến lược thực thi có chi phí thấp nhất.

Trong các HQT CSDL, hai phương pháp này thường được sử dụng kết hợp với nhau phục vụ việc tối ưu truy vấn.

II. TỐI UƯ TRUY VẤN DÙNG PHƯƠNG PHÁP DỰA TRÊN HEURISTIC

II.1 Thể hiện lại câu truy vấn

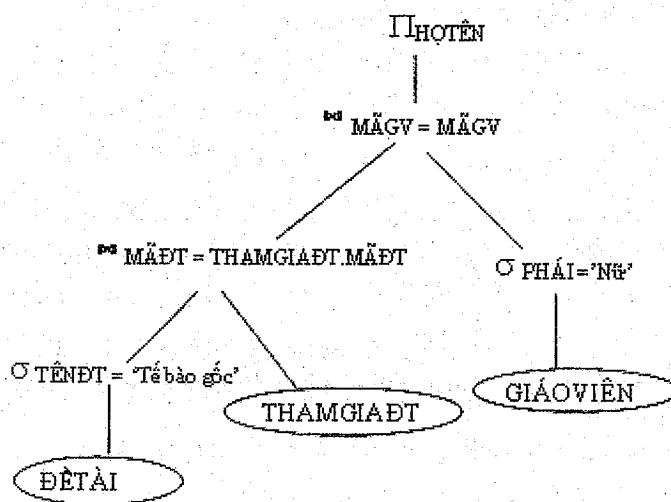
Như đã đề cập, SQL là ngôn ngữ truy vấn được dùng trong hầu hết các HQT CSDL quan hệ. Câu SQL ban đầu được hệ thống phân rã thành các câu truy vấn con (nếu có), mỗi câu chỉ liên quan đến 1 mệnh đề SELECT – FROM – WHERE – GROUP BY – HAVING. *Bộ tối ưu truy vấn* sẽ thực hiện tối ưu hóa cho từng câu truy vấn con.

Mỗi câu truy vấn con sau đó được chuyển sang đại số quan hệ và thể hiện dưới dạng *cây truy vấn*.

Cây truy vấn là một cấu trúc dữ liệu dạng cây để thể hiện một biểu thức đại số quan hệ. Nút lá của cây ứng với các quan hệ đầu vào, mỗi nút trung gian ứng với một phép toán của biểu thức đại số quan hệ. Cây truy vấn thể hiện thứ tự thực hiện các phép toán trong một biểu thức đại số quan hệ. Việc thực thi một câu truy vấn được thể hiện bởi cây truy vấn dựa theo nguyên tắc sau:

- Các phép toán được thực hiện theo hướng từ lá đến gốc. Thực thi phép toán (nút trung gian) khi các toán hạng tương ứng đã sẵn sàng.
- Thay thế nút trung gian đó bởi kết quả đạt được và cứ như thế, phép toán tiếp tục được thực hiện.
- Quá trình thực thi kết thúc khi nút gốc đã thực thi xong và kết quả đạt được là kết quả cần tìm.

Ví dụ: Hình IX-1 là cây truy vấn cho biết tên các giáo viên nữ có tham gia đề tài tên là ‘Tế bào gốc’.



Hình IX-1 Cây truy vấn.

II.2 Các luật biến đổi tương đương

Một câu truy vấn có thể có nhiều cách thể hiện khác nhau bằng ngôn ngữ đại số quan hệ, và vì thế cũng có nhiều cây truy vấn tương ứng. Bộ kiểm tra cú pháp đầu tiên phát sinh một cây truy vấn ứng với câu SQL ban đầu. Bộ tối ưu sẽ vận dụng các luật biến đổi tương đương để biến đổi cây truy vấn này thành cây truy vấn tối ưu hơn và cuối cùng được một cây truy vấn thể hiện một chiến lược thực thi hiệu quả nhất có thể được.

Sau đây là một số luật biến đổi tương đương được dùng đến trong quá trình tối ưu hóa. Ký hiệu \equiv để chỉ sự tương đương của hai quan hệ. Hai quan hệ là tương đương nhau nếu chúng có khả năng thể hiện thông tin giống nhau. Hai quan hệ có cùng tập hợp các thuộc tính nhưng thứ tự thuộc tính khác nhau và có số dòng dữ liệu như nhau cũng được xem là tương đương.

