

Chương 8:

Chuẩn hóa cơ sở dữ liệu

Tham khảo tài liệu [B]:

Chapter 13: Normalization

Nội dung chương 8:

- ◆ **Mục đích của việc chuẩn hóa CSDL**
- ◆ **Các lỗi liên quan đến việc dư thừa DL**
- ◆ **Nhận dạng các loại cập nhật DL dị thường (update anomalies) như khi chèn DL, xóa DL, chỉnh sửa DL**
- ◆ **Làm sao nhận định được tính thích hợp và chất lượng của việc thiết kế các quan hệ**

Nội dung chương 8: (tt)

- ◆ Cách dùng phụ thuộc hàm (functional dependencies) để nhóm các thuộc tính vào các quan hệ theo một dạng chuẩn định trước
- ◆ Cách thực hiện quá trình chuẩn hóa CSDL (process of normalization)
- ◆ Các nhận dạng một dạng chuẩn phổ biến (1NF, 2NF, 3NF, và dạng chuẩn Boyce–Codd)
- ◆ Giới thiệu dạng chuẩn 4 (4NF)

Chuẩn hóa (Normalization)

- ◆ Mục tiêu chính của việc phát triển mô hình mức logic của CSDL quan hệ là tạo được nơi lưu chứa và cung cấp dữ liệu có tính chính xác, thể hiện đầy đủ các mối quan hệ và ràng buộc của dữ liệu
- ◆ Để đạt được mục tiêu trên, tập các quan hệ trong CSDL phải có thiết kế thích hợp

Chuẩn hóa (Normalization) (tt)

- ◆ 4 dạng chuẩn thường dùng nhất là dạng chuẩn 1 (1NF), dạng chuẩn 2 (2NF), dạng chuẩn 3 (3NF), và dạng chuẩn Boyce–Codd (BCNF)
- ◆ Việc chuẩn hóa dựa vào các phụ thuộc hàm giữa các thuộc tính của quan hệ
- ◆ Một quan hệ có thể được chuẩn hóa thành một dạng chuẩn mới để ngăn ngừa việc xảy ra các dị thường trong cập nhật dữ liệu

Việc dư thừa dữ liệu (Data Redundancy)

- ◆ **Mục đích chính của việc thiết kế CSDL quan hệ là nhóm các thuộc tính vào các quan hệ nhằm giảm thiểu việc dư thừa dữ liệu**
- ◆ **Vấn đề dư thừa dữ liệu được minh họa sau đây qua quan hệ Staff, quan hệ Branch và quan hệ StaffBranch**

Việc dư thừa dữ liệu (Data Redundancy)

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Branch

branchNo	bAddress
B005	22 Deer Rd, London
B007	16 Argyll St, Aberdeen
B003	163 Main St, Glasgow

StaffBranch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Việc dư thừa dữ liệu (Data Redundancy)

- ◆ Quan hệ StaffBranch có dư thừa dữ liệu: chi tiết về 1 branch bị lặp lại trên các staff
- ◆ Ngược lại, chi tiết về 1 branch chỉ lưu 1 lần trong quan hệ Branch và chỉ có thuộc tính branchNo được lưu lặp lại để cho biết nơi làm của staff

Dị thường khi cập nhật (Update Anomalies)

- ◆ **Các quan hệ có dư thừa dữ liệu dễ bị dị thường khi cập nhật dữ liệu**
- ◆ **Các loại dị thường khi cập nhật dữ liệu:**
 - **Chèn (Insertion)**
 - **Xóa (Deletion)**
 - **Chỉnh sửa (Modification)**

Tính chất bảo toàn phụ thuộc hàm (Dependency Preservation) và ghép nối không mất thông tin (Lossless-join)

