BỘ MÔN HỆ THỐNG THÔNG TIN
KHOA CÔNG NGHỆ THỐNG TIN – ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP HCM

CƠ SỞ DỮ LIỆU NÂNG CAO

Chương 06:
DẠNG CHUẨN & CHUẨN HÓA

Giảng viên: TS. Nguyễn Trần Minh Thư

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Mục tiêu chương

- Mục tiêu thiết kế quan niệm
- Dư thừa dữ liệu
- Tiêu chuẩn dạng chuẩn
- Tiêu chuẩn thiết kế tương đương
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
  - Biểu diễn trọn vẹn
- Chuẩn hóa: Tiếp cận phân rã

Mục tiêu chương

- Mục tiêu thiết kế quan niệm
- Dư thừa dữ liệu
- Tiêu chuẩn dạng chuẩn
- Tiêu chuẩn thiết kế tương đương
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
  - Biểu diễn trọn vẹn
- Chuẩn hóa: Tiếp cận phân rã

Mục tiêu thiết kế mức quan niệm

- Hai tiêu chuẩn quan trọng cần đạt được trong quá trình thiết kế CSDL ở mức quan niệm:
  - Cấu trúc CSDL kết quả (đầu ra của giai đoạn thiết kế mức quan niệm) cần đạt dạng chuẩn cao nhất
  - Cấu trúc CSDL kết quả phải tương đương với cấu trúc ban đầu

Mục tiêu thiết kế mức quan niệm

- Tiêu chuẩn về dạng chuẩn:
  - Giảm tối đa sự trùng lắp thông tin ➔ tránh được một số bất tiện khi cập nhật CSDL
  - Các phụ thuộc dữ liệu được kiểm tra dễ dàng, đơn giản và tương đối ít tốn kém nhất
- Tiêu chuẩn tương đương:
  - Đảm bảo các thông tin trong quan hệ phổ quát sẽ được tìm thấy đầy đủ trong CSDL của cấu trúc kết quả (gồm nhiều quan hệ con).

Mục tiêu thiết kế mức quan niệm

- Một CSDL thỏa mãn hai tiêu chuẩn về DC và tính tương đương, đảm bảo cho việc khai thác nó được thuận lợi trên cả ba phương diện:
  - *Truy vấn*: tiêu chuẩn tương đương đảm bảo các thông tin được truy xuất từ CSDL là những thông tin đã được phân tích.
  - *Cập nhật:* Tiêu chuẩn dạng chuẩn giảm bớt các tình huống thông tin mâu thuẩn sau khi cập nhật.
  - *Kiểm tra RBTV:* cả hai tiêu chuẩn đều hướng đến mục tiêu là kiểm tra RBTV dạng phụ thuộc dữ liệu được thuận lợi.

Mục tiêu chương

- Mục tiêu thiết kế quan niệm
- Dư thừa dữ liệu
- Tiêu chuẩn dạng chuẩn
- Tiêu chuẩn thiết kế tương đương
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
  - Biểu diễn trọn vẹn
- Chuẩn hóa: Tiếp cận phân rã

Dư thừa dữ liệu (Data Redundancy)

EMPLOYEE2

EmpID	Name	DeptName	Salary	CourseTitle	DateCompleted
100	Margaret Simpson	Marketing	48,000	SPSS	6/19/201X
100	Margaret Simpson	Marketing	48,000	Surveys	10/7/201X
140	Alan Beeton	Accounting	52,000	Tax Acc	12/8/201X
110	Chris Lucero	Info Systems	43,000	Visual Basic	1/12/201X
110	Chris Lucero	Info Systems	43,000	C++	4/22/201X
19 0	Lorenzo Davis	Finance	55,000		
15 0	Susan Martin	Marketing	42,000	SPSS	6/19/201X
15 0	Susan Martin	Marketing	42,000	Java	8/12/201X

Question–Is this a relation?

Answer–Yes: Unique rows and no multivalued attributes

Question–What's the primary key?

Answer–Composite: EmpID, CourseTitle

Dị thường (Anomalies)

EMPLOYEE2

EmpID	Name	DeptName	Salary	CourseTitle	DateCompleted
100	Margaret Simpson	Marketing	48,000	SPSS	6/19/201X
100	Margaret Simpson	Marketing	48,000	Surveys	10/7/201X
140	Alan Beeton	Accounting	52,000	Tax Acc	12/8/201X
110	Chris Lucero	Info Systems	43,000	Visual Basic	1/12/201X
110	Chris Lucero	Info Systems	43,000	C++	4/22/201X
19 0	Lorenzo Davis	Finance	55,000		
15 0	Susan Martin	Marketing	42,000	SPSS	6/19/201X
15 0	Susan Martin	Marketing	42,000	Java	8/12/201X

Insertion

Can't enter a new employee without having the employee take a class (or at least empty fields of class information)

Deletion

If we remove employee 140, we lose information about the existence of a Tax Acc class)

Modification

Giving a salary increase to employee 100 forces us to update multiple records

Dị thường (Anomalies)

EMPLOYEE2

EmpID	Name	DeptName	Salary	CourseTitle	DateCompleted
100	Margaret Simpson	Marketing	48,000	SPSS	6/19/201X
100	Margaret Simpson	Marketing	48,000	Surveys	10/7/201X
140	Alan Beeton	Accounting	52,000	Tax Acc	12/8/201X
110	Chris Lucero	Info Systems	43,000	Visual Basic	1/12/201X
110	Chris Lucero	Info Systems	43,000	C++	4/22/201X
190	Lorenzo Davis	Finance	55,000		
150	Susan Martin	Marketing	42,000	SPSS	6/19/201X
150	Susan Martin	Marketing	42,000	Java	8/12/201X