1. Tách phép chọn có điều kiện chọn phức thành các phép chọn đơn:

$$\sigma_{c1} \text{AND } c2 \text{ AND } \dots \text{ AND } cn (R) \equiv \sigma_{c1} (\sigma_{c2} (\dots \sigma_{cn} (R) \dots))$$

2. Tính giao hoán của phép chọn σ :

Chương 9 Tối ưu truy vấn

$$\sigma_{c1}(\sigma_{c2}(R)) \equiv \sigma_{c2}(\sigma_{c1}(R))$$

3. Khi có nhiều phép chiếu lồng nhau, chỉ cần giữ lại phép chiếu cuối cùng:

$$\Pi_{L1}(\Pi_{L2}(\dots(\Pi_{Ln}(R))\dots)) \equiv \Pi_{L1}(R)$$

4. Tính giao hoán giữa phép σ và Π : nếu điều kiện chọn c chỉ liên quan đến các thuộc tính chiếu $L1, L2, \dots, Ln$ thì có thể hoán vị phép chọn và phép chiếu.

$$\Pi_{L1, L2, \dots, Ln}(\sigma_c(R)) \equiv \sigma_c(\Pi_{L1, L2, \dots, Ln}(R))$$

5. Tính giao hoán của phép kết và phép tích Đè - các:

$$R_1 \bowtie_c R_2 \equiv R_2 \bowtie_c R_1$$

$$R_1 \times R_2 \equiv R_2 \times R_1$$

6. Tính giao hoán giữa phép chọn và phép tích Đè - các: nếu các thuộc tính trong điều kiện chọn c chỉ liên quan đến một trong hai quan hệ tham gia vào phép kết thì có thể giao hoán chúng như sau:

$$\sigma_c(R_1 \bowtie R_2) \equiv (\sigma_c(R_1)) \bowtie R_2$$

Nếu c có thể viết là $c1 \text{ AND } c2$, $c1$ liên quan đến các thuộc tính của $R1$, $c2$ liên quan đến các thuộc tính của $R2$, các phép toán có thể được giao hoán như sau:

$$\sigma_c(R_1 \bowtie R_2) \equiv (\sigma_{c1}(R_1)) \bowtie (\sigma_{c2}(R_2))$$

7. Giao hoán giữa phép Π và \bowtie (hoặc \times): giả sử tập thuộc tính chiếu $L = \{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m\}$ trong đó A_i là các thuộc tính của R_1 , B_i là các thuộc tính của R_2 và điều kiện kết c chỉ liên quan đến tập thuộc tính L , khi đó tính giao hoán được thể hiện như sau:

$$\Pi_L(R_1 \bowtie R_2) \equiv (\Pi_{A1, A2, \dots, An}(R_1)) \bowtie_c (\Pi_{B1, B2, \dots, Bm}(R_2))$$

Nếu điều kiện c chứa thêm các thuộc tính không thuộc L , thì phải bổ sung các thuộc tính này vào tập thuộc tính chiếu và cuối cùng thực hiện thêm một phép toán chiếu nữa. Giả sử A_{n+1}, \dots, A_{n+k} và B_{m+1}, \dots, B_{m+k} là các thuộc tính có trong điều kiện c nhưng không thuộc tập L , khi đó có thể giao hoán chúng như sau:

$$\Pi_L(R_1 \bowtie_c R_2) \equiv \Pi_L ((\Pi_{A1, A2 \dots, An} A_{n+1}, \dots, A_{n+k} (R_1)) \bowtie_c (\Pi_{B1, B2 \dots, Bm} B_{m+1}, \dots, B_{m+k} (R_2)))$$

Nếu là phép tích Đề-các, sẽ không có điều kiện c, có thể giao hoán chúng như sau:

$$\Pi_L(R_1 \times R_2) \equiv (\Pi_{A1, A2 \dots, An} (R_1)) \times (\Pi_{B1, B2 \dots, Bm} (R_2))$$

8. Phép hội \cup và phép giao \cap có tính giao hoán, nhưng phép trừ – thì không có tính giao hoán.
9. Tính kết hợp của các phép toán \bowtie , \times , \cup và \cap . Gọi θ là một trong 4 phép toán, ta có: $(R_1 \theta R_2) \theta R_3 \equiv R_1 \theta (R_2 \theta R_3)$
10. Giao hoán giữa σ và các phép toán tập hợp θ (gồm \cup , \cap và $-$):

$$\sigma_c (R_1 \theta R_2) \equiv (\sigma_c (R_1)) \theta (\sigma_c (R_2))$$

11. Giao hoán giữa phép Π và phép hội:

$$\Pi_L (R_1 \cup R_2) \equiv \Pi_L (R_1) \cup \Pi_L (R_2)$$

12. Thực hiện hai phép toán theo thứ tự phép tích Đề-các rồi đến phép chọn tương đương với việc thực hiện một phép kết:

$$\sigma_c (R_1 \times R_2) \equiv R_1 \bowtie_c R_2$$

II.3 Thuật toán

Như đã đề cập, phương pháp tối ưu dựa trên luật heuristic vận dụng các luật biến đổi tương đương trên để chuyển cây truy vấn ban đầu thành cây truy vấn có hiệu quả thi hành tối ưu hơn. Sau đây là các bước của thuật toán:

1. Dùng luật 1, tách các phép chọn phức thành các phép chọn đơn để có thể tự do di chuyển phép chọn xuống các nhánh của cây.
2. Dùng luật 2, 4, 6, 10 liên quan đến tính giao hoán giữa phép chọn và các phép toán khác để di chuyển các phép chọn xuống càng gần nút lá của cây càng tốt.
3. Dùng luật 5 và 9 liên quan đến tính chất giao hoán và kết hợp của các phép 2 ngôi, để sắp xếp lại các nút lá của cây, sao cho

các phép chọn cho kết quả có số bộ ít hơn được ưu tiên thực hiện trước và tránh việc phải thực hiện phép tích Đề-các.

4. Dùng luật 12, kết hợp tích đề-các và phép chọn thành phép kết, nếu có thể xem điều kiện chọn là điều kiện kết.
5. Dùng luật 3, 4, 7, 11 để tách và di chuyển các phép chiếu xuống các nhánh. Chỉ giữ lại các thuộc tính cần thiết cho kết quả cuối cùng hoặc cho các phép toán kế tiếp sau mỗi phép chiếu.
6. Nhận biết từng nhánh của cây gồm một nhóm các thao tác có thể thi hành bởi một thuật toán.

Như vậy, tư tưởng chung của thuật toán tối ưu truy vấn dựa trên heuristic là cho thực hiện trước các phép toán làm giảm kích thước kết quả trung gian. Cụ thể là cho thực hiện càng sớm càng tốt các phép chọn để chỉ giữ lại các bộ cần thiết và các phép chiếu để chỉ giữ lại các thuộc tính cần thiết. Ngoài ra, các phép chọn và phép kết cho ra kết quả có số dòng ít hơn nên cho thực hiện trước.

II.4 Ví dụ minh họa

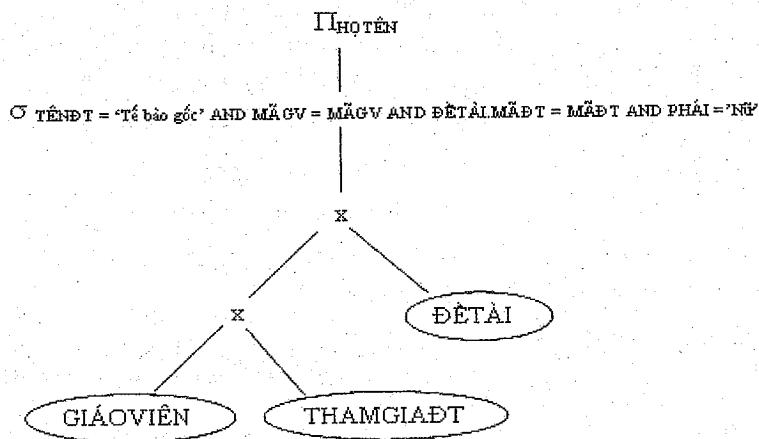
Tìm tên các giáo viên nữ có tham gia đề tài tên là ‘Tế bào gốc’.

SELECT HỌTÊN

FROM GIÁOVIÊN, THAMGIAĐT, ĐỀTÀI

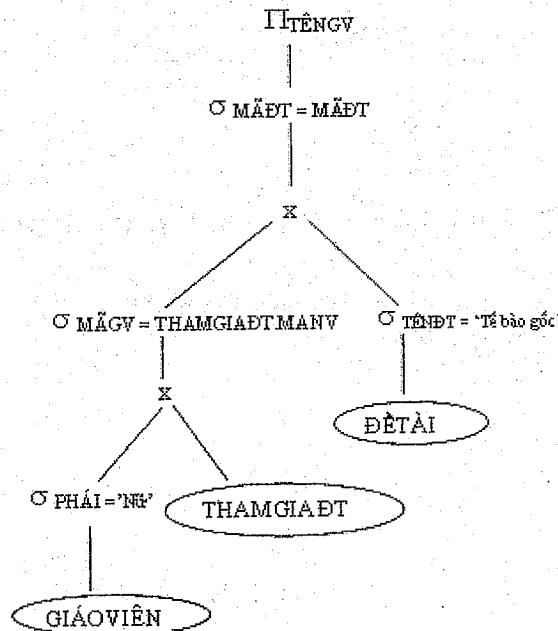
WHERE TÊNĐT = ‘Tế bào gốc’ AND PHÁI = ‘Nữ’ AND
GIÁOVIÊN.MÃGV = THAMGIAĐT.MÃGV AND ĐỀTÀI.MÃĐT =
THAMGIAĐT.MÃĐT

