

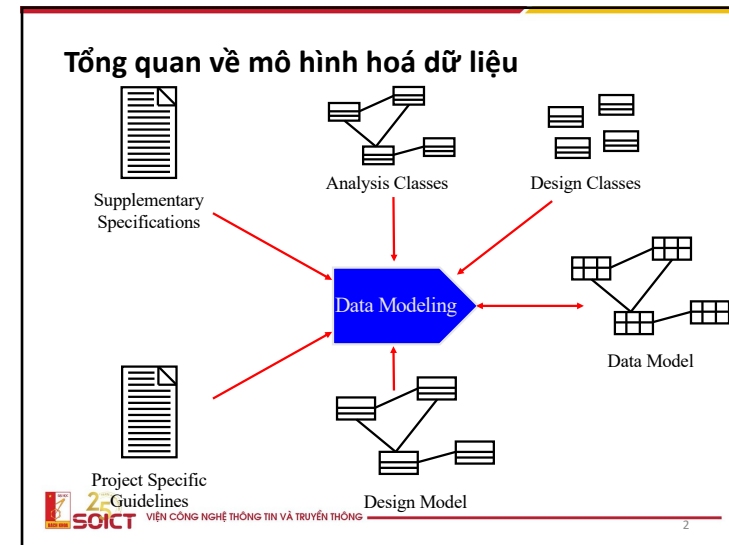
ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

# IT4490 – Thiết kế và xây dựng phần mềm

## Bài 7. Mô hình hoá dữ liệu

1



2

Nội dung

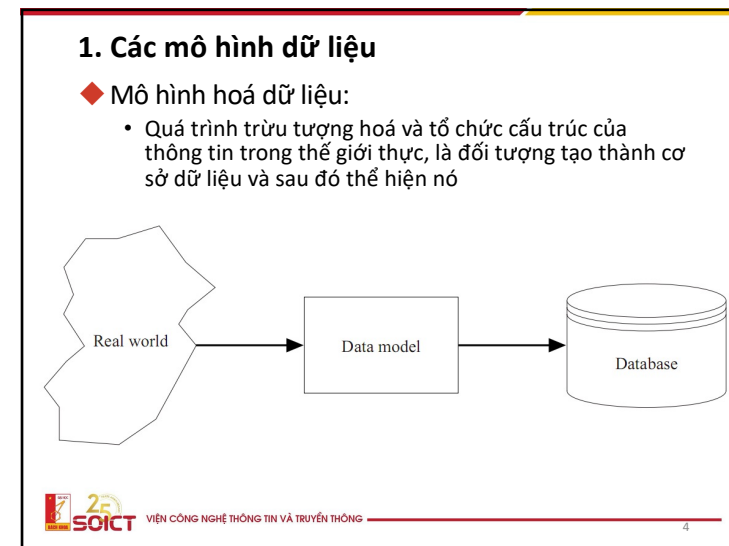
1. Các mô hình dữ liệu

2. Mô hình đối tượng và mô hình dữ liệu quan hệ

3. Ánh xạ sơ đồ lớp sang sơ đồ E-R

4. Chuẩn hoá

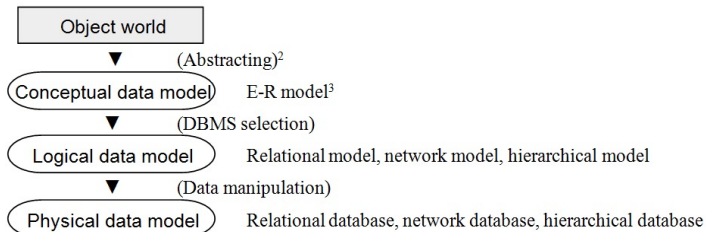
3



4

## 1. Các mô hình dữ liệu (2)

❖ 3 loại mô hình dữ liệu:



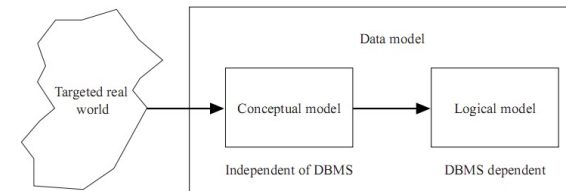
5

## 1.1. Mô hình dữ liệu khái niệm

❖ Những biểu diễn tự nhiên không có các ràng buộc do DBMS áp đặt

❖ Mô hình E-R

- Biểu diễn bằng sơ đồ E-R

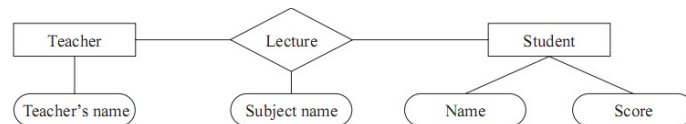


6

## Biểu đồ E-R

❖ Ba thành phần

- Các thực thể (Entities)
- Các mối quan hệ (Relationships)
- Các thuộc tính (Attributes)



