

# Chương 8: Chuẩn hóa cơ sở dữ liệu

**Tham khảo tài liệu [A]:**

**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 là kỹ thuật dùng để tạo ra một tập các quan hệ có các đặc điểm mong muốn dựa vào các yêu cầu về dữ liệu của 1 xí nghiệp
- Chuẩn hóa là 1 cách tiếp cận từ dưới lên (bottom-up approach) để thiết kế CSDL, bắt đầu từ các mối liên hệ giữa các thuộc tính

- Quá trình chuẩn hóa được thực hiện qua nhiều bước. Mỗi bước tương ứng một dạng chuẩn
- Các dạng chuẩn:
  - Dạng chuẩn 1(1NF – first normal form)
  - Dạng chuẩn 2(2NF- second normal form)
  - Dạng chuẩn 3(3NF – third normal form)
  - Dạng chuẩn BCNF – Boyce Codd
  - Dạng chuẩn 4NF



# Chuẩn hóa (Normalization) (tt)

- 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

**Staff Branch**

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)





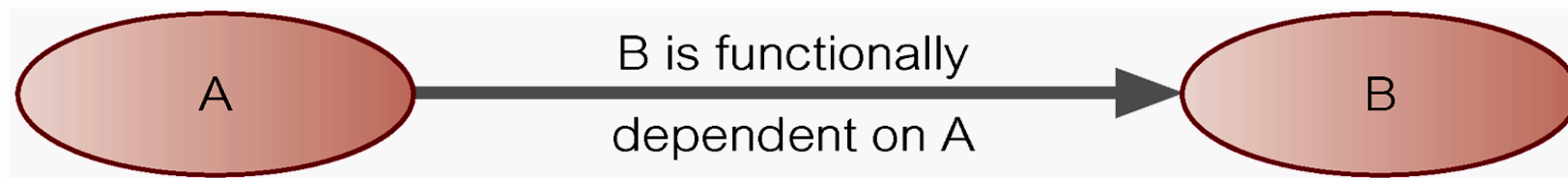
# Phụ thuộc hàm (Functional Dependency)

- 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)