Hình IX-2 là cây truy vấn ban đầu ứng với câu SQL trên. Thực thi câu truy vấn theo cách này cho ra kết quả trung gian có dung lượng rất lớn vì là tích Đề-các của 3 quan hệ GIÁOVIÊN, THAMGIAĐT, ĐỀTÀI. Thật ra, kết quả cuối cùng chỉ liên quan đến một dòng trong quan hệ ĐỀTÀI (tên đề tài là ‘Tế bào gốc’), một số dòng trong quan hệ GIÁOVIÊN (chỉ các giáo viên nữ) và các dòng liên quan trong quan hệ THAMGIAĐT.



Hình IX-2 Cây truy vấn ban đầu.

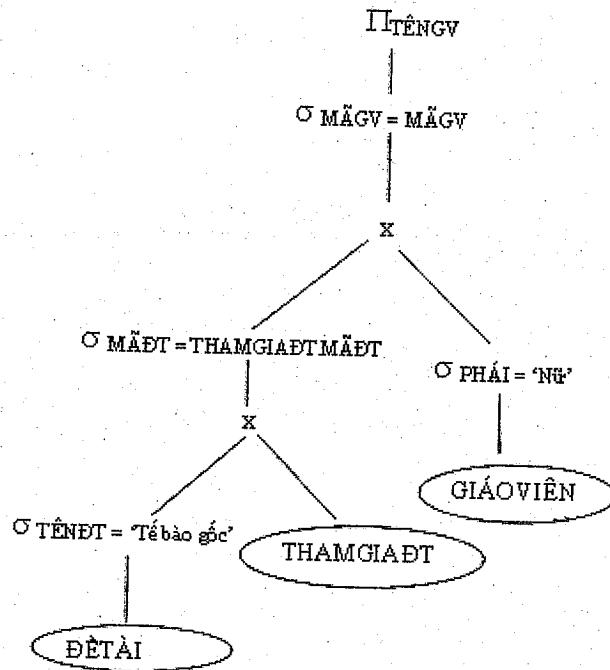
Hình IX-3 là cây truy vấn có được sau khi áp dụng bước 1 và 2 của thuật toán. Phép chọn được đẩy xuống gần nút lá của cây làm giảm số bộ của các quan hệ tham gia vào các phép tích Đề-các.



Hình IX-3 Cây truy vấn có được sau khi áp dụng bước 1 và 2 của thuật toán.

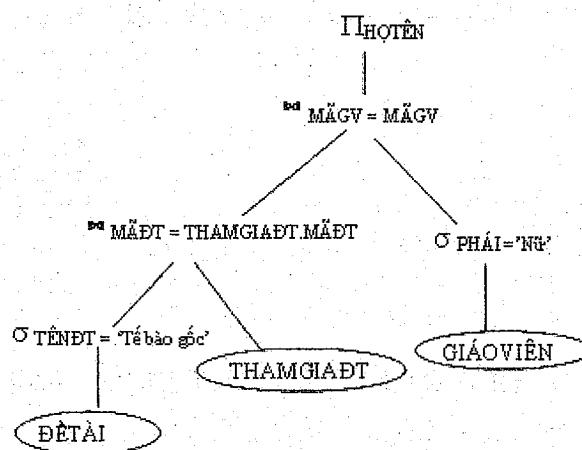
Chương 9 Tối ưu truy vấn

Hình IX-4 là cây truy vấn có được sau khi áp dụng bước 3 của thuật toán. Khi đó, phép chọn cho ra kết quả có ít dòng dữ liệu hơn sẽ được thực hiện trước. Hai quan hệ ĐỀTÀI và GIÁOVIÊN được đổi vị trí, kết quả không đổi nhưng chi phí để xử lý chắc chắn sẽ thấp hơn.



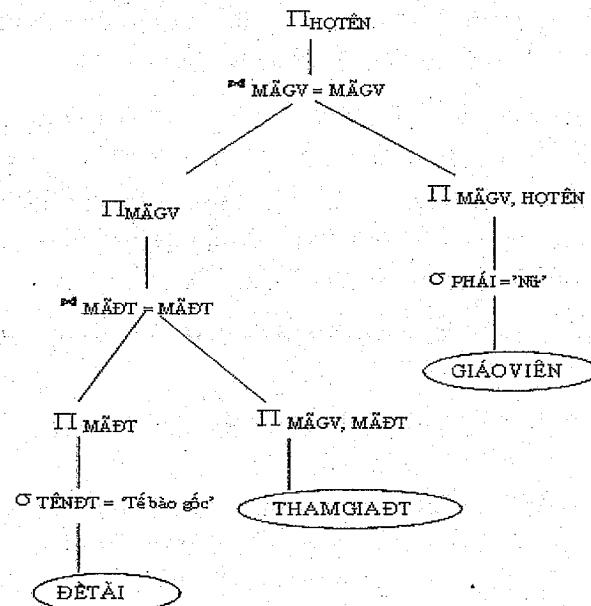
Hình IX-4 Cây truy vấn có được sau khi áp dụng bước 3 của thuật toán.

