

Lý thuyết thiết kế cơ sở dữ liệu quan hệ



Giới thiệu

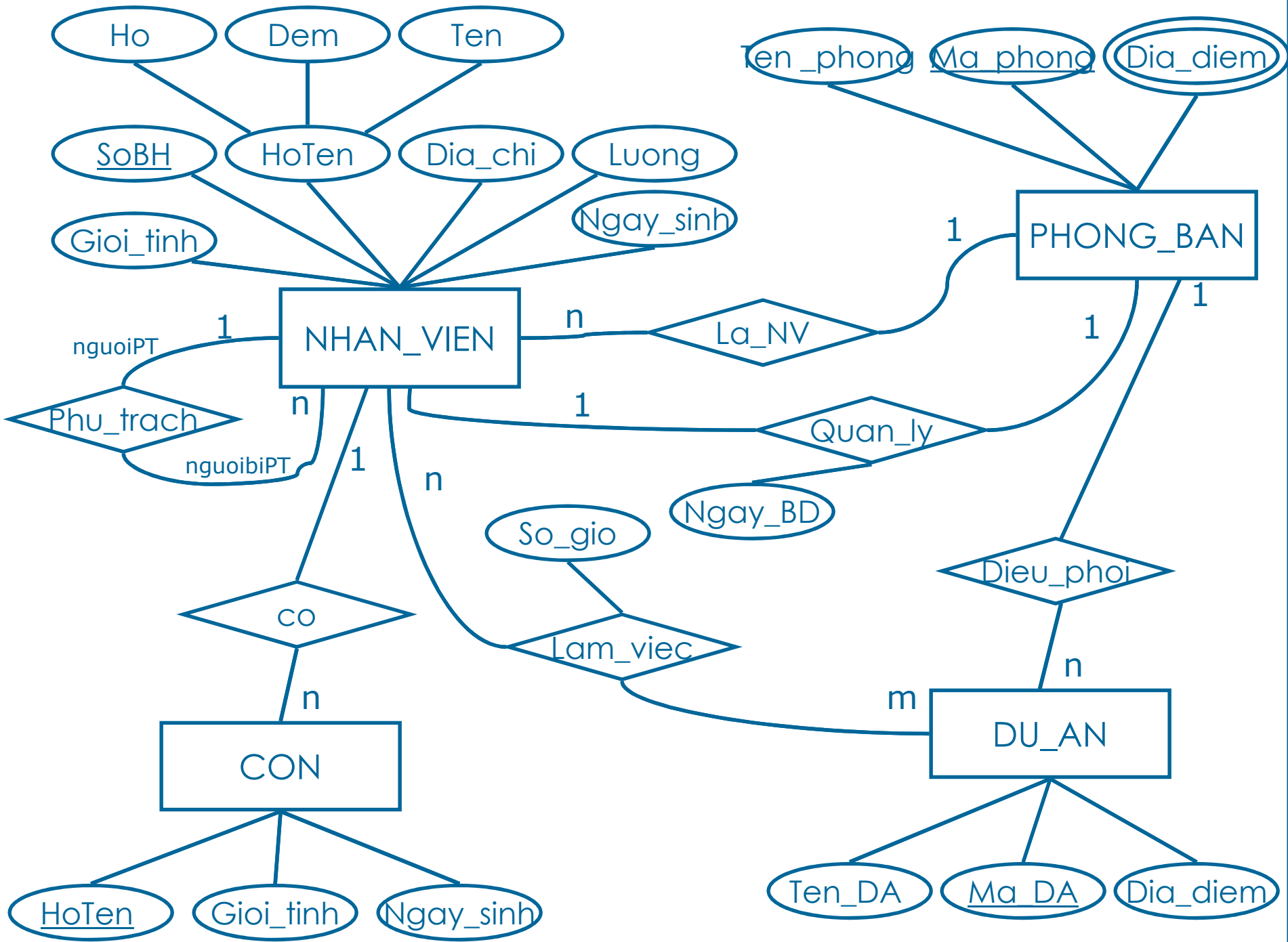
- 2 cách tiếp cận đ/v thiết kế CSDL quan hệ
 - Top - down: Sử dụng mô hình thực thể liên kết (ERD)
 - Bottom - up: Sử dụng các phụ thuộc hàm & chuẩn hóa
- Tiếp cận top – down trong thiết kế CSDL quan hệ
 - Thiết lập **ERD** từ các yêu cầu trong thế giới thực của xí nghiệp
 - Biến đổi ERD thành sơ đồ quan hệ

Tổng quan về tiếp cận top-down đ/v thiết kế CSDLQH

- Thiết lập sơ đồ thực thể - liên kết
 - Bước 1: Xác định các thực thể và các thuộc tính
 - Bước 2: Xác định các liên kết giữa các thực thể
 - Bậc của liên kết
 - Kiểu của liên kết (1-1, 1-n, n-m, đệ quy)

Thiết lập sơ đồ ER

- Bài toán: phân tích và thiết kế 1 CSDL gồm các thông tin trong 1 công ty (nhân viên, phòng ban, dự án)
 - Công ty được tổ chức bởi các phòng ban. Mỗi phòng ban có 1 tên duy nhất, 1 số duy nhất và 1 người quản lý (thời điểm bắt đầu công tác quản lý của người này cũng được lưu lại trong CSDL). Mỗi phòng ban có thể có nhiều trụ sở làm việc khác nhau
 - Mỗi phòng điều phối một số dự án. Mỗi dự án có 1 tên và 1 mã số duy nhất, thực hiện tại một địa điểm duy nhất
 - Các thông tin về nhân viên cần được quan tâm gồm: tên, số bảo hiểm, địa chỉ, lương, giới tính, ngày sinh. Mỗi nhân viên làm việc tại một phòng ban nhưng có thể tham gia nhiều dự án khác nhau. Những dự án này có thể được điều phối bởi các phòng ban khác nhau. Thông tin về số giờ làm việc trong từng dự án (theo tuần) cũng như người quản lý trực tiếp của các nhân viên cũng được lưu trữ
 - Thông tin về con cái của từng nhân viên: tên, giới tính, ngày sinh