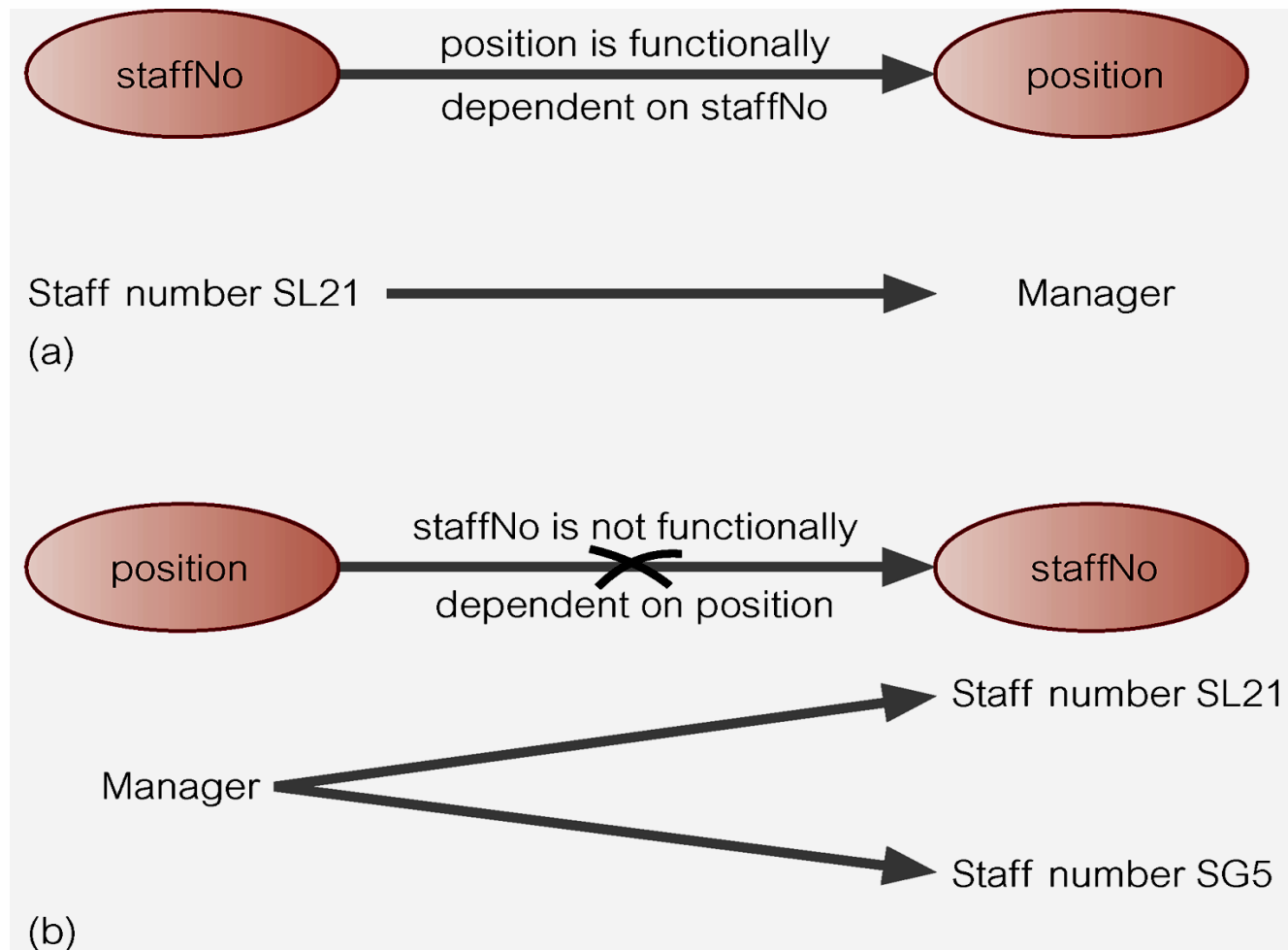
- Phụ thuộc hàm thể hiện ý nghĩa của sự liên hệ giữa các thuộc tính với nhau trong 1 quan hệ