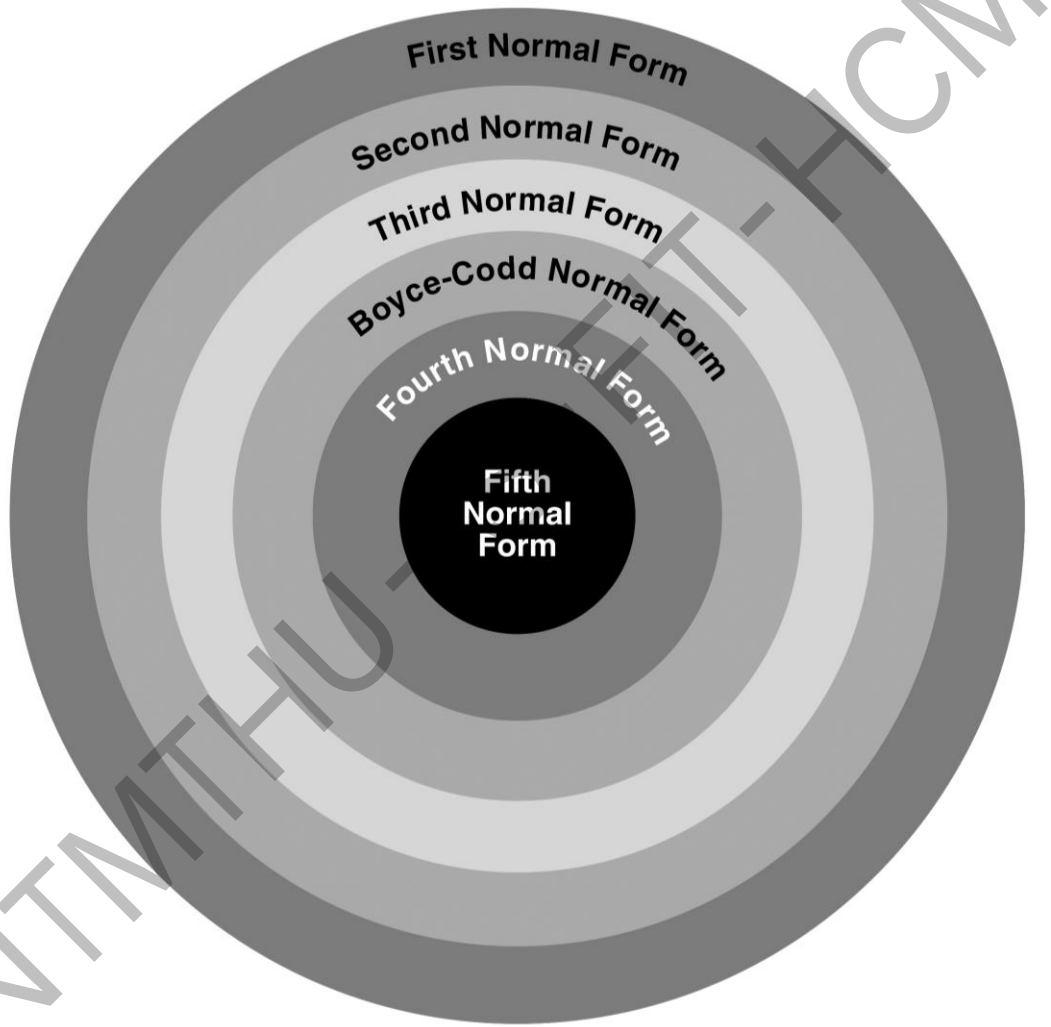
Why do these anomalies exist?

Because there are two themes (entity types) in this one relation. This results in data duplication and an unnecessary dependency between the entities.

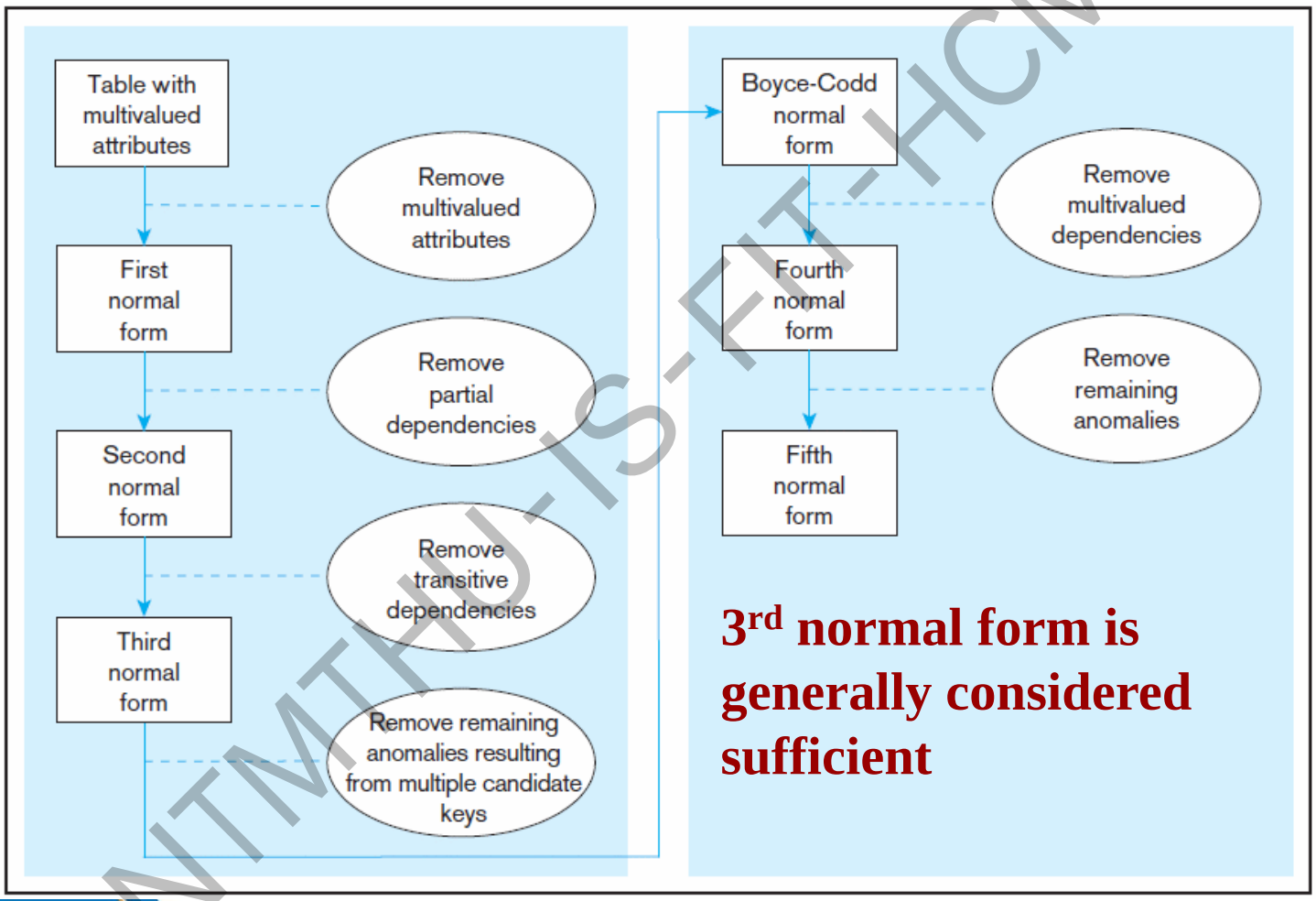
Mục tiêu chương

- Mục tiêu thiết kế quan niệm
- Dư thừa dữ liệu
- Tiêu chuẩn dạng chuẩn
- Tiêu chuẩn thiết kế tương đương
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
  - Biểu diễn trọn vẹn
- Chuẩn hóa: Tiếp cận phân rã

DẠNG CHUẨN (NORMAL FORMS)



CHUẨN HÓA (NORMALIZATION)



FIRST NORMAL FORM (1 NF)

A table is in first normal form (1NF) if it meets the following criteria:

1. The data are stored in a two-dimensional table.

2. Every attribute value is atomic

3. There are no repeating groups (no multivalued attributes)

Note: A repeating group is directly analogous to a multivalued attribute in an ER diagram.