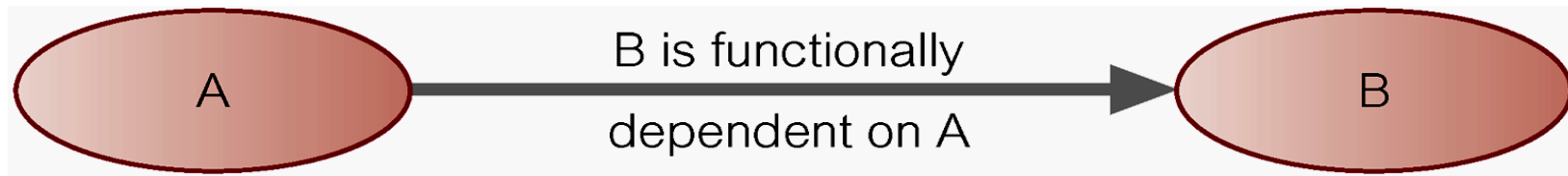
- ◆ 2 tính chất quan trọng của phép phân rã:
 - Tính chất ghép nối không mất thông tin (Lossless-join property): cho phép chúng ta tìm bất kỳ các thể hiện của quan hệ nguyên thủy từ các quan hệ tương ứng nhỏ hơn
 - Tính chất bảo toàn phụ thuộc hàm (Dependency preservation property): cho phép chúng ta áp dụng ràng buộc trên quan hệ nguyên thủy bằng cách áp dụng một số ràng buộc trên các quan hệ nhỏ hơn

Phụ thuộc hàm (Functional Dependency)

- ◆ Phụ thuộc hàm là khái niệm liên quan mật thiết với việc chuẩn hóa CSDL
- ◆ Phụ thuộc hàm
 - Miêu tả mối quan hệ giữa các thuộc tính trong một quan hệ (ký hiệu R là tập các thuộc tính)
 - Nếu A và B là các thuộc tính của quan hệ R , B phụ thuộc hàm vào A , hay còn nói A xác định B (ký hiệu $A \rightarrow B$), nếu với mỗi giá trị của A trong R thì sẽ ứng với đúng một giá trị của B trong R

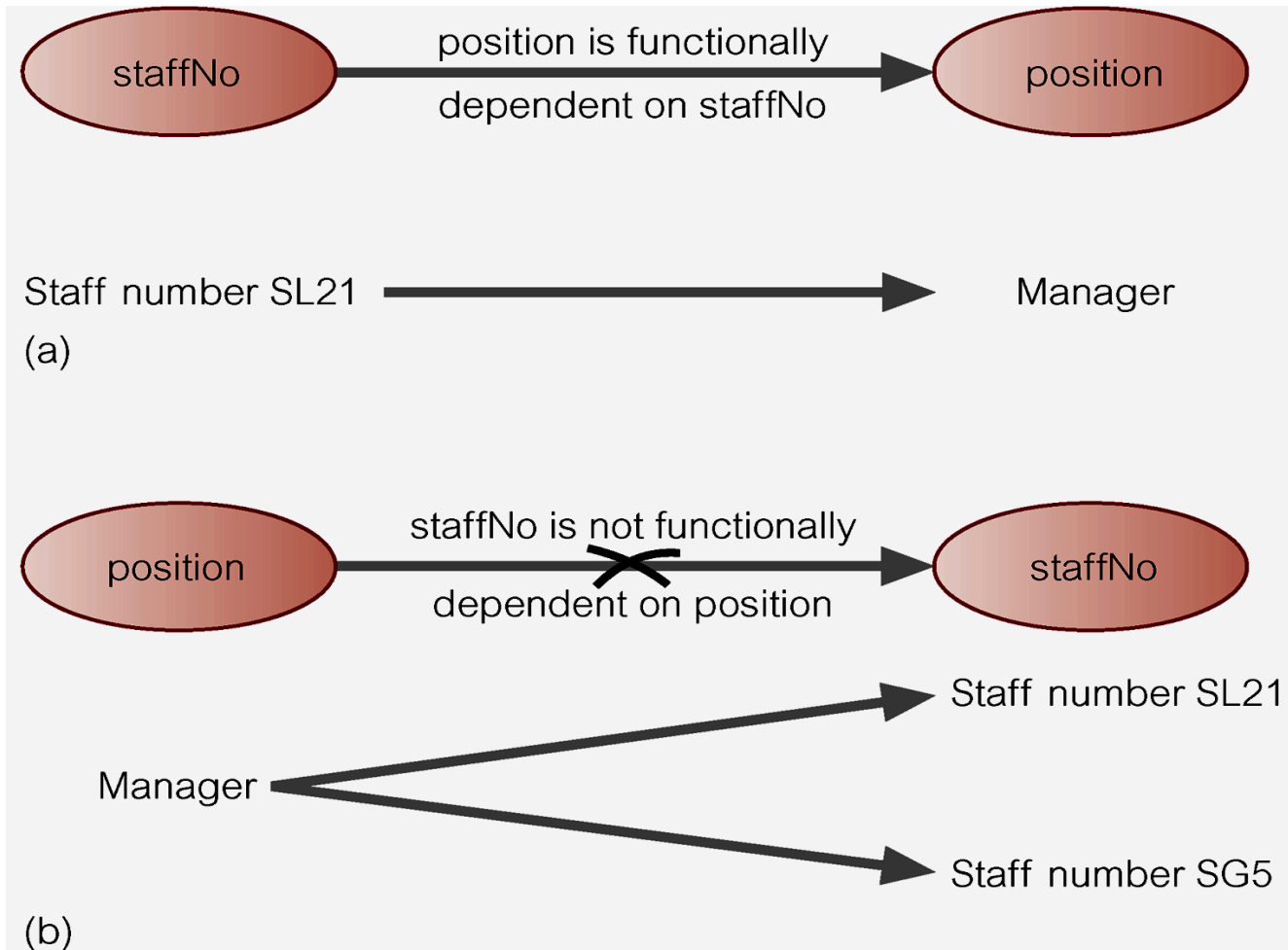
Phụ thuộc hàm (Functional Dependency)