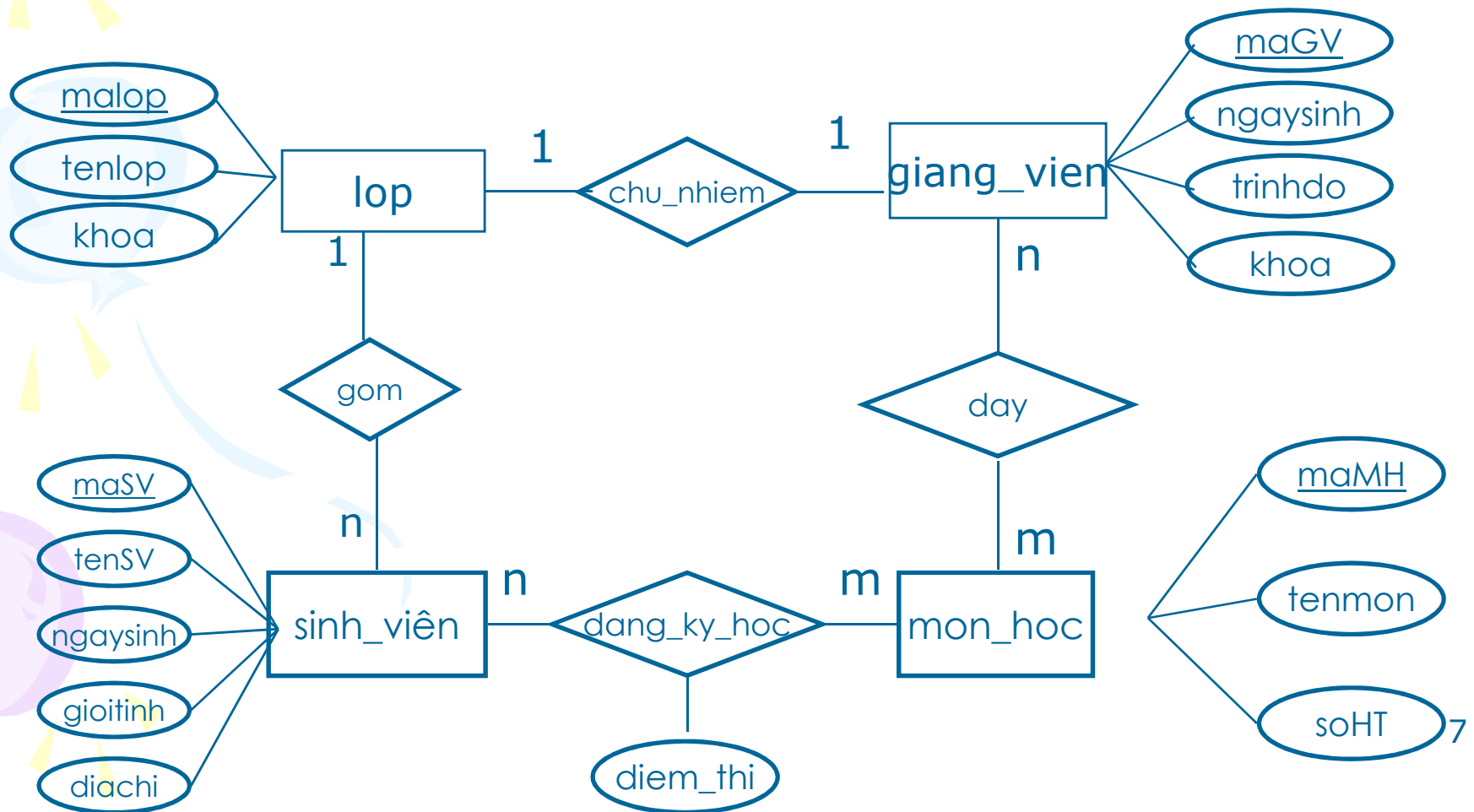


Biến đổi sơ đồ thực thể liên kết sang sơ đồ quan hệ

- Biến đổi tập các thực thể
- Xác định khóa của tập các thực thể và của các quan hệ
- Biến đổi các liên kết
- Rút gọn với các quan hệ có khoá chung

Ví dụ minh họa

Cho sơ đồ thực thể liên kết



Sơ đồ cơ sở dữ liệu quan hệ kết quả

MON_HOC

maMH	tenmon	soHT
CNTT01	Nhập môn CSDL	4
CNTT02	Truyền DL và mạng	4
CNTT03	Phân tích và thiết kế hệ thống	4
HTTT01	Quản lý dự án	3

LOP

malop	tenlop	khoa	GVCN
IT4	Tin 4	CNTT	Ng. V. Anh
IT5	Tin 5	CNTT	Lê A. Văn
IT6	Tin 6	CNTT	Ng. T. Thảo
IT7	Tin 7	CNTT	Ng. V. Quý

SINH_VIEN

maSV	tenSV	ngaysinh	gt	diachi	malop
SV0011	Trần T. Bình	1/4/1981	0	21 T. Q. B	IT4
SV0025	Ng. Đ. Trung	3/2/1980	1	56 Đ. C. V	IT5
SV0067	Trần M. Quế	26/3/1982	0	45 H. B. T	IT6
SV0034	Ng. T. Phương	29/2/1980	0	86 L. T. N	IT7

GIẢNG-VIÊN

maGV	tenGV	trinhdo	khoa
GV0011	Trần T. Hà	Thạc sĩ	CNTT
GV0022	Ng. Đ. Tung	Tiến sĩ	CNTT
GV0065	Trần M. Quang	Tiến sĩ	CNTT

ĐĂNG_KÝ_HỌC

maSV	maMH	diem_thi
SV0011	CNTT01	7
SV0025	CNTT02	8
SV0067	CNTT03	6
SV0034	HTTT01	9



MH_GV

maMH	maGV
CNTT01	GV0011
CNTT01	GV0022
CNTT03	GV0065

Tổng quan về tiếp cận bottom-up đ/v thiết kế CSDLQH

- Các vấn đề liên quan đến 1 thiết kế CSDLQH:

Giả sử ta cần thiết kế một cơ sở dữ liệu lưu trữ thông tin về các hãng cung ứng. Sơ đồ quan hệ được thiết kế trong đó tất cả các thuộc tính cần thiết được lưu trong đúng 1 quan hệ với **sơ đồ quan hệ**:

Suppliers(sid, sname, city, status, product, quantity)

Suppliers

sid	sname	city	status	product	quantity
S1	Smith	London	100	Screw	50
S1	Smith	London	100	Nut	100
S2	J&J	Paris	124	Screw	78
S3	Blake	Tokyo	75	Bolt	100

Tổng quan về tiếp cận bottom-up đ/v thiết kế CSDLQH

- **Dư thừa dữ liệu:** Hãng nào cung ứng nhiều hơn 1 mặt hàng thì thông tin của hãng đó sẽ bị lặp lại trong bảng (VD S1), mặt hàng được cung ứng bởi nhiều hãng cũng bị lặp lại (VD Screw)
- **Dị thường dữ liệu khi thêm:** Nếu có một hãng chưa cung cấp mặt hàng nào, vậy giá trị cho thuộc tính product và quantity trong bộ dữ liệu mới được thêm vào sẽ không được xác định
- **Dị thường dữ liệu khi xóa:** Nếu một hãng chỉ cung cấp 1 mặt hàng, nếu ta muốn xóa thông tin về sự cung cấp này thì ta sẽ mất thông tin về hãng cung cấp
- **Dị thường dữ liệu khi sửa đổi:** Do thông tin bị lặp lại nên việc sửa đổi 1 bộ dữ liệu có thể dẫn đến vấn đề không nhất quán trong dữ liệu, ví dụ như về một hãng nếu sơ sót không sửa đổi trên toàn bộ các bộ giá trị liên quan đến hãng đó
- Nguyên nhân chính gây nên các dị thường là sự dư thừa dữ liệu hay **sự phụ thuộc giữa các tập thuộc tính**

Tổng quan về tiếp cận bottom-up đ/v thiết kế CSDLQH

- Thay thiết kế trên với 1 sơ đồ quan hệ
Suppliers(sid, sname, city, status, product, quantity)
bởi 2 sơ đồ quan hệ
 - *Supp(sid, sname, city, status)*
 - *Supply(sid, product, quantity)*

Thực hiện phép tách quan hệ Suppliers thành 2 quan hệ Supp và Supply với các sơ đồ quan hệ tương ứng, tất cả các vấn đề nêu ở trên đã được loại bỏ. Tuy nhiên, khi tìm kiếm dữ liệu thì chúng ta phải thực hiện kết nối 2 bảng chứ không chỉ là chọn và chiếu trên 1 bảng như ở cách thiết kế trước

- Thiết kế sau với 2 sơ đồ quan hệ 'tốt hơn' thiết kế đầu với 1 sơ đồ quan hệ
- Phép tách quan hệ cho phép giảm dư thừa và tránh các dị thường cập nhật