Emp#	First	Last	Children's Names	Children's Birthdates
1001	Jane	Doe	Mary, Sam	1/9/02, 5/15/04
1002	John	Doe	Lisa, David	1/9/00, 5/15/01
1003	Jane	Smith	John, Pat, Lee, Mary	10/5/04, 10/12/00, 6/6/2006, 8/21/04
1004	John	Smith	Michael	7/4/06
1005	Jane	Jones	Edward, Martha	10/21/05, 10/15/99

FIRST NORMAL FORM (1 NF)

REPEATING GROUP

Emp#	First	Last	Child Name 1	Child Bdate 1	Child Name 2	Child Bdate 2	Child Name 3	Child Bdate 3
1001	Jane	Doe	Mary	1/1/02	Sam	5/15/04		
1002	John	Doe	Lisa	1/1/00	David	5/15/01		
1003	Jane	Smith	John	10/5/04	Pat	10/12/00	Lee	6/6/06
1004	John	Smith	Michael	7/4/06				
1005	Joe	Jones	Edward	10/21/05	Martha	10/15/99		

A relation handling repeating group in the wrong way

FIRST NORMAL FORM (1 NF)

REPEATING GROUP

Employee

Emp#	First	Last
1001	Jane	Doe
1002	John	Doe
1003	Jane	Smith
1004	John	Smith
1005	Joe	Jones

Children

1001	Mary	1/1/02
1001	Sam	5/15/04
1002	Lisa	1/1/00
1002	David	5/15/01
1003	John	10/5/04
1003	Pat	10/12/00
1003	Lee	6/6/06
1003	Mary	8/21/04
1004	Michael	7/4/06
1005	Edward	10/21/05
1005	Martha	1015/99

The correct way to handle a repeating group

FIRST NORMAL FORM (1 NF)

Table (INVOICE) with multivalued attributes, not in 1NF

OrderID	Order Date	Customer ID	Customer Name	Customer Address	ProductID	Product Description	Product Finish	Product StandardPrice	Ordered Quantity
1006	10/24/2010	2	Value Furniture	Plano, TX	7	Dining Table	Natural Ash	800.00	2
					5	Writer's Desk	Cherry	325.00	2
					4	Entertainment Center	Natural Maple	650.00	1
1007	10/25/2010	6	Furniture Gallery	Boulder, CO	11	4–Dr Dresser	Oak	500.00	4
					4	Entertainment Center	Natural Maple	650.00	3

Note: This is NOT a relation.

FIRST NORMAL FORM (1 NF)

Table with no multivalued attributes and unique rows, in 1NF

OrderID	Order Date	Customer ID	Customer Name	Customer Address	ProductID	Product Description	Product Finish	Product StandardPrice	Ordered Quantity
1006	10/24/2010	2	Value Furniture	Plano, TX	7	Dining Table	Natural Ash	800.00	2
1006	10/24/2010	2	Value Furniture	Plano, TX	5	Writer's Desk	Cherry	325.00	2
1006	10/24/2010	2	Value Furniture	Plano, TX	4	Entertainment Center	Natural Maple	650.00	1
1007	10/25/2010	6	Furniture Gallery	Boulder, CO	11	4–Dr Dresser	Oak	500.00	4
1007	10/25/2010	6	Furniture Gallery	Boulder, CO	4	Entertainment Center	Natural Maple	650.00	3

Note: This is a relation, but not a well-structured one.

FIRST NORMAL FORM (1 NF)

Anomalies in INVOICE Table

- **Insertion**–if new product is ordered for order 1007 of existing customer, customer data must be re-entered, causing duplication
- **Deletion**–if we delete the Dining Table from Order 1006, we lose information concerning this item's finish and price
- **Update**–changing the price of product ID 4 requires update in multiple records

Why do these anomalies exist?

Because there are multiple themes (entity types) in one relation. This results in duplication and an unnecessary dependency between the entities.

SECOND NORMAL FORM (2NF)

A table is in second normal form (2NF) if it meets the following criteria:

1. The relation is in first normal form.

2. Every non-key attribute is fully functionally dependent on the ENTIRE primary key:

- Every non-key attribute must be defined by the entire key, not by only part of the key
- No partial functional dependencies

Partial dependency: Functional dependence in which the determinant is only part of the primary key

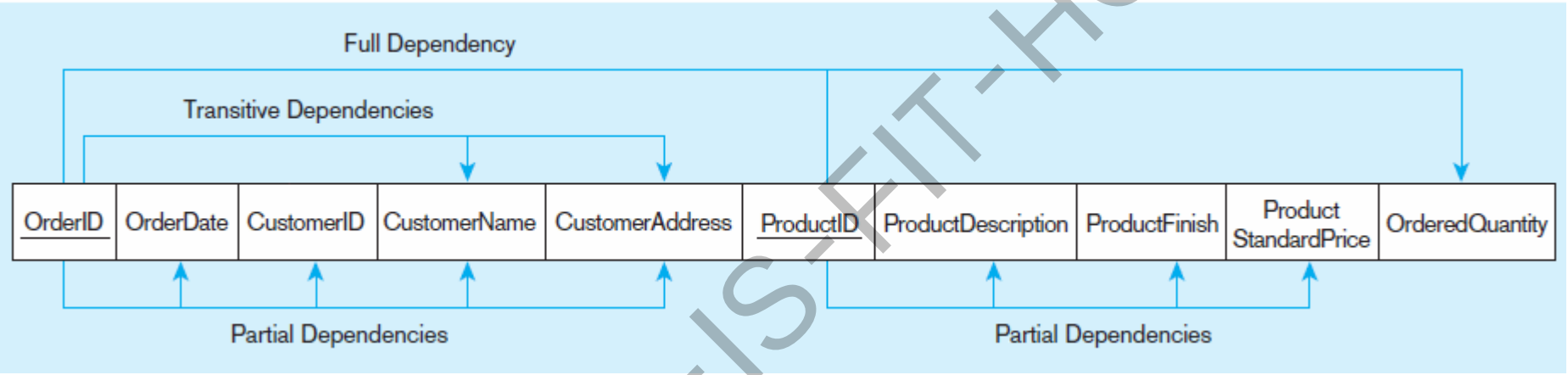
Assumption - One candidate key

Straight forward

Easy to identify

SECOND NORMAL FORM (2NF)