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

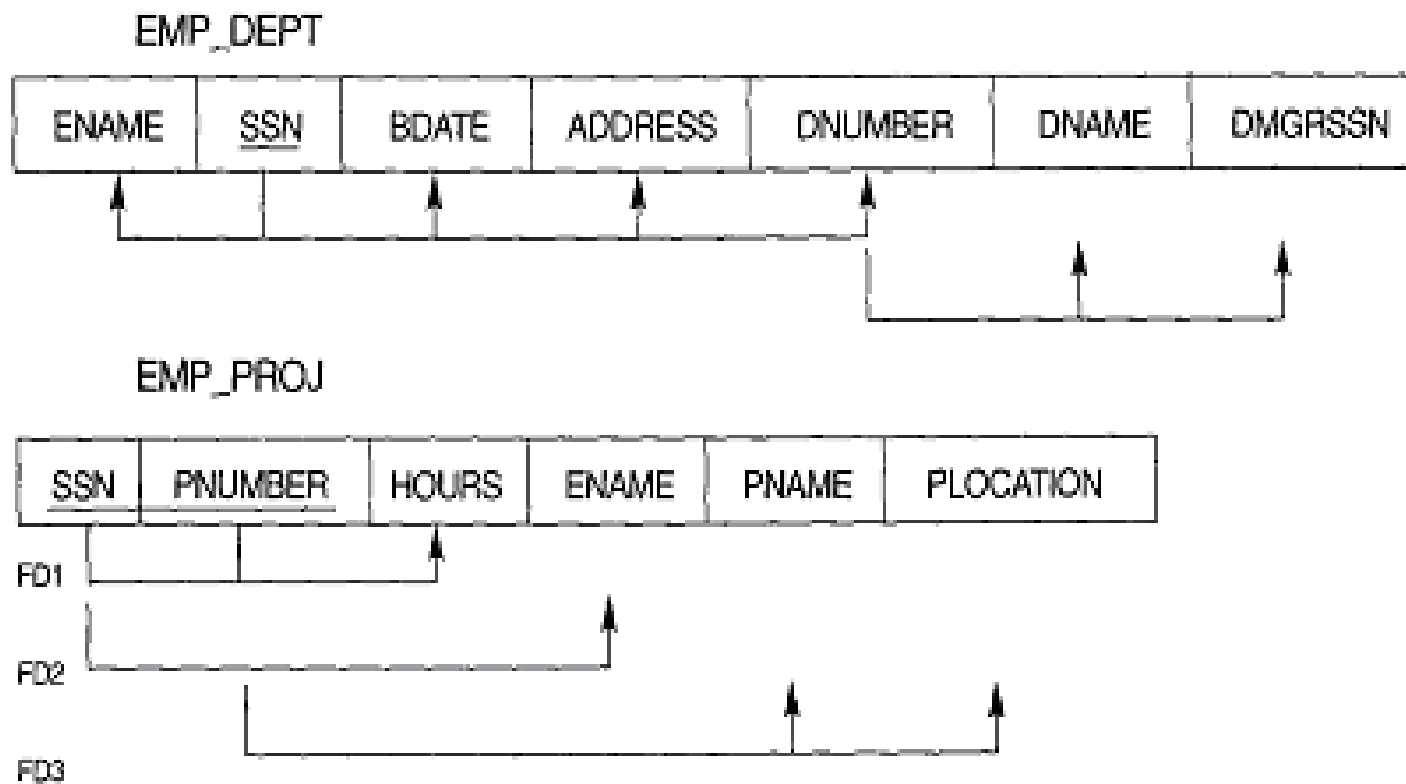


# Ví dụ về phụ thuộc hàm





# Ví dụ về phụ thuộc hàm





# Giải thuật kiểm tra phụ thuộc hàm

- Bài toán: cho quan hệ  $r$  và 1 phụ thuộc hàm  $f: X \rightarrow Y$ . Kiểm tra xem  $r$  thỏa mãn  $f$  hay không?
- Function Satisfies( $r, f: X \rightarrow Y$ )
  - Sắp thứ tự các bộ trong  $r$  theo các thuộc tính của  $X$
  - If mỗi tập các bộ có cùng giá trị  $X$  thì có cùng giá trị  $Y$  then
    - Satisfies = true
  - Else
    - Satisfies = false







# Phụ thuộc hàm (Functional Dependency)

- **Đặc điểm 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:

F1. Tính phản xạ (Reflexivity)

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

F2. Tính tăng trưởng (Augmentation)

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

F3. Tính bắc cầu (Transitivity)

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



# Hệ tiên đề Armstrong

- F4. Hợp (additivity):  $X \rightarrow Y$  và  $X \rightarrow Z \Rightarrow X \rightarrow YZ$
- F5. Chiều (projectivity):  $X \rightarrow YZ \Rightarrow X \rightarrow Y$
- F6. Bắc cầu giả (pseudotransitivity):  $X \rightarrow Y$  và  $YZ \rightarrow W \Rightarrow XZ \rightarrow W$





# Bao đóng của tập phụ thuộc hàm

- Bao đóng (closure) của tập phụ thuộc hàm  $F$  là 1 tập phụ thuộc hàm nhỏ nhất chứa  $F$  sao cho không thể áp dụng hệ tiên đề Armstrong trên tập này để tạo ra 1 phụ thuộc hàm khác không có trong tập hợp này
- Ký hiệu  $F^+$
- $F^+$  là 1 tập hợp các FD được suy diễn từ  $F$





# Giải thuật tìm bao đóng của tập thuộc tính trên tập phụ thuộc hàm

- Input: tập thuộc tính  $X$  và tập phụ thuộc hàm  $F$
- Output: bao đóng của  $X$  dựa trên  $F$

Procedure Closure( $X, F, \text{Closure\_X}$ )