Mục đích của thiết kế CSDLQH

- Xác định được 1 tập các sơ đồ quan hệ cho phép tìm kiếm thông tin một cách dễ dàng, **đồng thời** tránh được dư thừa dữ liệu
- Hướng tiếp cận: Một trong những kỹ thuật được sử dụng là Tách các sơ đồ quan hệ 'chưa tốt' thành những sơ đồ quan hệ 'tốt hơn'. Sự phụ thuộc giữa các tập thuộc tính có thể được sử dụng để nhận biết các sơ đồ 'chưa tốt' và cho phép thực hiện phép tách để thu được những sơ đồ quan hệ 'tốt hơn'.
- Xét 2 loại phụ thuộc dữ liệu:
 - phụ thuộc hàm và
 - phụ thuộc đa trị

Phụ thuộc hàm

- *Định nghĩa 1*: Cho $R(U)$ là một sơ đồ quan hệ với U là tập thuộc tính $\{A_1, A_2, \dots, A_n\}$. X, Y là tập con của U . Nói rằng X xác định Y hay Y là phụ thuộc hàm vào X ($X \rightarrow Y$) nếu với 1 quan hệ r xác định trên $R(U)$ và 2 bộ bất kỳ t_1, t_2 thuộc r mà $t_1[X] = t_2[X]$ thì ta có $t_1[Y] = t_2[Y]$

Ví dụ

- Ví dụ 1:

A	B	C
a1	b1	c1
a2	b2	c2
a3	b1	c1
a4	b3	c2

- $A \rightarrow B, A \rightarrow C, B \rightarrow C$
- Ví dụ 2: trong thiết kế CSDLQH đầu chương, ta có quan hệ Suppliers, với mỗi giá trị của sid đều xác định một giá trị duy nhất đối với các thuộc tính: sname, city và status. Do đó ta có $sid \rightarrow sname, sid \rightarrow city, sid \rightarrow status$

Hệ tiên đề Amstrong đối với phụ thuộc hàm

Cho

- $R(U)$ là 1 sơ đồ quan hệ, U là tập các thuộc tính.
- $X, Y, Z, W \subseteq U$, ký hiệu: $XY = X \cup Y$

Hệ tiên đề Amstrong :

- Phản xạ (*reflexivity*)

Nếu $Y \subseteq X$ thì $X \rightarrow Y$

- Tăng trưởng (*augmentation*)

Nếu $X \rightarrow Y$ thì $XZ \rightarrow YZ$

- Bắt cầu (*transitivity*)

Nếu $X \rightarrow Y, Y \rightarrow Z$ thì $X \rightarrow Z$

Hệ tiên đề Amstrong đối với phụ thuộc hàm

- **Bổ đề 1:** Cho $R(U)$ là 1 sơ đồ quan hệ thỏa mãn Hệ tiên đề Amstrong đối với phụ thuộc hàm. Với $X, Y, Z, W \subseteq U$, ký hiệu: $XY = X \cup Y$. Các luật sau là đúng dẫn:

- Luật hợp (*union*)

Nếu $X \rightarrow Y, X \rightarrow Z$ thì $X \rightarrow YZ$.

- Luật tựa bắc cầu (*pseudotransitivity*)

Nếu $X \rightarrow Y, WY \rightarrow Z$ thì $XW \rightarrow Z$.

- Luật tách (*decomposition*)

Nếu $X \rightarrow Y, Z \subseteq Y$ thì $X \rightarrow Z$

Ví dụ

- Ví dụ 1:

Cho tập phụ thuộc hàm $\{AB \rightarrow C, C \rightarrow A\}$

Chứng minh: $BC \rightarrow ABC$

$$C \rightarrow A$$

$$AB \rightarrow C$$

$$BC \rightarrow AB, AB \rightarrow ABC$$

$$BC \rightarrow AB$$

$$AB \rightarrow ABC$$

$$BC \rightarrow ABC$$

- Ví dụ 2:

Cho sơ đồ quan hệ $R(ABEIJGH)$ và tập phụ thuộc hàm $F = \{AB \rightarrow E, AG \rightarrow J, BE \rightarrow I, E \rightarrow G, GI \rightarrow H\}$

Chứng minh: $AB \rightarrow GH$

Bao đóng của một tập phụ thuộc hàm

- **Định nghĩa 2:** Cho F là một tập phụ thuộc hàm. Bao đóng của F ký hiệu là F^+ là tập tất cả các phụ thuộc hàm có thể được suy ra từ các phụ thuộc hàm trong F
 - Số các phụ thuộc hàm trong bao đóng của một tập phụ thuộc hàm có thể rất lớn.
 - Vấn đề tính bao đóng của 1 tập phụ thuộc hàm là không thực tế.
 - Vấn đề kiểm tra một phụ thuộc hàm $X \rightarrow Y$ được suy diễn ra từ một tập phụ thuộc hàm cho trước '**không nhất thiết**' phải tính Bao đóng của 1 tập phụ thuộc hàm mà chỉ cần quan tâm tới X và những thuộc tính có thể được suy diễn ra từ X

Bao đóng của một tập thuộc tính đối với tập phụ thuộc hàm

- **Định nghĩa 3:** Cho một sơ đồ quan hệ $R(U)$, F là một tập phụ thuộc hàm trên U . X là tập con của U . Bao đóng của tập thuộc tính X ký hiệu là X^+ là tập tất cả các thuộc tính được xác định hàm bởi X thông qua tập F

$$X^+ = \{A \in U \mid X \rightarrow A \in F^+\}$$

- Mặc dù định nghĩa về bao đóng của một tập thuộc tính dựa trên bao đóng của tập phụ thuộc hàm, trên thực tế, chúng ta có thể tính bao đóng của một tập thuộc tính khá dễ dàng với 1 thuật toán đơn giản.

Bao đóng của một tập thuộc tính đối với tập phụ thuộc hàm

- **Thuật toán 1:** Tìm bao đóng của một tập thuộc tính đối với tập phụ thuộc hàm
 - **Vào:** Tập hữu hạn các thuộc tính U , tập các phụ thuộc hàm F trên U , $X \subseteq U$
 - **Ra:** X^+
 - **Phương pháp:**
 - B^0** $X^0 = X$
 - B^i** Tính X^i từ X^{i-1}
 - Nếu $\exists Y \rightarrow Z \in F$ và $Y \subseteq X^{i-1}$ và $A \in Z$ và $A \notin X^{i-1}$
thì $X^i = X^{i-1} \cup A$
 - ngược lại, $X^i = X^{i-1}$
 - Nếu $X^i \neq X^{i-1}$
thì lặp B^i
 - ngược lại, chuyển B^n
 - B^n** $X^+ = X^i$

Bao đóng của một tập các thuộc tính đối với tập các phụ thuộc hàm

- Bổ đề 2: $X \rightarrow Y$ được suy diễn từ hệ tiên đề Armstrong khi và chỉ khi $Y \subseteq X^+$
- Chứng minh:
 - Giả sử $Y = A_1 \dots A_n$, với A_1, \dots, A_n là các thuộc tính và $Y \subseteq X^+$
 - Từ định nghĩa X^+ ta có $X \rightarrow A_i$. Áp dụng tiên đề Armstrong cho mọi i , suy ra $X \rightarrow Y$ nhờ luật hợp.
 - Ngược lại, giả sử có $X \rightarrow Y$, áp dụng hệ tiên đề Armstrong cho mỗi i , ta có $X \rightarrow A_i$, $A_i \in Y$ nhờ luật tách. Từ đó suy ra $Y \subseteq X^+$

Ví dụ

- Cho $R(U)$, $U = \{A, B, C, D, E, F\}$
 $F = \{AB \rightarrow C, BC \rightarrow AD, D \rightarrow E, CF \rightarrow B\}$