Functional dependency diagram for INVOICE



OrderID ➔ OrderDate, CustomerID, CustomerName, CustomerAddress

CustomerID ➔ CustomerName, CustomerAddress

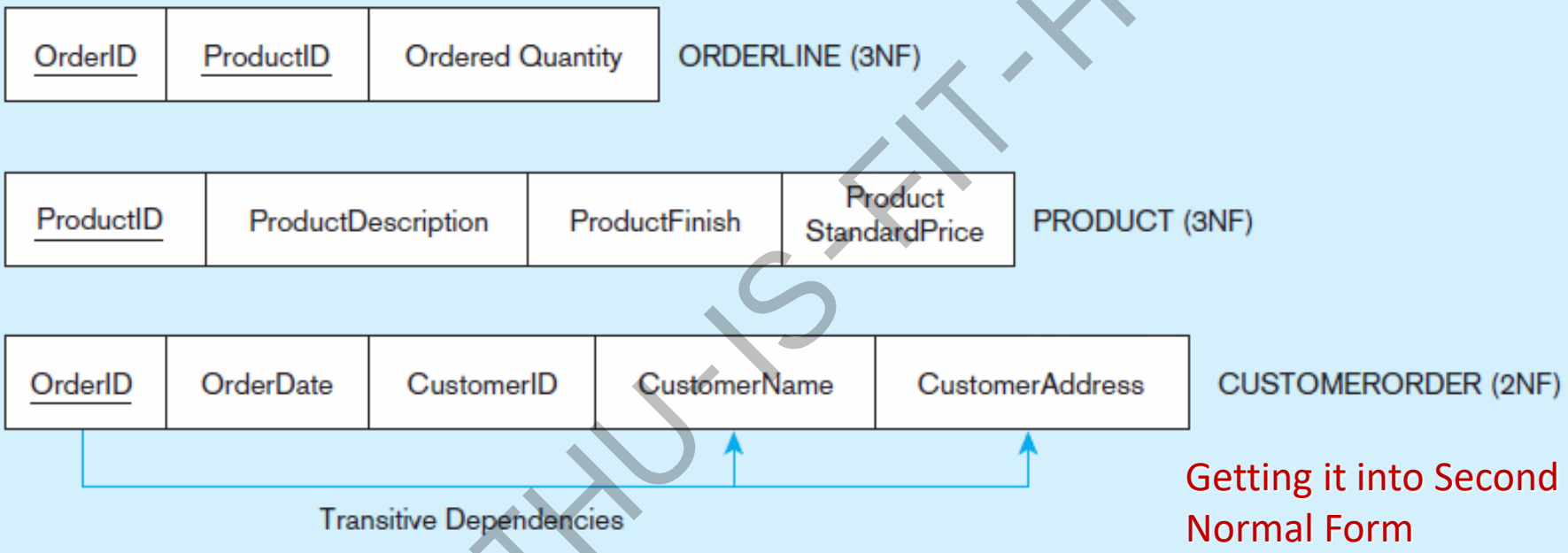
ProductID ➔ ProductDescription, ProductFinish, ProductStandardPrice

OrderID, ProductID ➔ OrderQuantity

Therefore, NOT in 2nd Normal Form

SECOND NORMAL FORM (2NF)

Removing partial dependencies



Partial dependencies are removed, but there are still transitive dependencies

THIRD NORMAL FORM (3NF)

A table is in third normal form (3NF) if it meets the following criteria:
1. The relation is in second normal form.
2. There are no transitive dependencies (functional dependencies on non-primary-key attributes):

Transitive dependency: An attribute functionally depends on another nonkey attribute
A →B and B → C and B **not→** A; therefore A → C

THIRD NORMAL FORM (3NF)

TRANSITIVE DEPENDENCE

Cho F = {MN→OPRX; NO →M; P →RY}

- P có phụ thuộc bắc cầu vào NO (NO → P)?

  NO → M ⇒ NO →MN : thỏa (i)

  MN → P:thỏa (ii)

  MN → O ⇒ MN →NO : không thỏa (iii)

  P không phụ thuộc bắc cầu vào NO

- R có phụ thuộc bắc cầu vào NO (NO → R)?

  NO → MN và MN→P ⇒ NO →P (i)

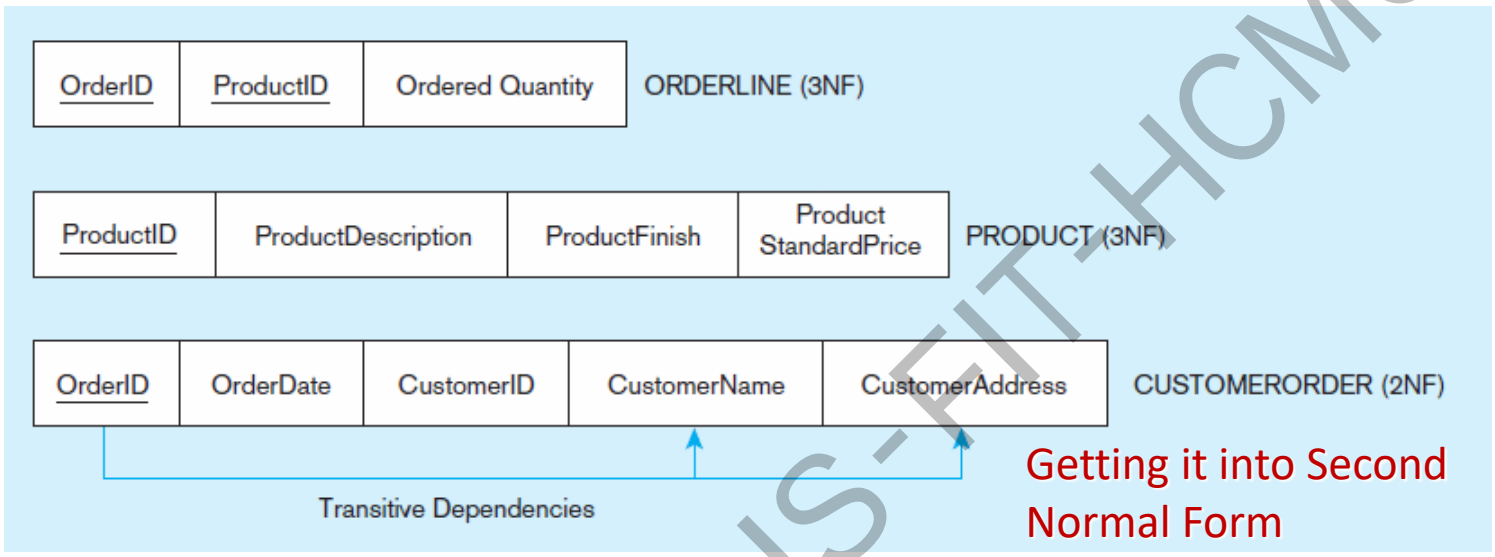
  P → R (ii)

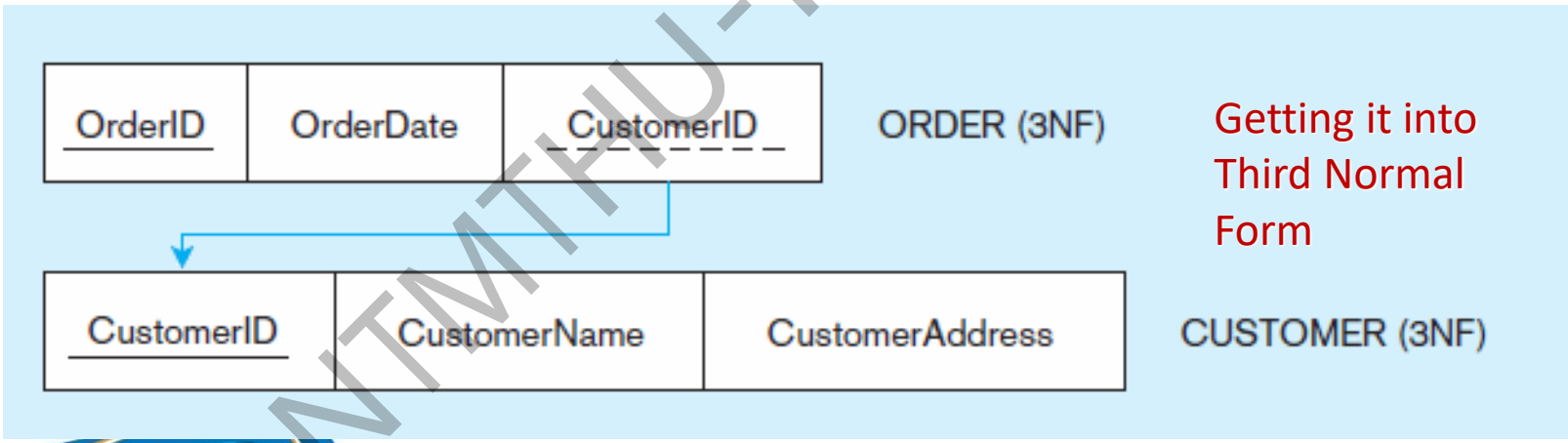
  P → NO ∉ F^+ (iii)