7

## 1.2. Mô hình dữ liệu logic

❖ 3 loại

- Mô hình quan hệ (relational model),
- Mô hình mạng (network model),
- Và mô hình phân cấp (hierarchical model)

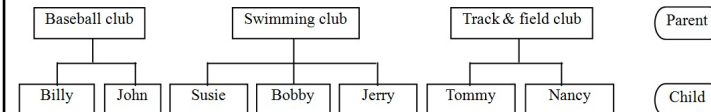
8

### 1.3. Mô hình dữ liệu vật lý

- ❖ Các mô hình dữ liệu logic, khi chúng được triển khai, sẽ trở thành mô hình dữ liệu vật lý:
  - Các cơ sở dữ liệu quan hệ (relational databases),
  - Các cơ sở dữ liệu mạng (network databases),
  - Hoặc các cơ sở dữ liệu phân cấp (hierarchical databases)

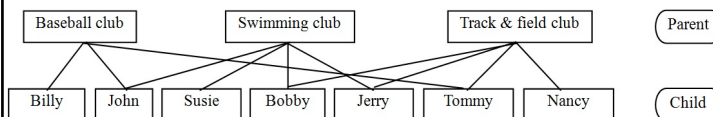
#### 1.3.1. Cơ sở dữ liệu phân cấp (Cơ sở dữ liệu cấu trúc cây)

- ❖ Chia các bản ghi thành cha và con và hiển thị mối quan hệ với cấu trúc phân cấp
- ❖ Tương ứng 1-nhiều (1: n) giữa bản ghi cha và bản ghi con



#### 1.3.2. Cơ sở dữ liệu mạng

- ❖ Bản ghi cha và bản ghi con không có tương ứng 1 - nhiều (1: n); đúng hơn, chúng ở dạng tương ứng nhiều-nhiều (m: n)
- ❖ Đôi khi được gọi là cơ sở dữ liệu CODASYL



#### 1.3.3. Cơ sở dữ liệu quan hệ

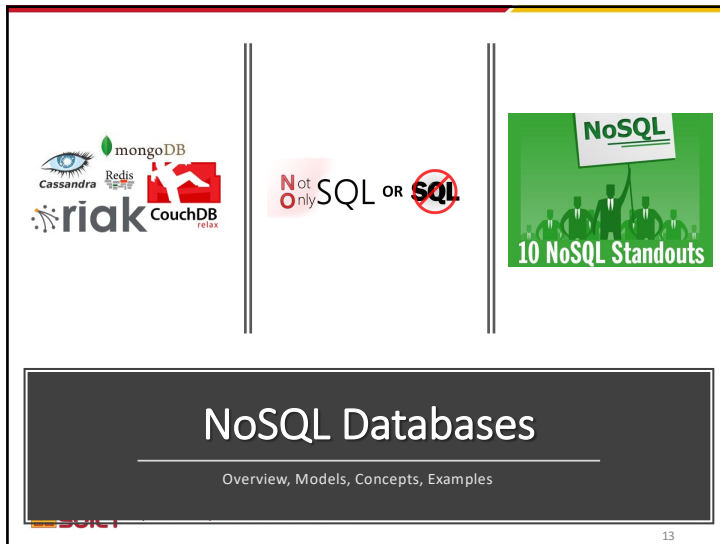
- ❖ Dữ liệu được thể hiện trong một bảng hai chiều.
  - Mỗi hàng của bảng tương ứng với một bản ghi và mỗi cột là một mục của các bản ghi.
  - Các cột được gạch chân thể hiện khóa chính

Name of the table: Employee\_tbl

Columns (items, attributes,):

<u>Employee_number</u>	Name	<u>Tel_number</u>
00100	Paul Smith	03-3456-0001
00200	Rick Martin	03-3456-0011
00300	Billy Graham	03-3456-0010
00400	John Wilson	03-3456-0200

← Row (pair, tuple, record)



Logo logos: mongoDB, Cassandra, Redis, riak, CouchDB, relax.

