

Kỹ thuật phần mềm ứng dụng

Chương 3

Mô hình dữ liệu quan hệ (Phần 3)

Mục đích

Nắm được các vấn đề và khái niệm

- *Các bất thường trong một quan hệ (Anomalies)*
- *Phụ thuộc hàm (Functional Dependencies)*
- *Khóa*
- *Sự tách lược đồ quan hệ (Decomposition)*

Các nội dung chính

1. *Giới thiệu về các bất thường trong một quan hệ*
2. *Các phụ thuộc hàm*
3. *Khái niệm các khóa*
4. *Các nguyên nhân gây ra các bất thường*
5. *Phép tách lược đồ QH*

1. Các bất thường trong quan hệ

- *Hãy quan sát bảng Student*

	ID	Name	Class	Department	Subject	Mark
1	E1-001	Nguyen Van A	E1	Electronics	Electronic Circuit	8
2	E1-001	Nguyen Van A	E1	Electronics	Digital Technique	7
3	E1-002	Tran Thi B	E1	Electronics	Digital Technique	9
4	E1-002	Tran Thi B	E1	Electronics	Electronic Circuit	8
5	E2-001	Nguyen Ho C	E2	Electronics	Digital Technique	6
6	IT1-001	Tran Thi B	IT1	IT	Electronic Circuit	10
7	IT1-002	Le Van D	IT1	IT	Digital Technique	8

1. Các bất thường trong quan hệ

Từ bảng trên có thể nhận thấy một số bất thường:

- ***Dư thừa (Redundancy)***: giá trị của một số thành phần trong các bộ bị lặp lại không cần thiết như : Name, Class, Department
- ***Bất thường khi cập nhật (Update Anomalie)***: xuất hiện khi cập nhật giá trị cho một bộ hiện có, vì thực tế thực hiện của thao tác có vẻ phức tạp hơn rất nhiều so với logic của thao tác đó.
- ***Bất thường khi bổ sung (Insertion Anomalie)***: xuất hiện khi bổ sung thêm một bộ mới.
- ***Bất thường khi xóa (Deletion Anomalie)***: xuất hiện khi xóa một bộ hiện có trong quan hệ.

1. Các bất thường trong quan hệ

Hậu quả của các bất thường trên

- *Sự không nhất quán dữ liệu có nguy cơ rất cao*
- *Tốn chỗ lưu trữ do dư thừa*
- *Các thao tác cơ bản trên CSDL không hiệu quả*

1. Các bất thường trong quan hệ

- *Vậy nguyên nhân của các bất thường này là gì?*
- ➔ *Khái niệm **Phụ thuộc hàm** có thể giúp chúng ta hiểu nguyên nhân, cũng như giúp tìm ra giải pháp cho vấn đề trên.*

2. Phụ thuộc hàm

- *Định nghĩa*
- *Ý nghĩa của PTH*
- *Hệ tiên đề Amstrong*
- *Một số loại PTH đặc biệt*
- *Khóa và thuộc tính khóa*

2. Phụ thuộc hàm

- *Định nghĩa về PTH (Functional Dependency)*
 - Cho lược đồ quan hệ $R(A_1, A_2, \dots, A_n)$, và $A = A_1 \cup A_2 \cup \dots \cup A_n$, và 2 tập các thuộc tính X và $Y \subseteq A$. Chúng ta nói rằng:
 - $X \rightarrow Y$ (X xác định hàm Y , hay Y phụ thuộc hàm vào X), nếu \forall thể hiện r của R , thì không tồn tại 2 bộ t_1, t_2 , sao cho $t_1[X] = t_2[X]$ và $t_1[Y] \neq t_2[Y]$
 - Nói cách khác: tập các thuộc tính Y phụ thuộc hàm vào tập các thuộc tính X khi và chỉ khi ứng với mỗi bộ giá trị của X trong R chỉ xác định đúng một giá trị của Y tại mọi thời điểm
 - X được gọi là **Quyết định** (determinant) của PTH.