Begin

$\text{Closure\_X} := X;$

    repeat

$\text{Old\_X} := \text{Closure\_X};$

        for mỗi  $W \rightarrow Z$  trong  $F$  do

            if  $W \subseteq \text{Closure\_X}$  then

$\text{Closure\_X} := \text{Closure\_X} \cup Z;$

    until  $\text{Closure\_X} = \text{Old\_X};$

End;



# Ví dụ tìm bao đóng của X

$$F = \{ \text{SSN} \rightarrow \text{ENAME}, \\ \text{PNUMBER} \rightarrow \{ \text{PNAME}, \text{PLOCATION} \}, \\ \{ \text{SSN}, \text{PNUMBER} \} \rightarrow \text{HOURS} \}$$

Suy ra:

$$\{ \text{SSN} \}^+ = \{ \text{SSN}, \text{ENAME} \}$$
$$\{ \text{PNUMBER} \}^+ = \{ \text{PNUMBER}, \text{PNAME}, \text{PLOCATION} \}$$
$$\{ \text{SSN}, \text{PNUMBER} \}^+ = \{ \text{SSN}, \text{PNUMBER}, \text{ENAME}, \text{PNAME}, \text{PLOCATION}, \text{HOURS} \}$$





# Ví dụ tìm bao đóng của X

■ Cho  $R(A,B,C,D,E,G)$

và tập  $F=\{f1:D\rightarrow B, f2: A\rightarrow C, f3: AD\rightarrow E, f4:C\rightarrow G\}$

■ Tìm  $A^+_F$ :

- $A^+_F = \{A\}$

- Duyệt F lần 1: Từ  $f2 \rightarrow A^+_F = \{AC\}$ ; Từ  $f4 \rightarrow A^+_F = \{ACG\}$

- Duyệt F lần 2:  $A^+_F$  không thay đổi

→  $A^+_F = \{ACG\}$

■ Tìm  $\{AD\}^+_F$  ???

1. Cho  $R(A,B,C,D,E,G,H)$  và tập  $F=\{f1:B\rightarrow A, f2: DA\rightarrow CE, f3: D\rightarrow H, f4:GH\rightarrow C, f5:AC \rightarrow D\}$

$X=\{AC\}$ . Tìm  $X_F^+$

2. Cho  $F=\{D\rightarrow B, A\rightarrow C, AD\rightarrow E, C\rightarrow B\}$ . Kiểm tra  $F$  có bao hàm  $A\rightarrow B$

3. Cho  $Q(ABCDEFG)$  và tập  $F = \{AB\rightarrow E, C\rightarrow G, BG\rightarrow A, AD\rightarrow C, B\rightarrow E\}$

Tìm  $\{BA\}_F^+$  ,  $\{CEG\}_F^+$



# Giải thuật tìm khóa của lược đồ quan hệ

- Nhập:  $R(U)$  và tập phụ thuộc hàm  $F$
- Xuất: tập hợp  $K$  bao gồm tất cả khóa của  $R$
- Tập thuộc tính nguồn (TN) chứa tất cả các thuộc tính xuất hiện ở vế trái và không xuất hiện ở vế phải của các phụ thuộc hàm và các thuộc tính không xuất hiện ở cả vế trái lẫn vế phải của các phụ thuộc hàm

$$TN = U - \bigcup_{f \in F} \text{right}(f)$$



# Giải thuật tìm khóa của lược đồ quan hệ

- Tập thuộc tính đích (TD) chứa tất cả các thuộc tính có xuất hiện ở vế phải và không xuất hiện ở vế trái của các phụ thuộc hàm

$$TD = \bigcup \forall f \in F \text{ right}(f) - \bigcup \forall f \in F \text{ left}(f)$$

- Tập thuộc tính trung gian (TG) chứa tất cả các thuộc tính xuất hiện ở cả vế trái lẫn vế phải của các phụ thuộc hàm





# Thuật toán tìm tất cả khóa

- Bước 1: tạo tập thuộc tính nguồn TN. Tập thuộc tính trung gian TG
- Bước 2: if  $TG = \emptyset$  then lược đồ quan hệ chỉ có 1 khóa K  
    K=TN Kết thúc  
    Ngược lại qua bước 3
- Bước 3: tìm tất cả các tập con Xi của tập trung gian TG