Tính $(AB)^+$

- Thực hiện:
 - Bước 0: $X^0 = AB$
 - Bước 1: $X^1 = ABC$ (do $AB \rightarrow C$)
 - Bước 2: $X^2 = ABCD$ (do $BC \rightarrow AD$)
 - Bước 3: $X^3 = ABCDE$ (do $D \rightarrow E$)
 - Bước 4: $X^4 = ABCDE$

Các khoá đối với SĐQH

- **Định nghĩa 4:** Cho sơ đồ quan hệ $R(U)$, F là một tập các phụ thuộc hàm xác định trên U . K là một tập con của U , K được gọi là khoá tối thiểu của R nếu như
 - $K \rightarrow U$ là một phụ thuộc hàm trong F^+
 - Với mọi tập con thực sự K' của K thì $K' \rightarrow U$ không thuộc F^+
 - K là một khoá tối thiểu thì $K^+ = U$ và K là tập thuộc tính nhỏ nhất có tính chất như vậy

Các khoá đối với SĐQH

- Thuật toán 2: Tìm khoá tối thiểu
 - **Vào:** $U = \{A_1, A_2, \dots, A_n\}$, F
 - **Ra:** khoá tối thiểu K xác định được trên U và F

– Phương pháp

$$\mathbf{B^0} \quad K^0 = U$$

$$\mathbf{B^i} \quad \text{Nếu } (K^{i-1} \setminus \{A_i\}) \rightarrow U$$
$$\text{thì} \quad K^i = K^{i-1} \setminus \{A_i\}$$

$$\text{ngược lại,} \quad K^i = K^{i-1}$$

$$\mathbf{B^{n+1}} \quad K = K^n$$

Ví dụ

- Cho $U = \{A, B, C, D, E\}$
- $F = \{AB \rightarrow C, AC \rightarrow B, BC \rightarrow DE\}$. Tìm một khoá tối thiểu của một quan hệ r xác định trên U và F
- Thực hiện
- B0: $K^0 = U = ABCDE$
- B1: Kiểm tra xem có tồn tại phụ thuộc hàm $(K^0 \setminus \{A\}) \rightarrow U$ ($BCDE \rightarrow U$) hay không. Ta cần phải sử dụng thuật toán 1 để kiểm tra điều kiện tương đương là $(BCDE)^+ = U$ hay không. $(BCDE)^+ = BCDE$, khác U . Vậy $K^1 = K^0 = ABCDE$
- B2: Tương tự, thử loại bỏ B ra khỏi K^1 ta có $(ACDE)^+ = ABCDE = U$. Vậy $K^2 = K^1 \setminus \{B\} = ACDE$
- B3: $K^3 = ACDE$
- B4: $K^4 = ACE$
- B5: $K^5 = AC$
- Vậy AC là một khoá tối thiểu mà ta cần tìm

Nhận xét về phụ thuộc hàm

- Từ một tập các phụ thuộc hàm có thể suy diễn ra các phụ thuộc hàm khác
 - Trong một tập phụ thuộc hàm cho trước có thể có các phụ thuộc hàm bị coi là dư thừa
- Làm thế nào để có được một tập phụ thuộc hàm tốt?

Phủ của tập các phụ thuộc hàm

- **Định nghĩa 5:** Hai tập phụ thuộc hàm F và G xác định trên U là **tương đương**, ký hiệu là $F \approx G$, nếu $F^+ = G^+$.
 - Khi F và G là **tương đương**, chúng ta nói F là **phủ** của G hay G là **phủ** của F .
 - Có thể kiểm tra tính tương đương của 2 tập phụ thuộc hàm được không?

Phủ của tập các phụ thuộc hàm

- **Thuật toán 3:** Kiểm tra tính tương đương của 2 tập phụ thuộc hàm
 - **Vào:** 2 tập phụ thuộc hàm F và G xác định trên U
 - **Ra:** $F \approx G$?
 - **Phương pháp**
 - B.1. Với mỗi phụ thuộc hàm $Y \rightarrow Z \in F$, kiểm tra $Y \rightarrow Z \in G^+$? ($Z \subseteq Y^+$ với Y^+ được tính trên G)
Nếu với \forall phụ thuộc hàm $f \in F$, $f \in G^+$ thì $F^+ \subseteq G^+$
 - B.2. Tương tự, nếu \forall phụ thuộc hàm $g \in G$, $g \in F^+$ thì $G^+ \subseteq F^+$
 - B.3. Nếu $F^+ \subseteq G^+$ và $G^+ \subseteq F^+$ thì $F \approx G$

Ví dụ

- Cho sơ đồ quan hệ $R(U)$ với $U = \{A, B, C, D, E, F\}$
 $F = \{AB \rightarrow C, D \rightarrow EF, C \rightarrow BD\}$
 $G = \{AC \rightarrow B, D \rightarrow EF, B \rightarrow CD\}$

Hỏi F và G có phải là 2 tập pth tương đương hay không?

- Thực hiện:

Đối với các phụ thuộc hàm trong F

- $f_1 = AB \rightarrow C$. AB^+ (đối với G) = $ABCDEF = U$. Vậy f_1 thuộc G^+
- $f_2 = D \rightarrow EF$ thuộc G nên chắc chắn thuộc G^+
- $f_3 = C \rightarrow BD$. C^+ (đối với G) = C không chứa BD . Vậy f_3 không thuộc G^+
- Kết luận F không tương đương với G

Tập phụ thuộc hàm không dư thừa

- **Định nghĩa 6:** Tập phụ thuộc hàm F là không dư thừa nếu không $\exists X \rightarrow Y \in F$ sao cho $F \setminus \{X \rightarrow Y\} \approx F$.
- **Thuật toán 4:** Tìm phủ không dư thừa của 1 tập phụ thuộc hàm
 - Vào: Tập thuộc tính U , $F = \{L_i \rightarrow R_i : i = 1..n\}$
 - Ra : Phủ không dư thừa F' của F
 - Phương pháp
 - B⁰** $F^0 = F$
 - Bⁱ** Nếu $F^{i-1} \setminus \{L_i \rightarrow R_i\} \approx F^{i-1}$
thì $F_i = F^{i-1} \setminus \{L_i \rightarrow R_i\}$
ngược lại, $F_i = F^{i-1}$
 - Bⁿ⁺¹** $F' = F^n$

Tập phụ thuộc hàm tối thiểu

- *Định nghĩa 7* : F được gọi là **tập phụ thuộc hàm tối thiểu** nếu F thỏa mãn 3 điều kiện sau:

Đk1: Với $\forall f \in F$, f có dạng $X \rightarrow A$,
trong đó A là 1 thuộc tính

Đk2: Với $\forall f = X \rightarrow Y \in F$, $\nexists A \in X$ (A là 1 thuộc tính): $(F \setminus f) \cup \{(X \setminus A) \rightarrow Y\} \approx F$

Đk3: $\nexists X \rightarrow A \in F : F \setminus \{X \rightarrow A\} \approx F$

Tập phụ thuộc hàm tối thiểu

- **Thuật toán 5:** Tìm phủ tối thiểu của một tập phụ thuộc hàm

- **Vào:** Tập thuộc tính U , $F = \{L_i \rightarrow R_i : i = 1..n\}$
- **Ra:** phủ tối thiểu F_c của tập phụ thuộc hàm F
- **Phương pháp**

- B.1. Biến đổi F về dạng $F_1 = \{L_i \rightarrow A_j\}$**

trong đó A_j là 1 thuộc tính bất kỳ thuộc U (thoả mãn đk1)

- B.2. Loại bỏ thuộc tính thừa trong vế trái của các phụ thuộc hàm**

Lần lượt giản ước từng thuộc tính trong vế trái của từng phụ thuộc hàm trong F_1 thu được F_1' . Nếu $F_1' \approx F_1$ thì loại bỏ thuộc tính đang xét

Khi không có sự giản ước nào xảy ra nữa ta thu được F_2 thoả mãn đk2

- B.3. Loại bỏ phụ thuộc hàm dư thừa**

Lần lượt kiểm tra từng phụ thuộc hàm f . Nếu $F_2 \setminus f \approx F_2$ thì loại bỏ f

Khi không còn phụ thuộc hàm nào có thể loại bỏ thì thu được F_3 thoả mãn đk3

- B.4. $F_c = F_3$**

Ví dụ 1

- $U = \{A, B, C\}$

$F = \{A \rightarrow BC, B \rightarrow C, A \rightarrow B, AB \rightarrow C\}$. Tìm phủ tối thiểu của F ?