Not only SQL OR SQL

NoSQL

10 NoSQL Standouts

## NoSQL Databases

Overview, Models, Concepts, Examples

13

13

### Cơ sở dữ liệu NoSQL là gì?

- ❖ Cơ sở dữ liệu NoSQL (đám mây)
  - Sử dụng mô hình dựa trên tài liệu (không quan hệ: non-relational)
  - Lưu trữ tài liệu không có giản đồ (schema-free)
    - Vẫn hỗ trợ lập chỉ mục và truy vấn
    - Vẫn hỗ trợ các hoạt động CRUD (tạo, đọc, cập nhật, xóa)
    - Vẫn hỗ trợ xử lý đồng thời và giao dịch
  - Được tối ưu hóa cao cho việc nối / truy xuất
  - Hiệu suất tuyệt vời và có khả năng mở rộng
  - NoSQL == “Không có SQL” hoặc “Không chỉ SQL”?

SOICT VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

14

14

### Cơ sở dữ liệu Quan hệ và NoSQL

- ❖ Cơ sở dữ liệu quan hệ
  - Dữ liệu được lưu trữ dưới dạng các hàng trong bảng
  - Mối quan hệ giữa các hàng có liên quan
  - Một thực thể đơn có thể kéo dài nhiều bảng
  - Các hệ thống RDBMS rất trưởng thành, vững chắc
- ❖ Cơ sở dữ liệu NoSQL
  - Dữ liệu được lưu trữ dưới dạng tài liệu
  - Thực thể đơn (tài liệu) là một bản ghi duy nhất
  - Tài liệu không có cấu trúc cố định

SOICT VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

15

15

### Mô hình Quan hệ và NoSQL

#### Relational Model

Name	Svetlin Nakov
Gender	male
Phone	+359333777555
Email	nakov@abv.bg
Site	www.nakov.com

Street	Al. Malinov 31
Post Code	1729

Town	Sofia
------	-------

Country	Bulgaria
---------	----------

#### Document Model

```

Name: Svetlin Nakov
Gender: male
Phone: 359333777555
Address:
- Street: Al. Malinov 31
- Post Code: 1729
- Town: Sofia
- Country: Bulgaria
Email: nakov@abv.bg
Site: www.nakov.com

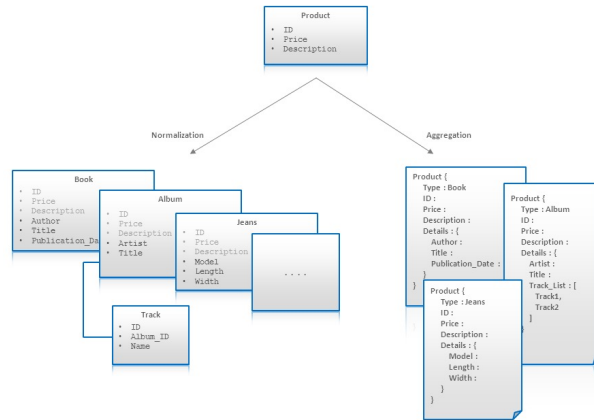
```