Hình IX-5 là kết quả vận dụng bước 4 của thuật toán. Các bộ hai theo thứ tự phép tích Đề-các rồi đến phép chọn được thay thế bởi các phép kết có chi phí thấp hơn.



Hình IX-5 Cây truy vấn có được sau khi áp dụng bước 4 của thuật toán.

Hình IX-6 là cây truy vấn có được sau khi vận dụng bước 5 của thuật toán. Khi đó, các phép chiếu chỉ giữ lại những thuộc tính cần thiết và được di chuyển xuống gần nút lá hơn.



Hình IX-6 Cây truy vấn có được sau khi áp dụng bước 5 của thuật toán.

III. TỐI UƯU TRUY VẤN DÙNG PHƯƠNG PHÁP ƯỚC LƯỢNG CHI PHÍ

III.1 Giới thiệu

Ta biết rằng ứng với mỗi câu truy vấn sẽ có nhiều chiến lược thực thi khác nhau. Dù các chiến lược thực thi này cho ra cùng kết quả nhưng chúng có chi phí khác nhau. Tối ưu truy vấn dựa trên việc ước lượng chi phí sẽ tính chi phí cho các chiến lược thực thi khác nhau và chọn ra chiến lược thực thi có chi phí thấp nhất.

III.2 Các loại chi phí và dữ liệu cần thiết

Chi phí cho một chiến lược thực thi truy vấn bao gồm:

1. Chi phí truy cập bộ nhớ phụ: là chi phí cho việc tìm kiếm, đọc và ghi các *khối* (*block*) dữ liệu trên đĩa. Chi phí tìm kiếm mẩu tin trên tập tin phụ thuộc vào cách thức lưu trữ dữ liệu vật lý (ví dụ tập tin có thứ tự hay không, có sử dụng kỹ thuật băm để lưu trữ dữ liệu hay có sử dụng chỉ mục hay không, ...). Ngoài ra, các yếu tố về vị trí định vị các khối trên đĩa là liên tục hay không liên tục cũng ảnh hưởng đến chi phí truy cập.
2. Chi phí lưu trữ: là chi phí lưu kết quả trung gian khi thực hiện truy vấn.
3. Chi phí tính toán: là chi phí thực hiện các tác vụ trong bộ nhớ chính trong suốt quá trình một chiến lược thực thi được thi hành, gồm tác vụ tìm kiếm, sắp xếp, trộn, ...
4. Chi phí sử dụng bộ nhớ chính dùng trong suốt quá trình thi hành câu truy vấn.
5. Chi phí truyền thông là chi phí truyền kết quả trả về đến nơi đã gửi yêu cầu truy vấn.

Khi làm việc với CSDL lớn, người ta chỉ chú trọng việc giảm thiểu chi phí truy cập bộ nhớ phụ. Vì vậy hàm chi phí để đánh giá các chiến lược thực thi khác nhau thường chỉ quan tâm đến số khối được chuyển từ đĩa lên bộ nhớ chính và bỏ qua các loại chi phí khác. Đối với CSDL nhỏ hơn, hầu hết dữ liệu liên quan đến câu truy vấn đều có thể lưu trữ trong bộ nhớ chính, khi đó hàm chi phí chỉ quan tâm đến chi phí tính

toán trong bộ nhớ chính. Với CSDL phân tán, chi phí truyền thông là mối quan tâm hàng đầu.

Để tính chi phí cho một chiến lược thực thi, hệ thống cần phải lưu lại những thông tin sau:

- Số bộ của quan hệ: r .
- Kích thước (hoặc kích thước trung bình) của 1 bộ R .
- Số khối cần thiết để lưu quan hệ: b .
- Số mẫu tin trong một khối (blocking factor): bfr .
- Thông tin liên quan đến vấn đề lưu trữ vật lý của quan hệ: dữ liệu có thứ tự hay không, nếu có thì được sắp theo thuộc tính nào, có chỉ mục sơ cấp hay không, có chỉ mục thứ cấp hay không và trên các thuộc tính nào. Ngoài ra, số mức x của chỉ mục đa cấp, số khối để lưu mức đầu tiên của cây chỉ mục bi cũng cần thiết cho việc tính chi phí.
- Số giá trị phân biệt của từng thuộc tính: d .
- Tỉ lệ các bộ thỏa mãn một điều kiện chọn trên 1 thuộc tính: sl .
- Số bộ trung bình thỏa mãn một điều kiện chọn trên 1 thuộc tính s . ($s = sl * r$, với một thuộc tính là thuộc tính khóa thì $d = r$, $sl = 1/r$ và $s = 1$).