# Thuật toán tìm tất cả khóa (tt)

- Bước 4: tìm các siêu khóa  $S_i$  bằng cách  $\forall X_i$   
if  $(TN \cup X_i)^+ = Q^+$  then  $S_i = TN \cup X_i$
- Bước 5: tìm khóa bằng cách loại bỏ các siêu khóa không tối thiểu  
 $\forall S_i, S_j \in S$   
if  $S_i \subset S_j$  then Loại  $S_j$  ra khỏi tập siêu khóa  $S$   
 $S$  còn lại chính là tập khóa cần tìm



- Cho  $R(A,B,C,D,E,G)$  và  $F=\{D \rightarrow B, A \rightarrow C, AD \rightarrow E, C \rightarrow G\}$ . Tìm tất cả các khóa của  $R$
- B1:  $TN=\{AD\}$ ,  $TG=\{C\}$
- $X_i$  là các tập con của  $TG$

$X_i$	$X_i \cup TN$	$(X_i \cup TN)^+$	Siêu khóa	Khóa
$\emptyset$	AD	ADBCEG= $R^+$	AD	AD
C	ADC	ADBCEG= $R^+$	ADC	

- Cho  $R(A,B,C,D,E,G)$  và  $F=\{A \rightarrow D, C \rightarrow AG, AB \rightarrow EC\}$ . Tìm khóa của  $R$ ?
- $TN=\{B\}$  ,  $TG=\{AC\}$
- Khóa của  $R$  là  $\{AB\}$ ,  $\{BC\}$

$X_i$	$X_i \cup TN$	$(X_i \cup TN)^+$	Siêu khóa	Khóa
$\emptyset$	B	B		
C	CB	ABCDEG=R+	BC	BC
A	AB	ABCDEG=R+	AB	AB
AC	ABC	ABCDEG=R+	ABC	



- Cho  $Q(ABCDEFG)$  và tập  $F =$
- $\{AB \rightarrow E,$
- $C \rightarrow G,$
- $BG \rightarrow A,$
- $AD \rightarrow C,$
- $B \rightarrow E\} .$
- Tìm khóa của  $Q$ ?



# 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



# 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



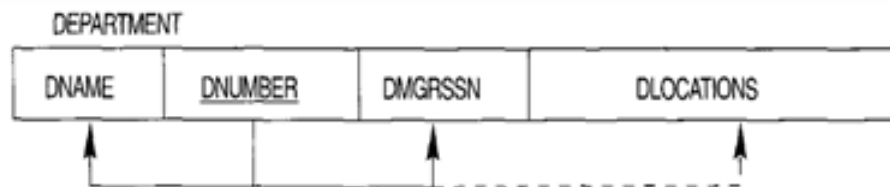


# Dạng chuẩn 1 (1NF)

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



# Dạng chuẩn 1 (1NF)



DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

Dư  
thừa

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	<u>DLOCATION</u>
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston



# Chuẩn hóa UNF thành 1NF

- Chọn ra 1 hay 1 nhóm thuộc tính đóng vai trò là khóa của bảng
- Xác định các nhóm lặp gây ra việc lặp các thuộc tính khóa
- Loại bỏ các nhóm lặp bằng cách:
  - Nhập dữ liệu vào các ô lặp lại còn trống hoặc bằng cách
  - Tạo 1 quan hệ mới chứa các thuộc tính lặp và các thuộc tính khóa

# Dạng chuẩn 1 (1NF)

DEPARTMENT:

DName	<u>DNumber</u>	DMgrSsn
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

DEPT\_LOCATIONS:

<u>DNumber</u>	<u>DLocation</u>
5	Bellaire
5	Sugarland
5	Houston
4	Stafford
1	Houston