- $F_1 = \{A \rightarrow B, A \rightarrow C, B \rightarrow C, AB \rightarrow C\}$

- Xét các pth trong F_1 mà vế trái có nhiều hơn 1 thuộc tính $AB \rightarrow C$. Giản ước A thì ta còn $B \rightarrow C$ có trong F_1 , vậy A là thuộc tính thừa. Tương tự ta cũng tìm được B là thừa, vậy loại bỏ luôn $AB \rightarrow C$ khỏi F_1 . $F_2 = \{A \rightarrow B, A \rightarrow C, B \rightarrow C\}$

- Bỏ pth thừa: $A \rightarrow C$ là thừa.

Vậy $F_c = \{A \rightarrow B, B \rightarrow C\}$

Ví dụ 2

- Tìm phủ tối thiểu của tập phụ thuộc hàm
 $F = \{A \rightarrow B, ABCD \rightarrow E, EF \rightarrow G, ACDF \rightarrow EG\}$
 - $F_1 = \{A \rightarrow B, ABCD \rightarrow E, EF \rightarrow G, ACDF \rightarrow E, ACDF \rightarrow G\}$
 - **Loại bỏ thuộc tính thừa** trong 3 phụ thuộc hàm $ABCD \rightarrow E$, $ACDF \rightarrow E$ và $ACDF \rightarrow G$

Xét $ABCD \rightarrow E$: Giả sử giảm ước A , ta còn $BCD \rightarrow E$, kiểm tra $BCD \rightarrow E$ có được suy ra từ F_1 không, ta tính $(BCD)^+$ (đối với F_1). $(BCD)^+ = BCD$, không chứa E , vậy thì $BCD \rightarrow E$ không được suy diễn ra từ F , vậy A không phải là thuộc tính thừa trong pth đang xét. B là thừa vì từ F_1 ta có $A \rightarrow B$ dẫn đến $(ACD)^+ = ABCDE$ có chứa E

Làm tương tự ta thấy không có thuộc tính nào là thừa nữa. $F_2 = \{A \rightarrow B, ACD \rightarrow E, EF \rightarrow G, ACDF \rightarrow E, ACDF \rightarrow G\}$

Ví dụ 2 (tiếp)

- **Loại bỏ pth thừa trong F_2 :** Lần lượt thử loại bỏ 1 pth ra khỏi F_2 , nếu tập pth thu được sau khi loại bỏ vẫn tương đương với F_2 thì pth vừa loại là thừa

$A \rightarrow B$ không thừa vì nếu loại pth này khỏi F_2 thì từ tập phụ thuộc hàm còn lại $A+$ không chứa B

Tương tự, $ACD \rightarrow E$, $EF \rightarrow G$ không thừa

$ACDF \rightarrow E$ là phụ thuộc hàm thừa vì nếu loại bỏ pth này, trong tập pth vẫn còn lại $ACD \rightarrow E$, theo tiên đề tăng trưởng ta sẽ suy ra được $ACDF \rightarrow E$

$ACDF \rightarrow G$ là thừa vì nếu loại bỏ pth này, trong tập pth còn lại vẫn có $ACD \rightarrow E$ và $EF \rightarrow G$, do đó ta vẫn có $(ACDF)^+ = ACDEFG$ có chứa G

- Vậy $F_c = \{ A \rightarrow B, ACD \rightarrow E, EF \rightarrow G \}$

Phép tách các sơ đồ quan hệ

- Mục đích
 - Thay thế một sơ đồ quan hệ $R(A_1, A_2, \dots, A_n)$ bằng một tập các sơ đồ con $\{R_1, R_2, \dots, R_k\}$ trong đó $R_i \subseteq R$ và $R = R_1 \cup R_2 \cup \dots \cup R_k$
- Yêu cầu của phép tách
 - Bảo toàn thông tin
 - Bảo toàn tập phụ thuộc

Phép tách không mất mát thông tin

- **Định nghĩa 8:** Cho sơ đồ quan hệ R phép tách R thành các sơ đồ con $\{R_1, R_2, \dots, R_k\}$ được gọi là **phép tách không mất mát thông tin** đ/v một tập phụ thuộc hàm F nếu với mọi quan hệ r xác định trên R thỏa mãn F thì:

$$r = \Pi_{R_1}(r) \bowtie \Pi_{R_2}(r) \bowtie \dots \bowtie \Pi_{R_k}(r)$$

- Ví dụ: Phép tách **mất mát** thông tin

Supplier(sid, sname, city, status, pid, pname, colour, quantity)

→ S1(sid, sname, city, status)

và

SP1(pid, pname, colour, quantity)

- Ví dụ: Phép tách **không mất mát** thông tin

→ S1(sid, sname, city, NOE)

SP2(sid, pid, pname, colour, quantity)

và
39

Phép tách không mất mát thông tin

- **Định lý 1:** Cho sơ đồ quan hệ $R(U)$, tập pth F , phép tách R thành $R_1(U_1)$, $R_2(U_2)$ là một phép tách không mất mát thông tin nếu 1 trong 2 phụ thuộc hàm sau là thỏa mãn trên F^+ :

$$U_1 \cap U_2 \rightarrow U_1 - U_2$$

$$U_1 \cap U_2 \rightarrow U_2 - U_1$$

- **Hệ quả 1:** Cho sơ đồ quan hệ $R(U)$ và phụ thuộc hàm $X \rightarrow Y$ thỏa mãn trên $R(U)$. Phép tách R thành 2 sơ đồ con $R_1(U_1)$, $R_2(U_2)$ là một phép tách không mất mát thông tin với:

$$U_1 = XY$$

$$U_2 = XZ$$

$$Z = U \setminus XY$$

Phép tách không mất mát thông tin

- **Thuật toán 5:** Kiểm tra tính không mất mát thông tin của 1 phép tách

- **Vào:** $R(A_1, A_2, \dots, A_n)$, F , phép tách $\{R_1, R_2, \dots, R_k\}$
- **Ra:** phép tách là mất mát thông tin hay không
- **Phương pháp**

B.1. Thiết lập một bảng k hàng, n cột

Nếu A_j là thuộc tính của R_i thì điền a_j vào ô (i,j) .

Nếu không thì điền b_{ij} .