2. Phụ thuộc hàm — ví dụ

- Một số PTH từ bảng *Student*:
 - $ID \rightarrow Name$;
 - $ID \rightarrow Class$;
 - $Class \rightarrow Department$;
 - $ID, Subject \rightarrow Mark$

2. Phụ thuộc hàm — Ý nghĩa

- Ý nghĩa của một PTH $X \rightarrow Y$ trong lược đồ R : \forall thể hiện r của R thì:
 - Hoặc không có bất kỳ 2 bộ t_1, t_2 nào, sao cho $t_1[X] = t_2[X]$
 - Hoặc nếu tồn tại 2 bộ t_1, t_2 mà $t_1[X] = t_2[X]$, thì $t_1[Y] = t_2[Y]$
- Trong cả hai trường hợp, \forall bộ t , với mỗi giá trị của $t[X]$ thì chỉ có một giá trị của $t[Y]$, do đó nếu biết trước giá trị $t[X]$, thì có thể xác định giá trị $t[Y]$.

2. Phụ thuộc hàm — Ý nghĩa

- Ý nghĩa của PTH $X \rightarrow Y$:
 - Thế thì PTH đến từ đâu?
 - Nó xuất phát các **quy tắc nghiệp vụ** (hay **logic nghiệp vụ**) của cơ sở dữ liệu. Các quy tắc nghiệp vụ này sẽ quy định các phụ thuộc giữa các thuộc tính, và từ đó sẽ tạo ra các phụ thuộc hàm.

2. Phụ thuộc hàm — Ý nghĩa

- Ví dụ: một số quy tắc nghiệp vụ trong lược đồ *Student*:
 - **r1**: Mỗi sinh viên có một ID duy nhất.
 - **r2**: Mỗi sinh viên chỉ có duy nhất 1 tên và chỉ thuộc về một lớp.
 - Từ *r1* và *r2*, ta có các PTH sau:
 $ID \rightarrow Name; ID \rightarrow Class$
 - **r3**: Với mỗi môn học, mỗi sinh viên chỉ thi một lần
 - Từ *r3*, ta có PTH: $(ID, Subject) \rightarrow Mark;$

2. Phụ thuộc hàm

- *Hệ tiên đề Amstrong*
 - *Tính phản xạ: $\forall Y \subseteq X$ thì $X \rightarrow Y$ (PTH tầm thường)*
 - *Tính tăng trưởng: $\forall X, Y, Z$, nếu $X \rightarrow Y$ thì $XZ \rightarrow YZ$ ($XZ = X \cup Z$)*
 - *Tính bắc cầu: $\forall X, Y, Z$, nếu $X \rightarrow Y$ và $Y \rightarrow Z$ thì $X \rightarrow Z$*
- *Các hệ quả:*
 - *Tính hợp : nếu $X \rightarrow Y$ và $X \rightarrow Z$ thì $X \rightarrow YZ$*
 - *Tính tách: nếu $X \rightarrow Y$ và $Z \subseteq Y$ thì $X \rightarrow Z$*
 - *Tính tựa bắc cầu: nếu $X \rightarrow Y$ và $YW \rightarrow Z$ thì $XW \rightarrow Z$*

2. Phụ thuộc hàm

- Một số loại PTH đặc biệt
 - **PTH bộ phận và PTH đầy đủ:** $X \rightarrow Y$ được gọi là **PTH bộ phận** nếu $\exists X' \subset X$ sao cho $X' \rightarrow Y$. Trái lại, nếu không tồn tại X' như trên thì **PTH này là đầy đủ**.
 - **PTH bắc cầu và PTH trực tiếp:** $X \rightarrow Y$ được gọi là **PTH bắc cầu** (hay gián tiếp) nếu $\exists Z \not\subset XY$ sao cho $X \rightarrow Z$ và $Z \rightarrow Y$. Trái lại, nếu không tồn tại Z như trên thì **PTH này là trực tiếp**.