  R ∉ NOP (iv)

  R phụ thuộc bắc cầu vào NO

THIRD NORMAL FORM (3NF)





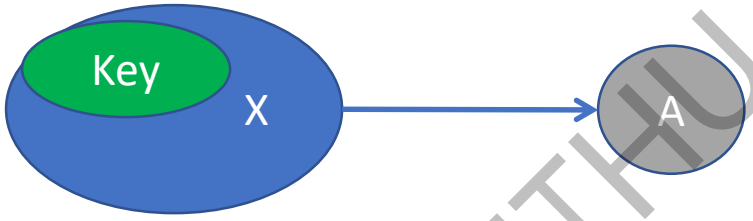
Transitive dependencies are removed

THIRD NORMAL FORM (3NF)

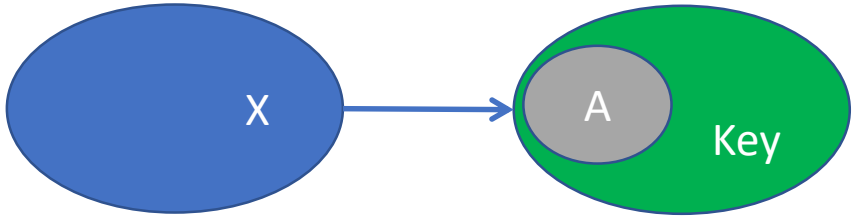
A table is in third normal form (3NF) if it meets the following criteria:

1. Let R be a relation scherna, F be the set of FDs given to hold over R, X be a subset of the attributes of R, and A be an attribute of R. R is in third normal form if, for every FD X → A in F, one of the following statements is true.

- **X is a superkey, or**
- **A is part of sorne key for R**



Case 1: X is a superkey

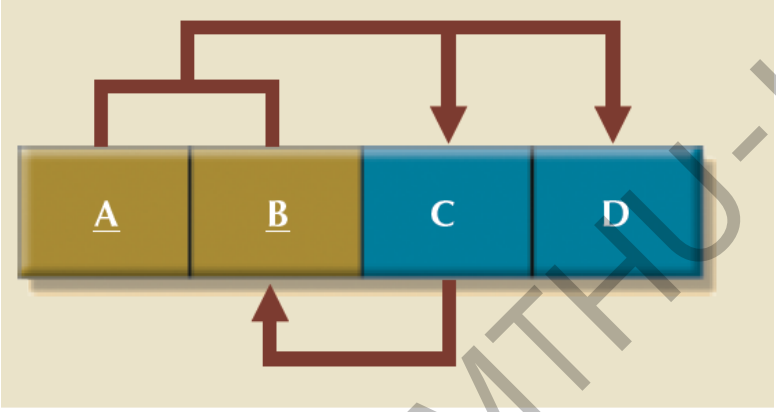


Case 2: A is part of sorne key for R

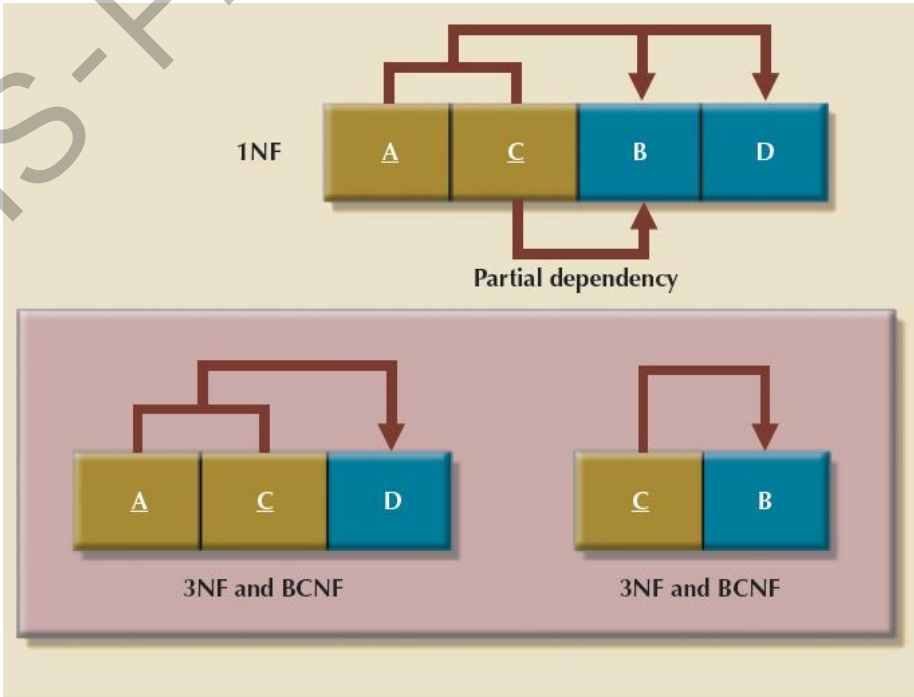
Boyce-Codd Normal Form (BCNF)

- A table is in Boyce-Codd Normal Form (BCNF) if it meets the following criteria:

1. The relation must be in third normal form.
2. All determinants must be candidate keys



In 3NF, but not in BCNF



Practice #1

- Cho quan hệ Hướng dẫn

Mã sinh viên	Chủ đề	Hướng dẫn viên
97001	Mạng truyền thông	Mr.Minh
97001	Hệ thống TT	Mr.Hà
97003	Cơ sở dữ liệu	Mr.Hiếu
97004	Mạng truyền thông	Mr.Nam
97005	Mạng truyền thông	Mr.Minh

Các qui tắc đặt trên quan hệ là:

- Mỗi sinh viên có thể theo một số chủ đề
- Mỗi chủ đề có thể có một số hướng dẫn viên
- Một hướng dẫn viên chỉ tư vấn cho một chủ đề
- Mỗi sinh viên cụ thể theo một chủ đề có một hướng dẫn viên cụ thể
- Một hướng dẫn viên có thể tư vấn một số sinh viên

Cấu trúc tương đương

Nguyên tắc tương đương:

Nếu C ’= {<Qi, Di>} (i = 1..n) được dùng để lưu trữ dữ liệu thay cho C = <U,D>, có nghĩa là:

- Thông tin và dữ liệu được xác định ở giai đoạn phân tích & thu thập nhu cầu phải được tìm thấy đầy đủ trong C’
- Thông tin và dữ liệu được lưu trong C’ là những TT&DL lẽ ra được lưu trong C

Mục tiêu chương

- Mục tiêu thiết kế quan niệm
- Dư thừa dữ liệu
- Tiêu chuẩn dạng chuẩn
- Tiêu chuẩn thiết kế tương đương
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
  - Biểu diễn trọn vẹn
- Chuẩn hóa: Tiếp cận phân rã

Cấu trúc tương đương

- Khái niệm TT được lưu trữ trong CSDL:
  - TT được lưu trữ trong Q: một bộ q của Q, là TT được lưu trong Q. q là tường minh nếu không chứa giá trị Trống (Null).
  - TT được lưu trữ trong C: TT lẽ ra được lưu trong U (hoặc trong C) được lưu thực tế trong các thể hiện của Q_i:
    - Cho TU, một thể hiện bất kỳ của U, $TQ_i = TU[Q_i^+]$
    - 2 điều kiện sau phải được thỏa:
      - $\forall u \in TU, u \in \bowtie_{i=1}^{n} TQ_i$
      - $\forall q \in \bowtie_{i=1}^{n} TQ_i, \ q \in TU$

Cấu trúc tương đương

S	P	D
s1	p1	d1
s2	p2	d2
s3	p1	d3

Instance r

S	P
s1	p1
s2	p2
s3	p1

$\pi_{SP}(r)$

P	D
p1	d1
p2	d2
p1	d3

$\pi_{PD}(r)$

S	P	D
s1	p1	d1
s2	p2	d2
s3	p1	d3
s1	p1	d3
s3	p1	d1

$\pi_{SP}(r) \bowtie \pi_{PD}(r)$

Cần có những **tiêu chí cụ thể** giúp người thiết kế xác định lược đồ CSDL QN có tương đương với lược quan hệ phổ quát không?

Không có ở thể hiện gốc

Cấu trúc tương đương

- Ba quan niệm về tính tương đương lược đồ CSDL:
  1) Tính chất bảo toàn phụ thuộc (dependence preservation)
  2) Tính chất bảo toàn thông tin (lossness-join)
  3) Biểu diễn trọn vẹn

Tính chất bảo toàn phụ thuộc

- Ưu tiên việc kiểm tra các phụ thuộc dữ liệu
- Quan niệm: các thông tin của CSDL được thể hiện thông qua phụ thuộc dữ liệu
- Đối với phụ thuộc hàm:

Cho $C_1 = <Q,F>$ và $C_2 = \{<Q_i, F_i>\}$, i=1..n, với $F_i = F^+[Q_i^+]$

(F_i là những pth của F được định nghĩa trên Q_i)

$C_1 \equiv C_2$ theo tính chất BTPT nếu thỏa hai điều kiện sau:

(i.1) $\cup_i Q_i^+ = Q^+$

(i.2) $(\cup_i F_i)^+ = F^+$ $(\cup_i F_i \equiv F)$

Kiểm tra bảo toàn PTH

Lược đồ 1: LỊCH_COI_THI (GVCT, N,G,P,M,GV)

F = { f1: GVCT → N,G,P : f2: M → GV ,f3:N,G,P → M}

Lược đồ 2:

<COI_THI (GVCT, N,G,P); F1 = {GVCT → N,G,P}>

<MON_CT(N,G,P,M,GV); F2 = {N,G,P→M; M →GV}>

Lược đồ 3:

<COI_THI (GVCT, N, G, P) F1 = {GVCT → N,G,P}>

<LỊCH_CT(N,G,P,M) F2 = {N,G,P→M}>

<GD(M,GV) F3 = {M →GV}>

Tính chất bảo toàn thông tin

$\underline{C_1} \equiv \underline{C_2}$ theo tính chất BTTT nếu thoả hai điều kiện sau:

(ii.1) $\cup_i Q_i^+ = Q^+$

(ii.2) $\bowtie_{i=1}^{n} Q[Q_i^+] = Q$, nghĩa là:

$\forall TQ \in Q, TQ = \bowtie_{i=1}^{n} TQ[Q_i^+]$

Tính chất bảo toàn thông tin

Phương tiện kiểm tra

- Cho: $C_1 = \langle Q, F \rangle$

  và $C_2 = \{\langle Q_i, F_i \rangle\}$, i=1..n, với $F_i = F^+[Q_i^+]$

- Bảng Tableau và qui trình thay thế đuổi, dựa vào *luật pth*, được sử dụng để kiểm tra tính bảo toàn thông tin của C_2.

Tính chất bảo toàn thông tin

Phương tiện kiểm tra:

- Bảng Tableau:
  - Có m cột tương ứng với m thuộc tính của quan hệ Q
  - n dòng tương tương ứng với n quan hệ con Q_i
  - Mỗi ô ở dòng i và cột j của bảng chứa một trong hai giá trị:
    - a_j nếu Q_i^+ có chứa thuộc tính tương ứng với cột j
    - b_k trong trường hợp ngược lại, với k bắt đầu từ 1 và tăng dần mỗi khi cần dùng đến giá trị b

Tính chất bảo toàn thông tin

Phương tiện kiểm tra:

- Luật phụ thuộc hàm áp dụng vào Tableau:

Với $X \rightarrow A \in F$:

Chọn hai dòng w_1 và w_2 trong Tableau sc $w_1.X = w_2.X$:

Nếu $w_1.A \neq w_2.A$ thì:

Nếu $w_1.A = a_j$ và $w_2.A = b_k$ thì thay thế b_k trong $w_2.A$ bằng a_j

Nếu $w_1.A = b_k$ và $w_2.A = a_j$ thì thay thế b_k trong $w_1.A$ bằng a_j

Nếu $w_1.A = b_k$ và $w_2.A = b_{k'}$ thì thay thế $b_{k'}$ trong $w_2.A$ bằng b_k

Cuối nếu

Tính chất bảo toàn thông tin

Phương tiện kiểm tra:

- Qui trình thay thế đuổi:

  1. $\forall f \in F$:

     Áp dụng luật pth cho f

     Cuối $\forall$

  2. Lặp lại bước 1 cho đến khi nào không còn thay đổi được giá trị nào trong bảng Tableau thì dừng.

Tính chất bảo toàn thông tin

Phương tiện kiểm tra:

- Gọi T' là bảng kết quả sau khi đã áp dụng qui trình thay thế đuổi trên bảng Tableau T
  - Nếu T' có một dòng chứa toàn các giá trị a_j thì $\underline{C_2}$ bảo toàn thông tin.
  - Ngược lại thì $\underline{C_2}$ không bảo toàn thông tin.