B.i. Xét $f = X \rightarrow Y \in F$

Nếu \exists 2 hàng $t1, t2$ thuộc bảng : $t1[X] = t2[X]$ thì đồng nhất

$t1[Y] = t2[Y]$, ưu tiên về giá trị a

Lặp cho tới khi không thể thay đổi được giá trị nào trong bảng

B.n. Nếu bảng có 1 hàng gồm các kí hiệu a_1, a_2, \dots, a_n thì phép tách là không mất mát thông tin ngược lại, phép tách là mất mát thông tin

Ví dụ

- $R = ABCD$ được tách thành $R1=AB$, $R2=BD$, $R3=ABC$, $R4=BCD$. $F = \{A \rightarrow C, B \rightarrow C, CD \rightarrow B, C \rightarrow D\}$
- B.1: Tạo bảng gồm 4 hàng, 4 cột

	A	B	C	D
R1	a1	a2	b31	b41
R2	b12	a2	b32	a4
R3	a1	a2	a3	b43
R4	b14	a2	a3	a4

Ví dụ (tiếp)

- B.2 & 3:
- Từ $A \rightarrow C$, ta có
- Từ $B \rightarrow C$, ta có

	A	B	C	D
R1	a1	a2	a3	b41
R2	b12	a2	b32	a4
R3	a1	a2	a3	b43
R4	b14	a2	a3	a4

	A	B	C	D
R1	a1	a2	a3	b41
R2	b12	a2	a3	a4
R3	a1	a2	a3	b43
R4	b14	a2	a3	a4

Ví dụ (tiếp)

- Từ $C \rightarrow D$, ta có

	A	B	C	D
R1	a1	a2	a3	a4
R2	b12	a2	a3	a4
R3	a1	a2	a3	a4
R4	b14	a2	a3	a4

- Vậy ta có 2 hàng có toàn các giá trị a. Chứng tỏ phép tách đã cho là không mất mát thông tin

Phép tách bảo toàn tập phụ thuộc hàm

- Hình chiếu của tập phụ thuộc hàm

Cho sơ đồ quan hệ R , tập phụ thuộc hàm F , phép tách $\{R_1, R_2, \dots, R_k\}$ của R trên F .

Hình chiếu F_i của F trên R_i là tập tất cả $X \rightarrow Y \in F+$:

$$XY \subseteq R_i$$

- Phép tách sơ đồ quan hệ R thành $\{R_1, R_2, \dots, R_k\}$ là một phép tách bảo toàn tập phụ thuộc hàm F nếu

$$(F_1 \cup F_2 \dots \cup F_k)+ = F+$$

hay hợp của tất cả các phụ thuộc hàm trong các hình chiếu của F lên các sơ đồ con sẽ suy diễn ra các phụ thuộc hàm trong F .

Ví dụ

- **Ví dụ 1:** $R = \{A, B, C\}$ $F = \{A \rightarrow B, B \rightarrow C, C \rightarrow A\}$ được tách thành $R_1 = AB, R_2 = BC$. Phép tách này có phải là bảo toàn tập phụ thuộc hàm không?
- **Ví dụ 2:** $R = \{A, B, C\}$, $F = \{AB \rightarrow C, C \rightarrow B\}$ được tách thành $R_1 = AB, R_2 = BC$. Phép tách này có bảo toàn tập pth không, có mất mát thông tin không?
- **Ví dụ 3:** $R = \{A, B, C, D\}$, $F = \{A \rightarrow B, C \rightarrow D\}$ được tách thành $R_1 = AB, R_2 = CD$. Phép tách này có bảo toàn tập pth không, có mất mát thông tin không?
- Vậy một phép tách có bảo toàn tập phụ thuộc hàm thì không đảm bảo là nó sẽ không mất mát thông tin và ngược lại

Các dạng chuẩn đối với SĐQH

- Quay lại vấn đề thiết kế cơ sở dữ liệu quan hệ, câu hỏi mà chúng ta đặt ra trong quá trình này là Có cần thiết phải tinh chỉnh thiết kế nữa hay không, thực sự thiết kế mà chúng ta có được đã là tốt hay chưa. Để giúp trả lời câu hỏi này, người ta đưa ra khái niệm về các dạng chuẩn. Khi một quan hệ thuộc vào một dạng chuẩn, một dạng dư thừa dữ liệu hay dị thường dữ liệu tương ứng với dạng chuẩn đã được khắc phục.
- Các dạng chuẩn đối với SĐQH sử dụng phụ thuộc hàm gồm có:
 - Dạng chuẩn 1 (1NF)
 - Dạng chuẩn 2 (2NF)
 - Dạng chuẩn 3 (3NF)
 - Dạng chuẩn Boye-Code (BCNF)

Dạng chuẩn 1 (1NF)

- **Định nghĩa 9:** Một sơ đồ quan hệ R được gọi là ở dạng chuẩn 1 nếu tất cả các miền giá trị của các thuộc tính trong R đều chỉ chứa giá trị nguyên tố
 - Giá trị nguyên tố là giá trị mà không thể chia nhỏ ra được nữa
- Một quan hệ r xác định trên sơ đồ quan hệ ở dạng chuẩn 1 thì quan hệ đấy là ở dạng chuẩn 1
- Ví dụ: Quan hệ không ở dạng chuẩn 1 và quan hệ sau khi chuẩn hóa về dạng chuẩn 1

sname	city	product
Blake	London	(Nut, 100)
		(Bolt, 120)
Smith	Paris	(Screw, 75)

sname	city	item	price
Blake	London	Nut	100
Blake	London	Bolt	120
Smith	Paris	Screw	75

Dạng chuẩn 2 (2NF)

- *Định nghĩa 10:* Cho sơ đồ quan hệ $R(U)$, F là tập phụ thuộc hàm trên R . $X, Y \subseteq U$. Y được gọi là **phụ thuộc đầy đủ** vào X nếu:
 - $X \rightarrow Y$ thuộc F^+
 - $\nexists X' \subset X : X' \rightarrow Y \in F^+$
- Các phụ thuộc hàm không đầy đủ còn gọi là **phụ thuộc bộ phận**



Dạng chuẩn 2 (2NF)

- *Định nghĩa 11*: Một sơ đồ quan hệ R được coi là ở dạng chuẩn 2 nếu
 - Sơ đồ quan hệ này ở 1NF
 - Tất cả các thuộc tính không khoá đều phụ thuộc hàm đầy đủ vào khoá chính
(Lưu ý, A là một thuộc tính khoá nếu A thuộc một khoá tối thiểu nào đó của R. Ngược lại A là thuộc tính không khoá)

Ví dụ

- Sales(sid, sname, city, item, price)
- $F = \{sid \rightarrow (sname, city), (sid, item) \rightarrow price\}$
- Khoá chính (sid, item), ta có sname, city không phụ thuộc hàm đầy đủ vào khoá chính \Rightarrow Quan hệ Sales không thuộc 2NF
- S(sid, sname, city) và Sales (sid, item, price) là quan hệ thuộc 2NF

Dạng chuẩn 3 (3NF)

- *Định nghĩa 12:* Cho sơ đồ quan hệ $R(U)$. F là tập phụ thuộc hàm trên $R(U)$. $X \subseteq U$, $A \in U$. Ta nói Z là phụ thuộc bắc cầu vào X nếu $\exists Y$, $Y \subseteq U$ sao cho $X \rightarrow Y$, $Y \rightarrow Z$ thuộc F^+ . Ngược lại, ta nói Z không phụ thuộc bắc cầu vào X



Dạng chuẩn 3 (3NF)

- *Định nghĩa 13*: Một sơ đồ quan hệ R được coi là ở dạng chuẩn 3 nếu
 - Sơ đồ quan hệ này ở 2NF
 - Mọi thuộc tính không khoá đều không phụ thuộc bắc cầu vào khoá chính

Ví dụ

- Ví dụ 1: Trong ví dụ tách về dạng chuẩn 2 ta có: $S(sid, sname, city)$ và $Sales(sid, item, price)$.