3. Khóa

Có một số loại khóa như sau:

- Siêu khóa (Super Keys).*
- Khóa ứng viên (Candidate Kyes).*
- Khóa chính (Primary Kyes).*
- Khóa nói chung (Keys).*
- Khóa ngoại.*

3. Khóa

Siêu khóa

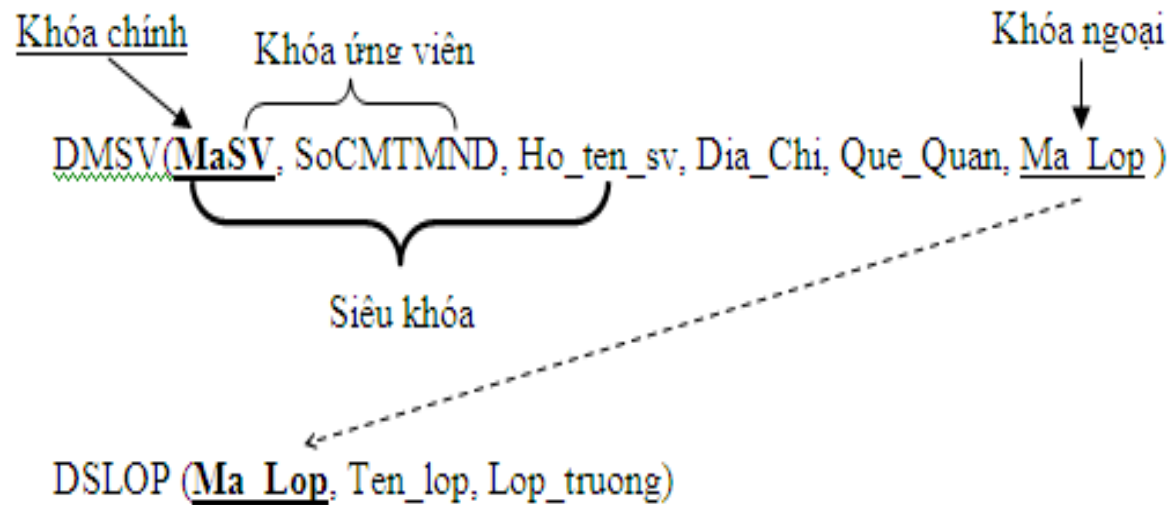
- *Là một hay một tập các cột xác định duy nhất một hàng trong bảng*
- *SK là siêu khóa của R khi với mọi $t1, t2$ thuộc bất kỳ thể hiện r của R ta luôn có $t1[SK] \neq t2[SK]$*
- *SK là siêu khóa của R \Leftrightarrow SK xác định hàm mọi thuộc tính của R ($SK \rightarrow R$).*

*Ví dụ: Ta có lược đồ **DMSV(MaSV, SoCMTND, Ho_ten_sv, Ngay_Sinh, Dia_chi, Que_quan)**. Ta có thể có siêu khóa sau (**DMDV(MaSV, SoCMTND, Ho_ten_sv)**), khi đó một khóa nói chung là thành phần tối thiểu của siêu khóa.*

3 Khóa (tiếp)

- *Khóa ứng viên (khóa ứng cử): Là một tập con của siêu khóa (cũng là một siêu khóa), nhưng không thể thu lại thành một siêu khóa khác được gọi là khóa ứng viên hay khóa dự tuyển, Ví dụ ta có tập các khóa ứng viên $\{(MaSV), (SoCMTND)\}$.*
- *Khóa chính: Là một khóa được lựa chọn một cách tùy ý từ tập các khóa ứng viên, và được sử dụng trong các chỉ mục của bảng đó.*
- ***Khóa ngoại: Một thuộc tính của quan hệ $R1$ được gọi là một khoá ngoại lai nếu nó không phải là khoá chính của quan hệ $R1$ nhưng các giá trị của nó là các giá trị của khoá chính trong một quan hệ $R2$ nào đó (quan hệ $R1$ và $R2$ không nhất thiết phải khác nhau).***
Khóa ngoại lai dùng để chứa dữ liệu dùng cho việc liên kết với các bảng khác trong CSDL.