## Dạng chuẩn 2 (2NF)

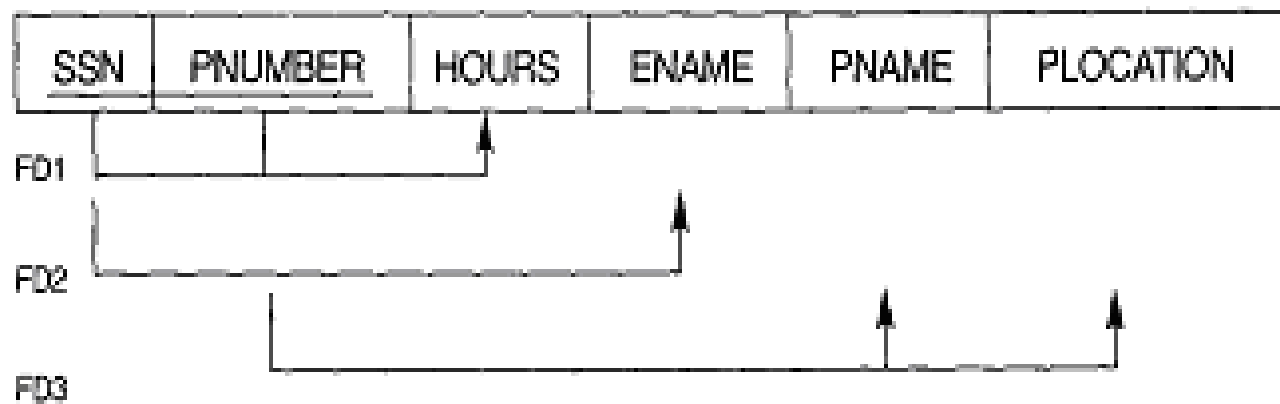
- Dựa vào khái niệm phụ thuộc hàm đầy đủ (full functional dependency):
  - A và B là các thuộc tính của 1 quan hệ
  - B phụ thuộc hàm đầy đủ vào A nếu: B phụ thuộc hàm vào A nhưng không phụ thuộc hàm vào bất kỳ tập con thật sự nào của A
- 2NF – Là quan hệ ở dạng chuẩn 1 và tất cả các thuộc tính không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





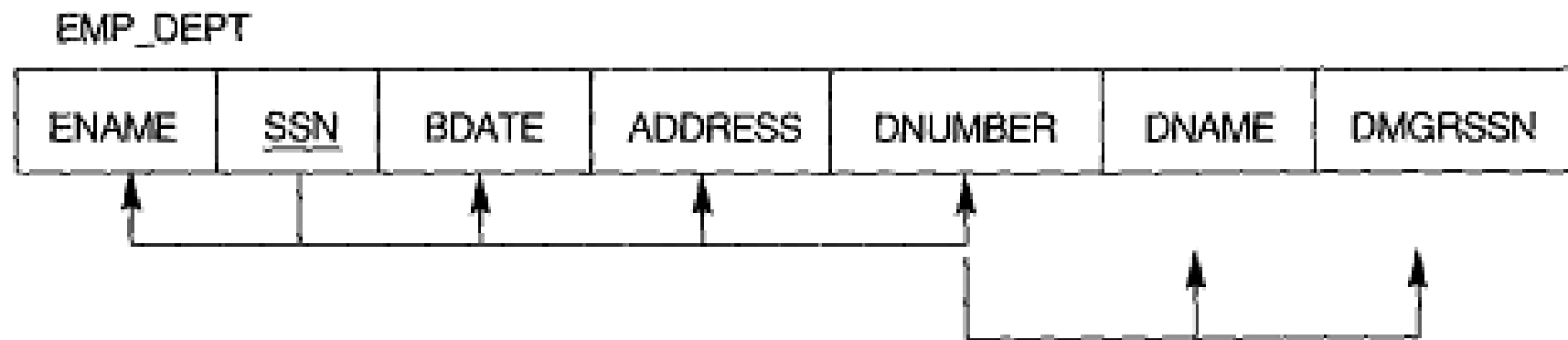
# Dạng chuẩn 2 (2NF)