Xét quan hệ S , pth $sid \rightarrow sname, city$ tồn tại trên S , sid là khoá chính, các thuộc tính không khoá $sname, city$ đều phụ thuộc trực tiếp vào sid . S thuộc 3NF. Tương tự ta có $Sales$ cũng thuộc 3NF

- Ví dụ 2:
 - $ItemInfo(item, price, discount)$. $F = \{item \rightarrow price, price \rightarrow discount\}$. Khoá chính là $item$, thuộc tính không khoá $discount$ phụ thuộc bắc cầu vào khoá chính $item$. Vậy quan hệ này không ở 3NF.
 - $ItemInfo(item, price)$ và $Discount(price, discount)$ thuộc 3NF.

Dạng chuẩn Boye-Codd (BCNF)

- **Định nghĩa 14:** Một sơ đồ quan hệ $R(U)$ với một tập phụ thuộc hàm F được gọi là ở dạng chuẩn Boye-Codd (BCNF) nếu với $\forall X \rightarrow A \in F^+$ thì
 - A là thuộc tính xuất hiện trong X hoặc
 - X chứa một khoá của quan hệ R .
- Ví dụ
 - $R = \{\underline{A}, \underline{B}, C\}$; $F = \{AB \rightarrow C, C \rightarrow B\}$.
 - R không phải ở BCNF vì $\exists C \rightarrow B$, C không phải là khoá
- Chú ý:
 - Một quan hệ thuộc 3NF thì chưa chắc đã thuộc BCNF. Nhưng một quan hệ thuộc BCNF thì thuộc 3NF

Tách bảo toàn tập phụ thuộc hàm về 3NF

- **Thuật toán 6:**

- **Vào:** $R(U)$, F (giả thiết F là phủ tối thiểu)
- **Ra:** Phép tách bảo toàn tập phụ thuộc hàm về 3NF
- **Phương pháp:**

B1. Với các $A_i \in U$, $A_i \notin F$ thì loại A_i khỏi R và lập 1 quan hệ mới cho các A_i

B2. Nếu $\exists f \in F$, f chứa tất cả các thuộc tính của R (đã bỏ các A_i ở bước trên) thì kết quả là R

B3. Ngược lại, với mỗi $X \rightarrow A \in F$, xác định một sơ đồ quan hệ $R_i(XA)$.

Nếu $\exists X \rightarrow A_i, X \rightarrow A_j$ thì tạo một sơ đồ quan hệ chung $R'(XA_iA_j)$

Ví dụ

Cho $R = \{A, B, C, D, E, F, G\}$

$F = \{A \rightarrow B, ACD \rightarrow E, EF \rightarrow G\}$ (đã tối thiểu)

- Xác định phép tách bảo toàn tập phụ thuộc hàm về 3NF

B1. Không lập được quan hệ nào mới.

B2. $\nexists f \in F$: f chứa tất cả các thuộc tính của R

B3. $A \rightarrow B \quad \Rightarrow R1(AB)$

$ACD \rightarrow E \quad \Rightarrow R2(ACDE)$

$EF \rightarrow G \quad \Rightarrow R3(EFG)$

Tách không mất mát thông tin và bảo toàn tập phụ thuộc hàm về 3NF

- Yêu cầu:

- Bảo toàn tập phụ thuộc hàm (như thuật toán trên)
- Đảm bảo là có một sơ đồ con chứa khoá của sơ đồ được tách

- Các bước tiến hành

B1. Tìm một khoá tối thiểu của sơ đồ quan hệ R đã cho

B2. Tách sơ đồ quan hệ R theo phép tách bảo toàn tập phụ thuộc hàm.

B3. Nếu 1 trong các sơ đồ con có chứa khoá tối thiểu thì kết quả của B2 là kết quả cuối cùng

Ngược lại, thêm vào kết quả đó một sơ đồ quan hệ được tạo bởi khoá tối thiểu tìm được ở B1

Ví dụ

- Cho $R(U)$ trong đó $U = \{A, B, C, D, E, F, G\}$. $F = \{A \rightarrow B, ACD \rightarrow E, EF \rightarrow G\}$
- Tìm một khoá tối thiểu của R :
 $K^0 = ABCDEFG$
 $K^1 = K^0$ do nếu loại A thì $BCDEFG \rightarrow U$ không thuộc F^+
 $K^2 = K^1 \setminus \{B\} = ACDEFG$ do $ACDEFG \rightarrow U$ thuộc F^+
 $K^3 = K^2$ do nếu loại C thì $ADEFG \rightarrow U$ không thuộc F^+
 $K^4 = K^3$ do nếu loại D thì $ACEFG \rightarrow U$ không thuộc F^+
 $K^5 = K^4 \setminus \{E\} = ACDFG$ do $ACDFG \rightarrow U$ thuộc F^+
 $K^6 = K^5$ do nếu loại F thì $ACDG \rightarrow U$ không thuộc F^+
 $K^7 = K^6 \setminus \{G\} = ACDF$ do $ACDF \rightarrow U$ thuộc F^+
- Vậy khoá tối thiểu cần tìm là $ACDF$

Ví dụ (tiếp)

- Dùng kết quả của ví dụ ở phần tách bảo toàn tập phụ thuộc hàm ta có một phép tách R thành 3 sơ đồ con $R_1 = AB$, $R_2 = ACDE$, $R_3 = EFG$
- Do khoá $ACDF$ không nằm trong bất kỳ một sơ đồ con nào trong 3 sơ đồ con trên, ta lập một sơ đồ con mới $R_4 = ACDF$
- Kết quả cuối cùng ta có phép tách R thành 4 sơ đồ con $\{R_1, R_2, R_3, R_4\}$ là một phép tách không mất mát thông tin và bảo toàn tập phụ thuộc hàm

Tách không mất mát thông tin về BCNF

- **Thuật toán 7:**
- **Vào:** Sơ đồ quan hệ R , tập phụ thuộc hàm F .
- **Ra:** phép tách không mất mát thông tin bao gồm một tập các sơ đồ con ở BCNF với các phụ thuộc hàm là hình chiếu của F lên sơ đồ đó.
- **Phương pháp:**
 - B1.** $KQ = \{R\}$,
 - B2.** Với mỗi $S \in KQ$, S không ở BCNF, xét $X \rightarrow A \in S$, với điều kiện X không chứa khoá của S và $A \notin X$. Thay thế S bởi S_1, S_2 với $S_1 = A \cup \{X\}$, $S_2 = \{S\} \setminus A$.
 - B3.** Lặp (B2) cho đến khi $\forall S \in KQ$ đều ở BCNF
KQ gồm các sơ đồ con của phép tách yêu cầu

Tóm tắt về thiết kế CSDL sử dụng các phụ thuộc hàm

- **Mục đích của thiết kế CSDL:**

- Tránh dư thừa dữ liệu
- Tránh dị thường dữ liệu khi thêm/xoá/sửa đổi
- Hiệu quả trong tìm kiếm

- **Đưa về các dạng chuẩn sử dụng các phụ thuộc hàm**

- 2NF: giảm ước sự dư thừa gây nên bởi các phụ thuộc hàm bộ phận và tránh các dị thường khi cập nhật
- 3NF: giảm ước sự dư thừa gây nên bởi các phụ thuộc hàm bắc cầu và tránh các dị thường khi cập nhật

Phụ thuộc đa trị

- **Định nghĩa 15:** Cho $R(U)$, $X, Y \in U$. X xác định đa trị Y hay Y phụ thuộc đa trị vào X ($X \twoheadrightarrow Y$) nếu với $\forall r$ xác định trên R và với hai bộ $t1$ và $t2$ bất kỳ mà $t1[X] = t2[X]$ thì \exists bộ $t3$:
 $t3[X] = t1[X]$, $t3[Y] = t1[Y]$ và $t3[Z] = t2[Z]$ với $Z = U \setminus XY$.