Tính chất bảo toàn thông tin

- Ví dụ: C_1= LỊCH_COI_THI (GVCT, N,G,P,M,GV)

  F = { f1: GVCT → N,G,P : f2: M → GV ,f3:N,G,P → M}

- Xét cấu trúc sau có tương đương với cấu trúc C_1:

  C_2 = { <COI_THI (GVCT, N, G, P), F1 = {f1:GVCT→N,G,P}>;
  <GD(M,GV); F2 = {f2 :M→GV}>;
  <CT_MON(GVCT, M); F3 = {f'1: GVCT→M}>}

	GV_CT	N	G	P	M	GV
COI_THI	a_1	a_2	a_3	a_4	b_1	b_2
GD	b_3	b_4	b_5	b_6	a_5	a_6
CT_MON	a_1	b_7	b_8	b_9	a_5	b_{10}

Áp dụng f_1

Áp dụng f_2

Biểu diễn trọn vẹn

- Cấu trúc CSDL $\underline{C_2}$ là một biểu diễn trọn vẹn của $\underline{C_1}$ nếu 3 điều kiện (i.1), (i.2) và (ii.2) đều được thỏa.
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
- $\underline{C_2}$ là biểu diễn trọn vẹn ≡ $\underline{C_1}$ vừa bảo toàn thông tin, vừa tạo thuận lợi cho việc kiểm tra PTH

Mục tiêu chương

- Mục tiêu thiết kế quan niệm
- Dư thừa dữ liệu
- Tiêu chuẩn dạng chuẩn
- Tiêu chuẩn thiết kế tương đương
  - Bảo toàn phụ thuộc hàm
  - Bảo toàn thông tin
  - Biểu diễn trọn vẹn
- Chuẩn hóa: Tiếp cận phân rã

Tiếp cận thiết kế quan niệm

- Có 2 cách tiếp cận:
  1) Cách tiếp cận phân rã
     - Thường được áp dụng để tinh chỉnh một lược đồ CSDL
  2) Cách tiếp cận tổng hợp
     - Thường được áp dụng để kiểm nghiệm một lược đồ CSDL

Cách tiếp cận phân rã

Cơ sở lý thuyết:

- Đối với phụ thuộc hàm:
  - Định lý 1: Cho <Q, F> $Q = Q_1 \bowtie Q_2$ không mất thông tin (* nghĩa là bttt: thoả (ii.1) & (ii.2) *)

    nếu và chỉ nếu:

    $$((Q_1^+ \cap Q_2^+) \rightarrow (Q_1^+ - Q_2^+)) \in F^+$$

    hoặc $((Q_1^+ \cap Q_2^+) \rightarrow (Q_2^+ - Q_1^+)) \in F^+$
  - Vận dụng định lý 1: $\forall f: X \rightarrow A \in F$, Q được phân rã thành $Q_1(XA)$ và $Q_2(Q^+ - A)$

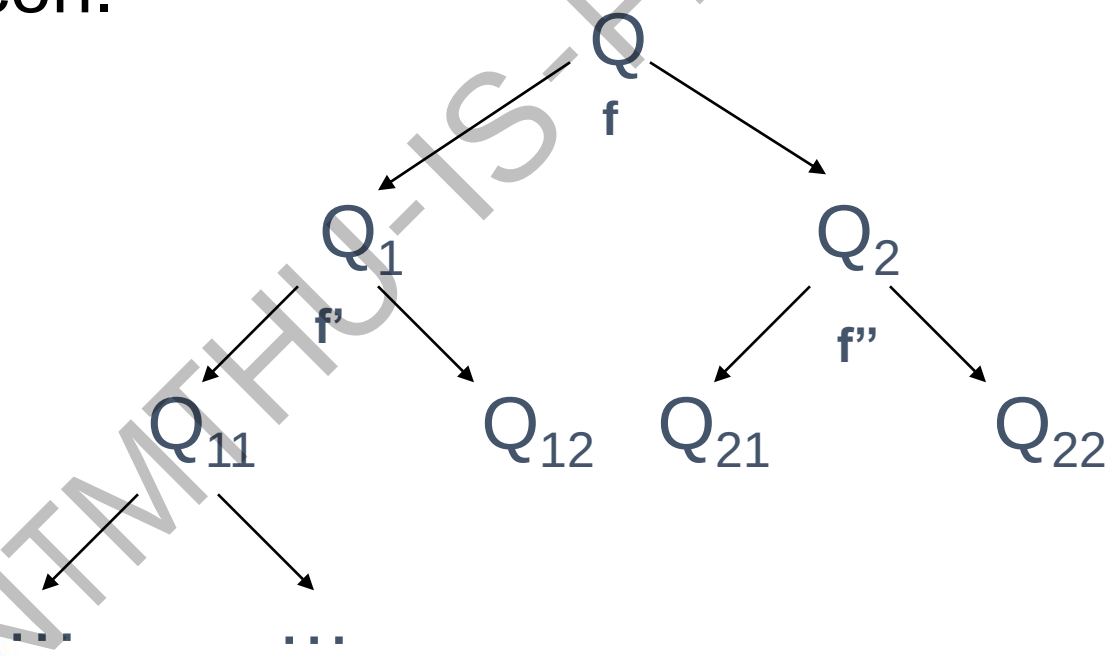
Cách tiếp cận phân rã

- Ví dụ: Cho lược đồ R(ABC) và F = {A $\rightarrow$ B, C $\rightarrow$ B} được phân rã thành 2 lược đồ R1(AB) và R2 (CB).

- Đây là một phân rã không bảo toàn thông tin, bởi vì:
  - $((Q_1^+ \cap Q_2^+) \rightarrow (Q_1^+ - Q_2^+))$: $B \rightarrow A \notin F^+$
  - $((Q_1^+ \cap Q_2^+) \rightarrow (Q_2^+ - Q_1^+))$: $B \rightarrow C \notin F^+$

Cách tiếp cận phân rã

- Ứng dụng định lý 1, từ một quan hệ cần phân rã, ta sẽ có cây phân rã nhị phân, trong đó, ở mỗi cấp, một quan hệ được tách thành hai quan hệ con.



Thuật toán phân rã

Thủ tục Phân_rã (Q)

- Xác định tập $F_Q = \{f: X \rightarrow Y \in D^+ \mid (XY) \subseteq Q^+\}$
- Nếu: $\forall (f: X \rightarrow Y \in F_Q), (XY) = Q^+$

  thì: $\rho := \rho \cup \{Q\}$

  nếu không:
  - Chọn $(f: X \rightarrow Y) \in D_Q$ với $(XY) \subset Q^+$ để phân rã (*)
  - Q1 = Q[XY] ; Q2 = Q[Q^+ -Y]
  - Phân_rã (Q1)
  - Phân_rã (Q2)