EMP\_PROJ





# Dạng chuẩn 2 (2NF)





# 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  $A, B$

# Dự thừa dữ liệu

EMP\_DEPT

ENAME	<u>SSN</u>	BDATE	ADDRESS	DNUMBER	DNAME	DMGRSSN
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5	Research	333445555
Wong, Franklin T.	333445555	1955-12-08	638 Voss, Houston, TX	5	Research	333445555
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4	Administration	987654321
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4	Administration	987654321
Narayan, Ramesh K.	666884444	1962-09-15	975 FireOak, Humble, TX	5	Research	333445555
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5	Research	333445555
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4	Administration	987654321
Borg, James E.	888665555	1937-11-10	450 Stone, Houston, TX	1	Headquarters	888665555

Dự thừa dữ liệu



## 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



# Chuẩn hóa 2NF thành 3NF

EMPLOYEE

ENAME	<u>SSN</u>	BDATE	ADDRESS	DNUMBER
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5
Wong, Franklin T.	333445555	1955-12-08	638 Voss, Houston, TX	5
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4
Narayan, Remesh K.	666884444	1962-09-15	975 Fire Oak, Humble, TX	5
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4
Borg, James E.	888665555	1937-11-10	450 Stone, Houston, TX	1

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

- Cho  $Q(ABCDE)$  và tập  $F =$   
 $\{A \rightarrow BC,$   
 $BD \rightarrow E,$   
 $B \rightarrow C\}.$
- Kiểm tra quan hệ  $Q$  với tập  $F$  trên có thuộc dạng chuẩn 2NF hoặc 3NF hay không?



- Cho lược đồ quan hệ **Q** như sau: Q(MAHV, MAMH, LANTHI, HOTEN, NGAYSINH, TENMH, SOTIET, DIEMTHI)

Với MAHV, MAMH, LANTHI là khóa.

- Cho biết quan hệ trên thoả mãn chuẩn 2NF hay chưa? Nếu chưa thoả chuẩn 2NF tìm cách đưa quan hệ trên thoả chuẩn 2NF nhưng không làm mất dữ liệu.
- Từ câu a kiểm tra quan hệ trên có thoả chuẩn 3NF hay chưa? Nếu chưa thoả đưa quan hệ trên thoả dạng chuẩn 3NF.



# 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
- Tất cả các phụ thuộc hàm  $X \rightarrow A$  trong R đều phải có X là khoá của R.