Ví dụ



3. Khóa (tiếp)

- ***Thuộc tính khóa*** (*Prime attribute*): 1 thuộc tính A là ***thuộc tính khóa*** nếu nó tham gia vào thành phần của khóa chính hoặc khóa ứng cử. Trái lại, thì nó là ***thuộc tính không khóa*** (hay ***thuộc tính mô tả***)

4. Nguyên nhân gây ra các bất thường

	<u>ID</u>	Name	Class	Department	<u>Subject</u>	Mark
1	E1-001	Nguyen Van A	E1	Electronics	Electronic Circuit	8
2	E1-001	Nguyen Van A	E1	Electronics	Digital Technique	7
3	E1-002	Tran Thi B	E1	Electronics	Digital Technique	9
4	E1-002	Tran Thi B	E1	Electronics	Electronic Circuit	8
5	E2-001	Nguyen Ho C	E2	Electronics	Digital Technique	6
6	IT1-001	Tran Thi B	IT1	IT	Electronic Circuit	10
7	IT1-002	Le Van D	IT1	IT	Digital Technique	8

4. Nguyên nhân gây ra các bất thường

- *Khóa chính của quan hệ Student:*
 $K = (ID, Subject)$
- **Nhận xét:** các thuộc tính dư thừa là các thuộc tính phụ thuộc hàm bộ phận hoặc bắc cầu vào Khóa chính, ví dụ như:
 - $(ID, Subject) \rightarrow Name, Class, Department$; PTH bộ phận vì $ID \rightarrow Name, ID \rightarrow Class, ID \rightarrow Department$

4. Nguyên nhân gây ra các bất thường

- *Là sự tồn tại của các phụ thuộc bộ phận hoặc bắc cầu vào khóa chính*
- ***Giải pháp:*** Cần loại bỏ các loại PTH này trong quan hệ, nhưng làm thế nào?
→ ***Phép tách lược đồ!***

4. Phép tách

- **Định nghĩa:** cho một LĐQH R . Một phép tách R là sự thay thế nó bằng các LĐ con (sub-schema) R_1, R_2, \dots, R_n sao cho $R = R_1 \cup R_2 \cup \dots \cup R_n$. Ký hiệu phép tách là: $\rho(R) = (R_1, R_2, \dots, R_n)$ (ρ called rho)
- Một số tính chất muốn có của phép tách:
 - Tách nối không mất thông tin (Lossless-join decomposition)
 - Bảo toàn các PTH (Preservation of FDs)

4. Phép tách

- **Tách nối không mất thông tin:** giả sử cho $\rho = (R_1, R_2, \dots, R_n)$ là 1 phép tách R . ρ được gọi là tách nối không mất thông tin nếu nó thỏa mãn:

\forall thể hiện r của R thì biểu thức sau luôn thỏa mãn: :

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

- **Bảo toàn các PTH:** cho LĐ R với tập các PTH F . Phép tách $\rho = (R_1, R_2, \dots, R_n)$ được gọi là bảo toàn PTH nếu nó thỏa mãn:

$$F = \Pi_{R_1}(F) \cup \Pi_{R_2}(F) \cup \dots \cup \Pi_{R_n}(F)$$

$$\text{Với } \Pi_{R_i}(F) = \{X \rightarrow Y \mid X \rightarrow Y \text{ and } XY \subseteq R_i\}$$

Ví dụ: tách bảng *Student* thành 3 bảng S1, S2 và S3

S1

<u>ID</u>	Name	Class
E1-001	Nguyen Van A	E1
E1-002	Tran Thi B	E1
E2-001	Nguyen Ho C	E2
IT1-001	Tran Thi B	IT1
IT1-002	Le Van D	IT1

S2

<u>Class</u>	Department
E1	Electronics
E2	Electronics
IT1	IT

S3

<u>ID</u>	<u>Subject</u>	Mark
E1-001	Electronic Circuit	8
E1-001	Digital Technique	7
E1-002	Digital Technique	9
E1-002	Electronic Circuit	8
E2-001	Digital Technique	6
IT1-001	Electronic Circuit	10
IT1-002	Digital Technique	8

4. Phép tách

- *Một phép tách tốt cần phải::*
 - *Loại bỏ tất cả các bất thường*
 - *Lưu giữ được các tính chất mong muốn*
- *Làm thế nào đạt được?*
 - ➔ *Quá trình chuẩn hóa và các dạng chuẩn (Normalization and Normal Forms)*