HQT CSDL cài đặt nhiều cách thực thi khác nhau đối với một phép toán trên CSDL quan hệ. Tùy vào cách tổ chức dữ liệu ở mức vật lý và tùy phép toán cụ thể cần thực hiện mà hệ thống chọn ra cách thực thi có chi phí thấp nhất. Do có nhiều kiến thức nằm ngoài yêu cầu của môn CSDL liên quan đến các cách cài đặt các phép toán đại số quan hệ, các cách tổ chức vật lý một tập tin, cách tính chi phí cho một chiến lược thực thi truy vấn, ... nên sẽ không đi sâu trình bày thuật toán về phương pháp tối ưu truy vấn dựa trên việc ước lượng chi phí. Tuy nhiên, ví dụ dưới đây sẽ trình bày một số khả năng tính chi phí cho một số chiến lược thực thi truy vấn nhằm minh họa cho phương pháp này.

III.3 Ví dụ

Quan hệ GIÁOVIÊN có số dòng $r = 10000$, lưu trên số khối $b = 2000$ khối (như vậy có 5 bộ trên mỗi khối $b_{fr} = 5$). Có 1 chỉ mục thứ cấp trên thuộc tính MÃGV có số mức $x_{MÃGV} = 4$. Có một chỉ mục thứ cấp trên thuộc tính MÃBM với số mức $x_{MÃBM} = 2$, số block của mức chỉ mục đầu tiên là $b_{1MÃBM} = 4$. Có 125 giá trị khác nhau cho MÃBM $d_{MÃBM} = 125$, nên $s_{MÃBM} = r/d_{MÃBM} = 8$.

Ví dụ 1: Tìm giáo viên có mã là 002.

Đại số quan hệ: $\sigma_{MÃGV} = '002'$ (GIÁOVIÊN)

Chi phí duyệt toàn bộ bảng để tìm thấy tất cả các dòng thỏa điều kiện chọn: $C1 = b = 2000$.

Chi phí trung bình khi chọn trên thuộc tính khóa: $C2 = b/2 = 1000$.

Chi phí khi dùng chỉ mục thứ cấp trên thuộc tính MÃGV: $C3 = x_{MÃGV} + 1 = 5$.

Vậy $C3$ là chiến lược thực thi có chi phí thấp nhất.

Ví dụ 2: Cho danh sách giáo viên thuộc bộ môn mã là HTTT.

Đại số quan hệ: $\sigma_{MÃBM} = 'HTTT'$ (GIÁOVIÊN)

Chi phí duyệt toàn bộ bảng để tìm thấy tất cả các dòng thỏa điều kiện chọn: $C1 = b = 2000$.

Chi phí khi dùng chỉ mục thứ cấp trên thuộc tính MÃBM: $C2 = x_{MÃBM} + (b_{1MÃBM}/2) + r/2 = 2 + (4/2) + (10000/2) = 5004$.

Vậy $C1$ là chiến lược thực thi có chi phí thấp hơn.

IV. KẾT LUẬN

Chương 9 trình bày các phương pháp tối ưu truy vấn. Phương pháp đầu tiên dựa trên các luật heuristic để làm giảm kích thước kết quả trung gian trong quá trình thực hiện truy vấn. Phương pháp thứ hai tính chi phí cho các chiến lược thực thi khác nhau của một câu truy vấn và chọn chiến lược có chi phí thấp nhất. Cả hai phương pháp

thường được kết hợp vận dụng trong một HQT CSDL nhằm phục vụ cho mục đích tối ưu truy vấn.

Câu hỏi ôn tập

1. Tối ưu truy vấn là gì? Vì sao HQT CSDL phải thực hiện tối ưu truy vấn?
2. Vì sao HQT CSDL phải chuyển câu truy vấn từ ngôn ngữ SQL sang ngôn ngữ đại số quan hệ trước khi thực hiện tối ưu truy vấn?
3. Hãy phân biệt hai phương pháp tối ưu: dựa trên các luật heuristic và ước lượng chi phí.
4. Hãy phân biệt chiến lược thực thi mức logic và chiến lược thực thi mức vật lý? Cho mỗi loại một ví dụ minh họa.
5. Hãy giải thích ý nghĩa tối ưu đối với từng luật heuristic. Hãy kể ra 3 luật tối ưu thường dùng nhất.