Cuối thủ tục Phân_rã *(*): xem nhận xét*

Thuật toán phân rã

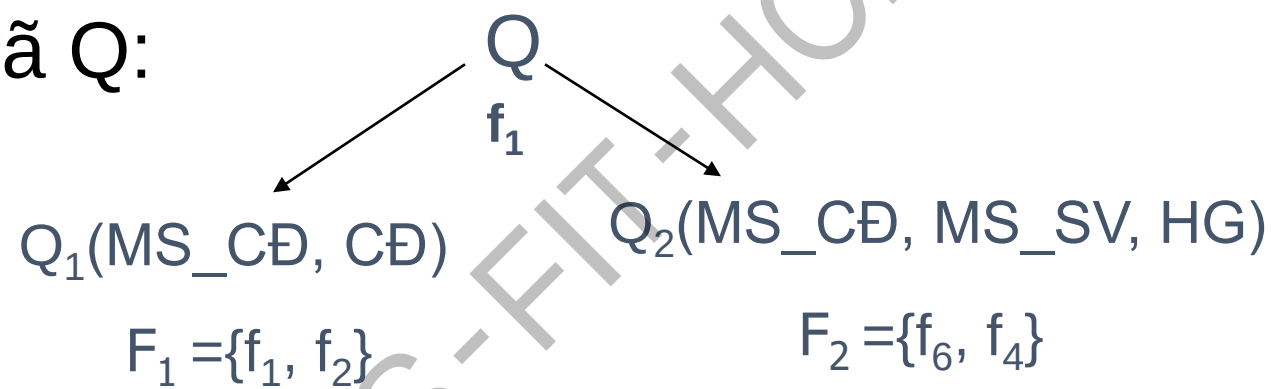
- Ví dụ:

  Cho quan hệ : Q(MS_CĐ, MSSV, CĐ, HG)

  D = { f1: MS_CĐ → CĐ;
  f2: CĐ →MS_CĐ;
  f3: CĐ, MS_SV → HG;
  f4: MS_CĐ, HG →MS_SV;
  f5: CĐ, HG →MS_SV;
  f6: MS_CĐ, MS_SV →HG;
  }

Thuật toán phân rã

- Phân rã Q:

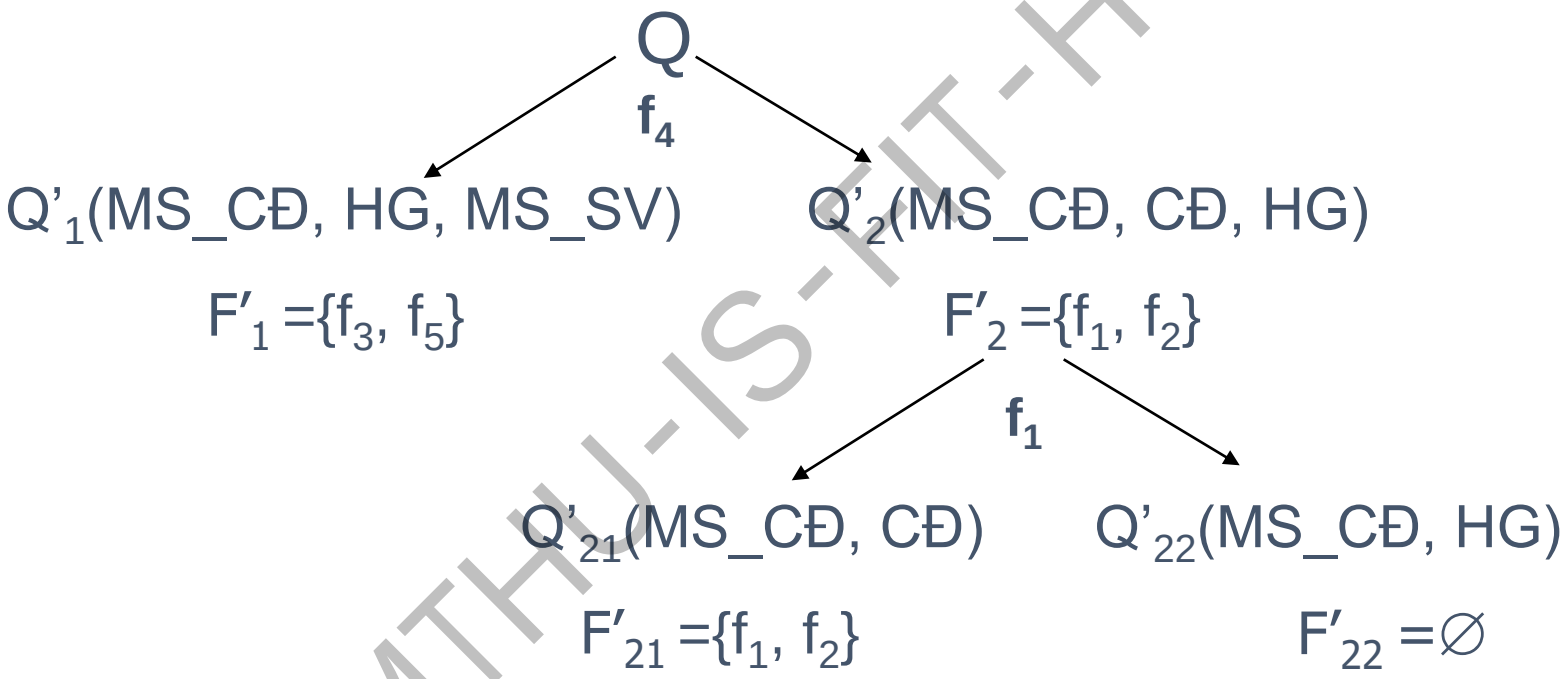


- C1 = { <Q1(MS_CĐ, CĐ), F_1>;
  <Q2(MS_CĐ, MS_SV, HG), F_2}

- C1 có bảo toàn phụ thuộc?

Thuật toán phân rã

- Phân rã Q với D có thứ tự khác:



- C2 = { < Q'1(MS_CĐ, HG, MS_SV), F'1>;
  <Q'21(MS_CĐ, CĐ), F'21>;
  < Q'22(MS_CĐ, HG), F'22> }

Thuật toán phân rã

Nhận xét:

- Tất cả các quan hệ trong lược đồ kết quả đều đạt DC BCK (nếu chỉ có pth).
- Phân rã bảo toàn thông tin (do áp dụng định lý 1), nhưng không đảm bảo bảo toàn pth.
- Thuật toán không quan tâm đến "chất lượng" của tập phụ thuộc F ban đầu: F có thể chứa các pth không đầy đủ, pth dư thừa.
- Thuật toán không quan tâm đến việc xác định khóa của các quan hệ con.

Thuật toán phân rã

- Tùy vào cách chọn phụ thuộc để phân rã (bước 3) mà cấu trúc kết quả sẽ khác nhau:
  - Trong ví dụ: Chọn theo thứ tự từ trên xuống: kết quả khác nhau tùy theo thứ tự của F, và hậu quả là trong lược đồ kết quả có thể chứa những quan hệ không có ý nghĩa (ví dụ Q'22)
  - Có một số nghiên cứu để cải tiến thuật toán phân rã: xây dựng chiến lược chọn phụ thuộc để phân rã hiệu quả hơn (bảo toàn pth,…)

Practice #02

Cho Q(CSJQPDV)

F= { f1: C→SJDPQV

f2: JP → C

f3: SD → P

f4: J →S}

a) Hãy phân rã Q thành Qi với SD → P được lựa chọn đầu tiên?

b) Hãy phân rã Q thành Qi với J → S được lựa chọn đầu tiên?