- ◆ Phụ thuộc hàm thể hiện ý nghĩa của sự liên hệ giữa các thuộc tính với nhau trong 1 quan hệ



- ◆ *Phần tử quyết định (determinant) của một phụ thuộc hàm là một hay một nhóm các thuộc tính nằm bên trái dấu mũi tên*

Ví dụ về phụ thuộc hàm



Phụ thuộc hàm (Functional Dependency)

- ◆ Đặc điểm của chính của phụ thuộc hàm dùng trong việc chuẩn hóa là:
 - Có mối liên hệ 1:1 giữa các thuộc tính bên trái và bên phải của phụ thuộc hàm
 - Có mối liên hệ luôn luôn đúng
 - Không phải là mối liên hệ mặc nhiên đúng

Phụ thuộc hàm (Functional Dependency)

- ◆ Tập hợp tất cả các phụ thuộc hàm của 1 quan hệ có thể là rất lớn
- ◆ Phải tìm cách giảm tập các phụ thuộc hàm để có thể tính toán được
- ◆ Ta mong muốn xác định tập phụ thuộc hàm X nhỏ hơn tập Y là tập chứa toàn bộ các phụ thuộc hàm của quan hệ, nhưng từ các phụ thuộc hàm trong X phải suy ra được tất cả các phụ thuộc hàm của Y

Phụ thuộc hàm (Functional Dependency)

- ◆ Tập chứa tất cả các phụ thuộc hàm sinh ra từ các phụ thuộc hàm của X gọi là bao đóng của tập X (closure) ký hiệu X^+
- ◆ Sử dụng các qui tắc suy diễn từ các tiên đề của Armstrong (Armstrong's axioms) để tìm ra các phụ thuộc hàm mới từ 1 số phụ thuộc hàm cho trước

Phụ thuộc hàm (Functional Dependency)

- ◆ Với A, B, và C là tập con của tập các thuộc tính của quan hệ R, 3 tiên đề của Armstrong phát biểu như sau:

1. Tính phản xạ (Reflexivity)

Nếu B là tập con của A, thì $A \rightarrow B$

2. Tính tăng trưởng (Augmentation)

Nếu $A \rightarrow B$, thì $A, C \rightarrow B, C$

3. Tính bắc cầu (Transitivity)

Nếu $A \rightarrow B$ và $B \rightarrow C$, thì $A \rightarrow C$

Phụ thuộc hàm (Functional Dependency)

Bốn luật suy diễn từ các tiên đề Armstrong:

1. Luật tự xác định:

$A \rightarrow A$ là một phụ thuộc hàm luôn đúng

2. Luật tách vế phải khi vế phải có nhiều thuộc tính:

Nếu $A \rightarrow B, C$ thì $A \rightarrow B$ và $A \rightarrow C$

3. Luật hợp vế phải khi vế trái giống nhau:

Nếu $A \rightarrow B$ và $A \rightarrow C$ thì $A \rightarrow B, C$

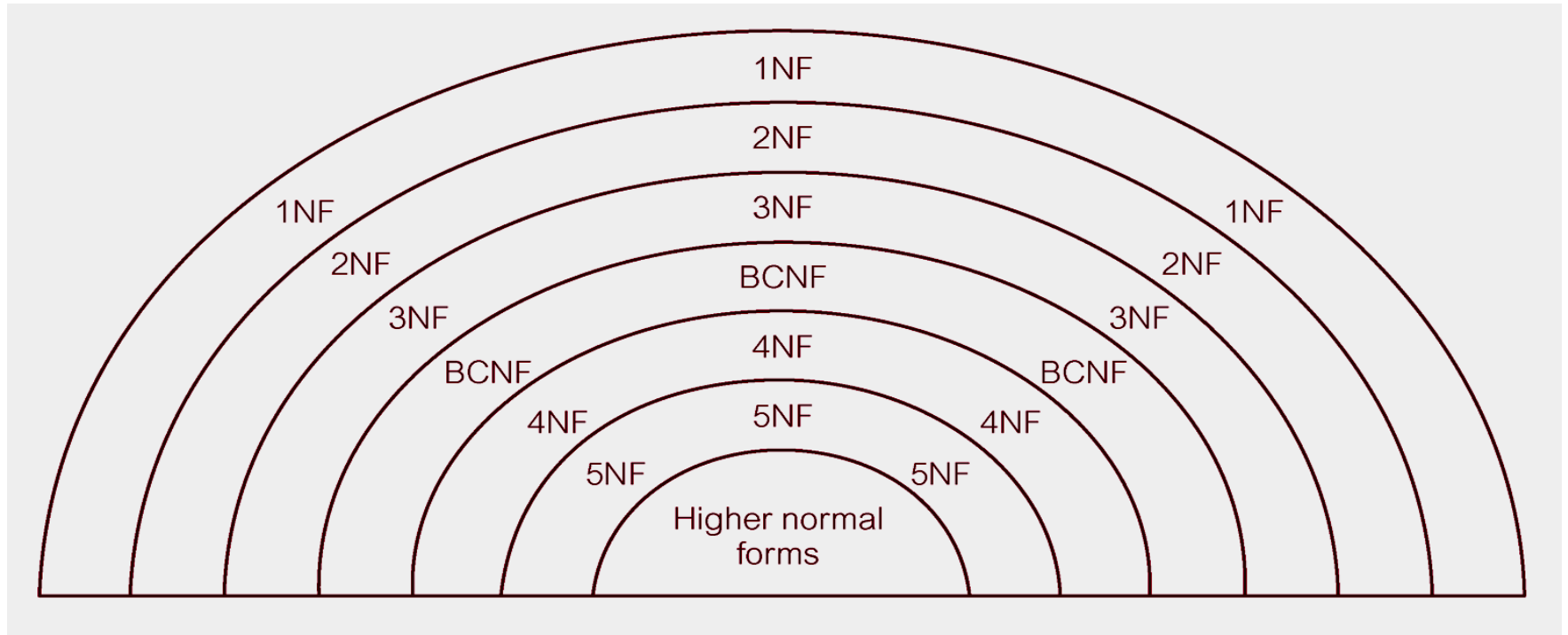
4. Luật tựa bắc cầu, hoặc bắc cầu giả:

Nếu $A \rightarrow B$ và $BZ \rightarrow C$ thì $AZ \rightarrow C$

Tiến trình chuẩn hóa

- ◆ Dựa vào khóa chính và các phụ thuộc hàm
- ◆ Gồm một loạt các bước, mỗi bước tương ứng với một dạng chuẩn
- ◆ Sau khi chuẩn hóa, các quan hệ sẽ bị ràng buộc nhiều hơn (tốt hơn) và ít bị dị thường khi cập nhật hơn

Mối quan hệ giữa các dạng chuẩn



Dạng chưa chuẩn hóa

Unnormalized Form (UNF)

- ◆ Một bảng có chứa một hay nhiều nhóm lặp
- ◆ Muốn tạo một bảng chưa chuẩn hóa:
 - Nhập thông tin từ 1 biểu mẫu nào đó vào 1 bảng

Dạng chuẩn 1 (1NF)

- ◆ Một ô dữ liệu phải chứa một giá trị đơn trị

Dạng chuẩn 2 (2NF)

- ◆ Dựa vào khái niệm phụ thuộc hàm đầy đủ (full functional dependency):
 - A và B là các thuộc tính của 1 quan hệ
 - B phụ thuộc hàm đầy đủ vào A nếu: B phụ thuộc hàm vào A nhưng không phụ thuộc hàm vào bất kỳ tập con thật sự nào của A
- ◆ 2NF – Là quan hệ ở dạng chuẩn 1 và tất cả các thuộc tính không phải là khóa phải phụ thuộc hàm đầy đủ vào khóa

Chuẩn hóa 1NF thành 2NF

- ◆ Xác định khóa của quan hệ đang ở 1NF
- ◆ Xác định các phụ thuộc hàm của quan hệ
- ◆ Nếu có phụ thuộc hàm $A \rightarrow B$ với A là con thật sự của khóa, thì ta loại thuộc tính B khỏi quan hệ đang xét và tạo 1 quan hệ mới chứa A, B

Dạng chuẩn 3 (3NF)

- ◆ Dựa vào khái niệm phụ thuộc hàm bắc cầu (transitive dependency):
 - A, B và C là các thuộc tính của 1 quan hệ có phụ thuộc hàm $A \rightarrow B$ và $B \rightarrow C$,
 - Ta nói C phụ thuộc hàm bắc cầu vào A qua B
- ◆ 3NF - Là quan hệ ở dạng chuẩn 2 và: không có thuộc tính không phải là khóa nào phụ thuộc hàm bắc cầu vào khóa

Chuẩn hóa 2NF thành 3NF

- ◆ Xác định khóa của quan hệ đang ở 2NF
- ◆ Xác định các phụ thuộc hàm của quan hệ
- ◆ Tách B, C thành 1 quan hệ mới và loại bỏ C khỏi quan hệ đang xét

Dạng chuẩn Boyce–Codd (BCNF)

- ◆ Là dạng chuẩn có tính đến tất cả các khóa ứng viên nên có nhiều ràng buộc hơn dạng chuẩn 3
- ◆ BCNF - Một quan hệ ở dạng chuẩn BCNF nếu và chỉ nếu mọi phần tử quyết định của các phụ thuộc hàm (nằm bên trái mũi tên) là các khóa ứng viên

So sánh 3NF và BCNF

- ◆ Xét phụ thuộc hàm $A \rightarrow B$, 3NF cho phép B là 1 thuộc tính của khóa và A không là 1 khóa ứng viên
- ◆ BCNF quy định A phải là khóa ứng viên
- ◆ Nếu là BCNF thì là 3NF
- ◆ Nếu là 3NF thì chưa chắc là BCNF

HẾT CHƯƠNG 8

**Mời các anh chị sinh viên tham khảo tài liệu
các nội dung tiếp theo.**

Chúc các anh chị học tốt.