Bài tập

1. Cho lược đồ CSDL như trong bài tập 1 chương 5, có các nhu cầu truy vấn sau:
 - 1.1 Danh sách các giáo viên nữ thuộc bộ môn Hệ thống thông tin. Kết quả gồm MÃGV, HỌTÊN.
 - 1.2 Cho danh sách các giáo viên thuộc bộ môn mã là 'HTTT' đã tham gia các đề tài cấp trường. Kết quả gồm MÃGV, HỌTÊN.
 - 1.3 Danh sách các giáo viên thuộc bộ môn 'Hệ thống thông tin' sinh trước năm 1975 đã tham gia các đề tài có kinh phí lớn hơn 50 triệu. Kết quả gồm MÃGV, HỌTÊN.
 - 1.4 Các đề tài thuộc chủ đề 'Quản lý giáo dục' do giáo viên thuộc bộ môn 'Hệ thống thông tin' làm chủ nhiệm gồm có công việc tên là 'Thiết kế hệ thống'. Kết quả gồm MÃĐT, TÊNĐT.
- a) Hãy thực hiện các nhu cầu truy vấn trên dùng ngôn ngữ SQL.
- b) Với mỗi nhu cầu truy vấn hãy viết 2 biểu thức đại số quan hệ khác nhau. Dùng cây truy vấn thể hiện lại 2 biểu thức đại số quan hệ này.
- c) Cho biết cây truy vấn ban đầu đối với từng nhu cầu truy vấn. Thực hiện tối ưu dần dùng các luật heuristic để cho ra cây truy vấn tối ưu cuối cùng.

Chương 9 Tối ưu truy vấn

- d) So sánh hai cây truy vấn có được ở câu b. với cây truy vấn ban đầu và cây truy vấn cuối cùng ở câu c.

BẢNG TRA TÙ

ALTER TABLE, 113

Assertion, 172

Bản số, 38

Bộ, 49

Cây truy vấn, 182

Dạng chuẩn

Dạng chuẩn 1, 67

Dạng chuẩn 2, 68

Dạng chuẩn 3, 70

Dạng chuẩn BCK, 72

DELETE, 146

DROP TABLE, 115

Dữ liệu, 13

Hệ cơ sở dữ liệu, 14

Hệ quản trị cơ sở dữ liệu, 14

INSERT INTO, 144

Khóa của lược đồ quan hệ, 52

Khóa ngoại, 53

Lược đồ ngoài, 23

Lược đồ quan hệ, 49

Lược đồ quan niệm, 23

Lược đồ trong, 23

Mệnh đề ORDER BY, 127

Mô hình dữ liệu, 26
Mô hình dữ liệu cấp cao, 32
MÔ HÌNH DỮ LIỆU QUAN HỆ, 49
Mô hình dữ liệu quan niệm, 32
Mô hình thực thể kết hợp, 26, 32
Mối kết hợp, 35
Ngôn ngữ của hệ QTCSQL, 23
Ngôn ngữ đại số quan hệ, 80
Ngôn ngữ định nghĩa dữ liệu, 107
CREATE TABLE, 107
Kiểu dữ liệu, 108
Ngôn ngữ SQL, 106
Bí danh, 118
Hàm tính tổng hợp, 140
Mệnh đề GROUP BY, 141
Mệnh đề HAVING, 141
SELECT-FROM-WHERE, 116
So sánh chuỗi, 124
Truy vấn lồng, 133
Người dùng cuối, 22
Phép tính quan hệ, 150
Biến bộ, 151
Biến kết buộc, 154
Biến tự do, 154
Lượng tử tồn tại, 153
Lượng tử với mọi, 153

- Phép tính quan hệ có biến là bộ*, 150
Phép tính quan hệ có biến là miền, 150, 157
Quan hệ miền, 151
Phép toán đại số quan hệ, 80
 Hàm tính tổng hợp và gom nhóm, 94
 Phép chia, 92
 Phép chiếu, 85
 Phép chọn, 84
 Phép đổi tên, 87
 Phép gán, 87
 Phép giao, 82
 Phép hiệu, 82
 Phép hội, 81
 Phép kết, 88
 Kết bằng, 89
 Kết ngoài, 89
 Kết tự nhiên, 89
 Phép tích Đề-các, 83
 Sửa, 96
 Thêm, 95
 Xóa, 96
Phụ thuộc hàm, 64
 Phụ thuộc hàm đầy đủ, 65
Quan hệ, 49
Quản trị viên, 21
RBTV, 162