# 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





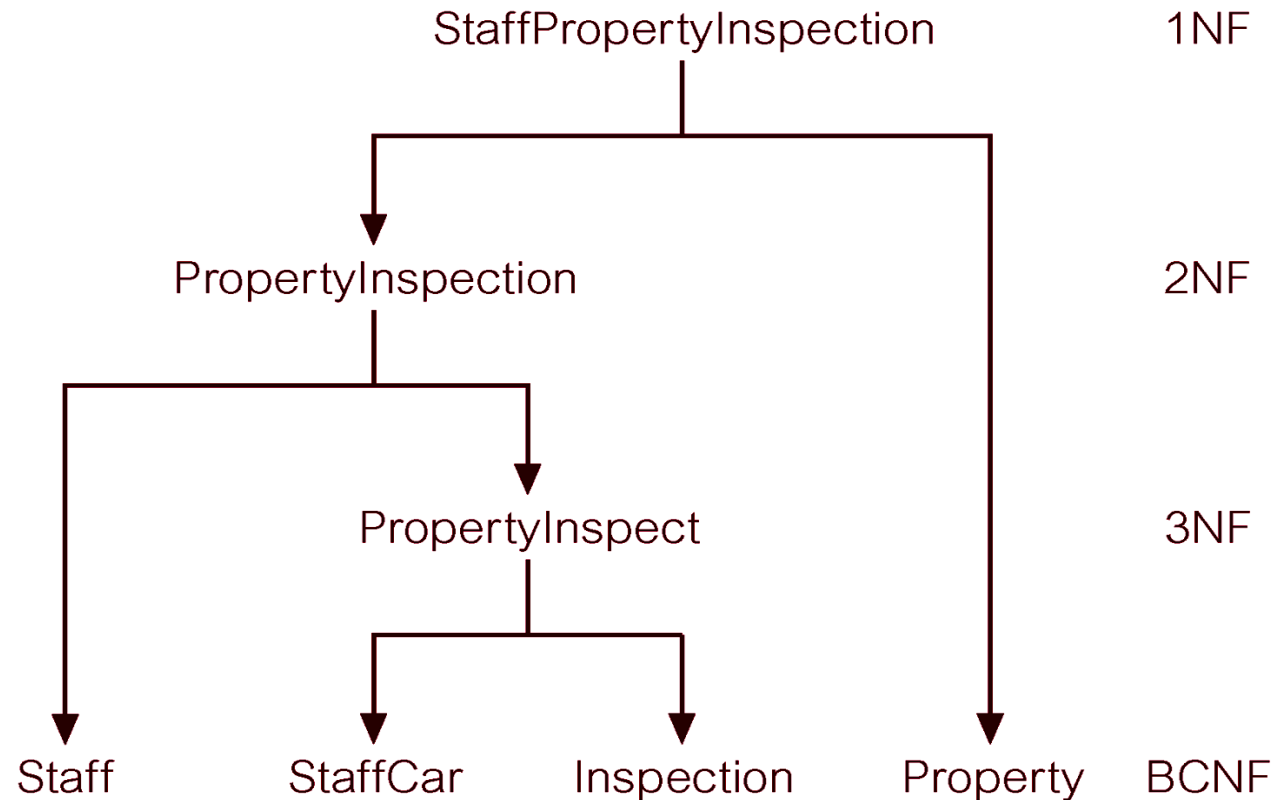
# Dạng chuẩn BCNF

- Trường hợp có thể gây ra sự vi phạm dạng chuẩn BCNF trong 1 quan hệ:
  - Có chứa các khóa ứng viên có 2 hay nhiều thuộc tính
  - Các khóa ứng viên có phần giao nhau (tức có thuộc tính chung)





# Ví dụ chuẩn hóa từ UNF thành BCNF



- Xét  $U = \{ABCD\}$ ,  $F = \{AB \rightarrow CD, AC \rightarrow BD\}$  có 2 khóa: AB và AC
- Vì 2 phụ thuộc hàm này đều có phía trái là khóa, nên lược đồ ở dạng BCNF



# Phụ thuộc đa trị

- Mặc dầu dạng chuẩn BCNF loại bỏ được sự dị thường nhờ vào xét các phụ thuộc hàm, có 1 loại phụ thuộc khác có thể gây ra việc dư thừa dữ liệu gọi là phụ thuộc đa trị (multi-valued dependency (MVD))
- MVD: xét A, B, C là các thuộc tính của 1 quan hệ, mỗi giá trị của A xác định 1 tập giá trị của B và 1 tập giá trị của C, nhưng tập giá trị của B và của C không phụ thuộc lẫn nhau. Ký hiệu:

$A \twoheadrightarrow B$

$A \twoheadrightarrow C$



# Phụ thuộc đa trị (tt)

- Xét  $A \longrightarrow \triangleright \triangleright B$
- Là phụ thuộc đa trị mặc nhiên nếu:
  - B là tập con của A, hay
  - $A \cup B = R$
- Là phụ thuộc đa trị không mặc nhiên nếu không phải là phụ thuộc đa trị mặc nhiên
- Phụ thuộc đa trị không mặc nhiên có tạo ràng buộc cho quan hệ, còn phụ thuộc đa trị mặc nhiên không có tạo ràng buộc cho quan hệ





# Dạng chuẩn 4 (4NF)

- Là quan hệ ở dạng chuẩn BCNF và không chứa các phụ thuộc đa trị không mặc nhiên



**BranchStaffOwner**

branchNo	sName	oName
B003	Ann Beech	Carol Farrel
B003	David Ford	Carol Farrel
B003	Ann Beech	Tina Murphy
B003	David Ford	Tina Murphy



**BranchStaff**

branchNo	sName
B003	Ann Beech
B003	David Ford

**BranchOwner**

branchNo	oName
B003	Carol Farrel
B003	Tina Murphy

**HẾT CHƯƠNG 8**