Ví dụ

- DAY_HOC

MônHọc	GiảngViên	Đối Tượng
Cơ sở dữ liệu	Lê Thanh Hà	Đại học
Cơ sở dữ liệu	Lê Thanh Hương	Đại học
Cơ sở dữ liệu	Lê Thanh Hà	Cao đẳng
Cơ sở dữ liệu	Lê Thanh Hương	Cao đẳng
Xử lý ảnh	Vũ Quốc Huy	Đại học

MônHọc →→ GiảngViên, MônHọc →→ Đối Tượng

Hệ tiên đề đối với các phụ thuộc hàm và phụ thuộc đa trị

Cho $R(U)$, $X, Y, Z, W \subseteq U$ ($XY = X \cup Y$)

- A1: Phản xạ đối với FD (*reflexivity*):
Nếu $Y \subseteq X$ thì $X \rightarrow Y$.
- A2: Tăng trưởng đối với FD (*augmentation*):
Nếu $X \rightarrow Y$ thì $XZ \rightarrow YZ$.
- A3: Bắt cầu đối với FD (*transitivity*):
Nếu $X \rightarrow Y$, $Y \rightarrow Z$ thì $X \rightarrow Z$.
- A4: Luật bù đối với MVD (*complementation*):
Nếu $X \twoheadrightarrow Y$ thì $X \twoheadrightarrow U \setminus XY$.

Hệ tiên đề đối với các phụ thuộc hàm và phụ thuộc đa trị (2)

Cho $R(U)$, $X, Y, Z, W \subseteq U$ ($XY = X \cup Y$)

- A5: Tăng trưởng đối với MVD (*augmentation*):

Nếu $X \twoheadrightarrow Y$ và $V \subseteq W$ thì $WX \twoheadrightarrow VY$.

- A6: Bắc cầu đối với MVD (*transitivity*):

Nếu $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$ thì $X \twoheadrightarrow Z \setminus Y$.

- A7:

Nếu $X \rightarrow Y$ thì $X \twoheadrightarrow Y$.

- A8:

Nếu $X \twoheadrightarrow Y$, $W \rightarrow Z$ với $Z \subseteq Y$ và $W \cap Y = \emptyset$ thì $X \rightarrow Z$.

Các luật suy diễn bổ sung đối với các phụ thuộc đa trị

- Luật hợp (*union*):

Nếu $X \twoheadrightarrow Y$, $X \twoheadrightarrow Z$ thì $X \twoheadrightarrow YZ$.

- Luật tựa bắc cầu (*pseudotransitivity*):

Nếu $X \twoheadrightarrow Y$, $WY \twoheadrightarrow Z$ thì $WX \twoheadrightarrow Z \setminus WY$.

- Luật tựa bắc cầu hỗn hợp (*mixed pseudotransitivity*):

Nếu $X \twoheadrightarrow Y$, $XY \rightarrow Z$ thì $X \rightarrow Z \setminus Y$.

- Luật tách (*decomposition*):

Nếu $X \twoheadrightarrow Y$, $X \twoheadrightarrow Z$ thì

$X \twoheadrightarrow Y \cap Z$, $X \twoheadrightarrow Y \setminus Z$, $X \twoheadrightarrow Z \setminus Y$.

Bao đóng của tập phụ thuộc hàm và phụ thuộc đa trị

- **Định nghĩa 16:** Bao đóng của tập các phụ thuộc hàm và phụ thuộc đa trị D là tập tất cả các phụ thuộc hàm và các phụ thuộc đa trị được suy diễn logic từ D
 - Ký hiệu: D^+

Tính cơ sở phụ thuộc

- **Vào:** Tập các phụ thuộc đa trị M trên tập thuộc tính U và tập thuộc tính $X \subseteq U$.
- **Ra:** Cơ sở phụ thuộc của X đối với M .
- Cách tiến hành:
 - B1.** Đặt T là tập các tập con Z của U : với $W \twoheadrightarrow Y \in M$ mà $W \subseteq X$ thì Z là $Y \setminus X$ hoặc $U \setminus XY$.
 - B2.** T được thiết lập cho tới khi là một tập các tập rời nhau, nếu có một cặp Z_1, Z_2 không tách rời nhau thì thay chúng bởi $Z_1 \setminus Z_2, Z_2 \setminus Z_1, Z_1 \cap Z_2$ với điều kiện không ghi nhận tập rỗng. Gọi S là tập thu được sau bước này.
 - B3.** Tìm các phụ thuộc có dạng $V \twoheadrightarrow W$ trong M và một tập Y trong S : $Y \cap W \neq \emptyset, Y \cap V = \emptyset$
Thay Y bằng $Y \cap W$ và $Y \setminus W$ cho đến khi không thay đổi S được nữa.
 - B4.** Tập S thu được sau bước này là cơ sở phụ thuộc của X .

Phép tách không mất thông tin

- **Vào:** $R(A_1, A_2, \dots, A_n)$, F , M , phép tách $\{R_1, R_2, \dots, R_k\}$
 - **Ra:** phép tách là mất mát thông tin hay không
 - **Phương pháp** (tổng quát hoá thuật toán trình bày với pth)
- B.1.** Thiết lập một bảng k hàng, n cột (xem B1. slide 33)

B.i. Xét $f = X \rightarrow Y \in F$:

thực hiện đồng nhất bảng (xem B2. slide 33)

Xét $X \twoheadrightarrow Y$:

*nếu $\exists 2$ hàng t_1, t_2 thuộc bảng : $t_1[X] = t_2[X]$
thì thêm vào bảng đó một hàng mới u*

$u[X] = t_1[X], u[Y] = t_1[Y],$

$u[R \setminus XY] = t_2[R \setminus XY]$

Lặp cho tới khi không thể thay đổi được giá trị nào trong bảng

- B.n.** Nếu bảng có 1 hàng gồm các kí hiệu a_1, a_2, \dots, a_n thì phép tách là không mất mát thông tin.
ngược lại, phép tách không bảo toàn thông tin.

Dạng chuẩn 4 (4NF)

- **Định nghĩa 17:** Một quan hệ R ở dạng chuẩn bốn nếu với mỗi phụ thuộc đa trị $X \twoheadrightarrow Y$ với $Y \neq \emptyset$, $Y \not\subset X$ và $XY \subset R$ thì X chứa một khóa của R
- Chú ý: nếu R chỉ có các phụ thuộc hàm thì dạng chuẩn bốn chính là dạng chuẩn Boye-Codd và $X \twoheadrightarrow Y$ phải có nghĩa là $X \rightarrow Y$.

Kết luận

- Tầm quan trọng của thiết kế CSDL
 - ảnh hưởng đến chất lượng dữ liệu lưu trữ
 - Hiệu quả của việc khai thác dữ liệu
- Mục đích của thiết kế CSDL:
 - Tránh dư thừa dữ liệu
 - Tránh dị thường dữ liệu khi thêm/xoá/sửa đổi
 - Hiệu quả trong tìm kiếm
- **Đưa về các dạng chuẩn sử dụng các phụ thuộc hàm**
 - 2NF: giảm ước sự dư thừa gây nên bởi các phụ thuộc hàm bộ phận và tránh các dị thường khi cập nhật
 - 3NF: giảm ước sự dư thừa gây nên bởi các phụ thuộc hàm bắc cầu và tránh các dị thường khi cập nhật
 - 3NF: giảm ước sự dư thừa gây nên bởi các phụ thuộc đa trị và tránh các dị thường khi cập nhật