RBTV liên bộ trên một quan hệ, 166
RBTV liên bộ trên nhiều quan hệ, 169
RBTV liên thuộc tính liên quan hệ, 170
RBTV liên thuộc tính trên một quan hệ, 167
RBTV miền giá trị, 165
RBTV trên thuộc tính tổng hợp, 171
RBTV tham chiếu, 168
Siêu khóa, 52
Tập thực thể, 32
Thiết kế viên, 21
Thuộc tính, 33, 39, 50
Thuộc tính tổng hợp, 171
Tối ưu truy vấn, 181
Phương pháp dựa trên heuristic, 182
Phương pháp ước lượng chi phí, 190
Trigger, 173
Trùng lắp thông tin, 66
UPDATE, 146
Vai trò, 39

TÀI LIỆU THAM KHẢO

- 1) Codd E.F. A Relational Model for Large Shared Data Banks. Communications of the ACM, 13,6 (June 1970) pp 377-387.
- 2) Chen, P.P.S. The Entity-Relationship Model - Towards a Unified View of Data. ACM TODS, Vol.1, No.1, March 1976.
- 3) Ramez Elmasri and Shamkant B. Navathe. Fundamentals of Database Systems, fourth edition, Addison-Wesley, 2004. ISBN 0-321-12226-7.
- 4) Jim Melton, Alan R. Simon. Understanding the New SQL: a complete guide. Morgan Kauffmann, 1993.
- 5) Jeffrey A. Hoffer, Mary B. Prescott, Fred R. McFadden. Modern Database Management, sixth edition, 2002.

**ĐỒNG THỊ BÍCH THỦY – PHẠM THỊ BẠCH HUỆ
NGUYỄN TRẦN MINH THƯ**

GIÁO TRÌNH CƠ SỞ DỮ LIỆU

Chịu trách nhiệm xuất bản:

VÕ TUẤN HẢI

<i>Biên tập</i>	: TRƯƠNG THANH SƠN
	: PHẠM THỊ MAI
<i>Thiết kế bìa</i>	: VŨ QUỐC HOÀNG
<i>Trình bày</i>	: PHẠM THỊ MAI
<i>Sửa bản in</i>	: PHẠM THỊ MAI

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

70 Trần Hưng Đạo – Quận Hoàn Kiếm – Hà Nội

ĐT: (024) 3942 2443 Fax: (024) 3822 0658

Website: <http://www.nxbkhkt.com.vn> - Email: nxbkhkt@hn.vnn.vn

CHI NHÁNH NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

28 Đồng Khởi, 12 Hồ Huân Nghiệp – Quận 1 – TP. Hồ Chí Minh

ĐT: (028) 3822 5062 Fax: (028) 3829 6628

Email: chinhhanhnxbkhkt@yahoo.com.vn

In 1.000 bản, khổ 16^{cm} × 24^{cm} tại CÔNG TY TNHH MTV IN KINH TẾ.

Địa chỉ: Số 279 Nguyễn Tri Phương - P.5 - Q.10 - Thành phố Hồ Chí Minh.

Số ĐKXB: 1616- 2020/CXBIPH/7-40/KHKT.

Quyết định XB số: 173/QĐ-NXBKHKT, ngày 31/08/2020.

Mã ISBN: 978-604-67-1572-6.

In xong và nộp lưu chiểu quý IV năm 2020.

TÓM TẮT GIÁO TRÌNH

Giáo trình Cơ sở Dữ liệu đã được viết để phục vụ cho sinh viên theo học môn Cơ sở Dữ liệu tại Khoa Công nghệ Thông tin Trường Đại học Khoa học Tự nhiên (Đại học Quốc gia Thành phố Hồ Chí Minh).

Nội dung giáo trình và cách thức chúng tôi trình bày nội dung này bám sát mục tiêu đề ra cho môn học. Đó là cung cấp kiến thức căn bản nhất về mô hình dữ liệu quan hệ, các ngôn ngữ truy vấn và các nguyên tắc để tối ưu hóa câu truy vấn. Đối với hai cụm kiến thức đầu, chúng tôi tập trung trình bày sao cho bạn đọc nắm bắt được và biết sử dụng ở mức độ người dùng. Cụ thể là bạn đọc sẽ hiểu được một lược đồ cơ sở dữ liệu quan hệ, từ đó có thể phát biểu các ràng buộc toàn vẹn và đánh giá chất lượng lược đồ cơ sở dữ liệu liên quan đến yêu tố trùng lặp thông tin.

Để có thể khai thác được một cách cụ thể với một phần mềm quản trị cơ sở dữ liệu, ngôn ngữ truy vấn SQL cũng được trình bày. Nội dung liên quan đến tối ưu hóa câu truy vấn được trình bày ở mức độ nhập môn, để bạn đọc hiểu được cách thức một hệ quản trị cơ sở dữ liệu thực hiện một câu truy vấn.



220109-H00

ISBN: 978-604-67-1572-6



9 786046 715726

Giá: 119.000đ