SOICT VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

16

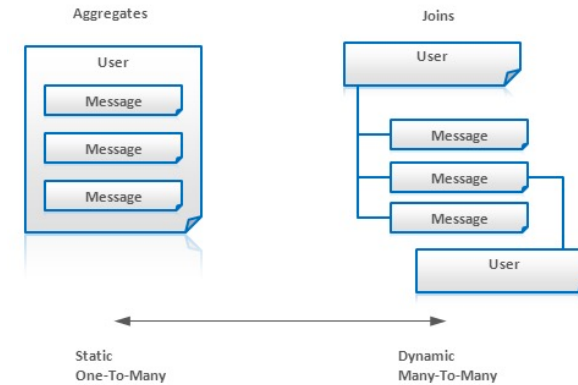
16

## Chuẩn hoá (Normalization) vs. Tập hợp (Aggregation)



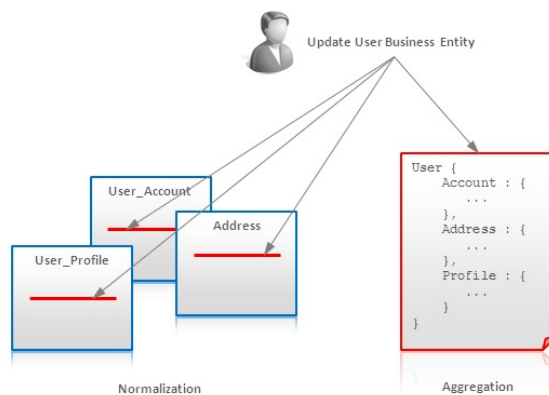
17

## Kết hợp (Aggregates) vs. Kết nối (Joins)



18

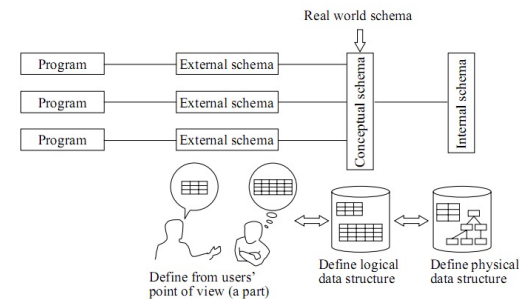
## Atomic Aggregates



19

## 1.3.4. Lược đồ ba lớp (3-layer schema)

- ❖ Một lược đồ (schema) là một mô tả về khung (framework) của một cơ sở dữ liệu
- ❖ Được phân thành 3 loại:



20

## Nội dung

1. Các mô hình dữ liệu
- ➔ 2. Mô hình đối tượng và mô hình dữ liệu quan hệ
3. Ánh xạ sơ đồ lớp sang sơ đồ E-R
4. Chuẩn hoá

21

## 2.1. Cơ sở dữ liệu quan hệ và HĐT (OO)

### ❖ RDBMS và Hướng đối tượng không hoàn toàn tương thích

#### ▪ RDBMS

- Trọng tâm là dữ liệu
- Phù hợp hơn cho các mối quan hệ đặc biệt và ứng dụng báo cáo
- Bộc lộ dữ liệu (các giá trị cột)

#### ▪ Hệ thống hướng đối tượng (Object Oriented system)

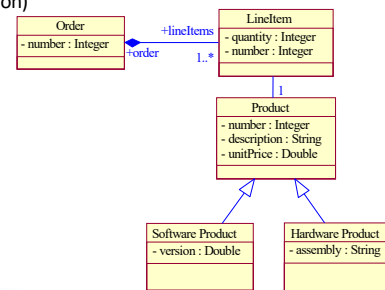
- Tập trung vào hành vi
- Phù hợp hơn để xử lý hành vi cho trạng thái xác định nơi dữ liệu là thứ yếu
- Che giấu dữ liệu (đóng gói)

22

## 2.2. Mô hình đối tượng (Object Model)

### ❖ Mô hình Đối tượng bao gồm

- Các lớp (Classes) (các thuộc tính - attributes)
- Các mối quan hệ (Relationships)
- Liên kết (Associations)
- Tổng quát hoá (Generalization)



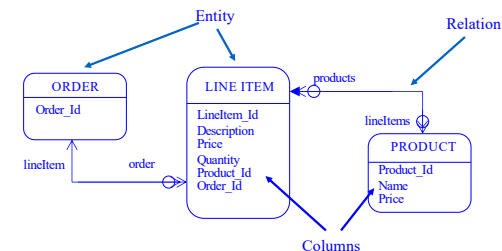
23

## 2.3. Mô hình dữ liệu quan hệ

### • Mô hình Dữ liệu quan hệ bao gồm

- Các thực thể (Entities) – Bảng (Table)
- Các liên kết (Relations) – Quan hệ (Relationship)

➔ Cũng được gọi là mô hình E-R



24

### 2.3.1. Thực thể/Bảng (Entities/Tables)

❖ Các thực thể được ánh xạ tới bảng khi thiết kế cơ sở dữ liệu vật lý

❖ Bao gồm

- Các cột: Thuộc tính
- Các hàng: Giá trị cụ thể của các thuộc tính

Columns

courseID	description	startDate	endDate	location
2008.11.001	This course...	12 Nov 2008	30 Nov 2008	D3-405
2008.11.002	This course...	22 Nov 2008	10 Dec 2008	T-403

Rows

### 2.3.2. Quan hệ (Relations/Relationships)

- Liên kết giữa các thực thể hoặc mối quan hệ giữa các bảng
- Bội số (Multiplicity/Cardinality)

- One-to-one (1:1)
- One-to-many (1:m)
- Many-to-one (m:1)
- Many-to-many (m:n)

(Thông thường, liên kết many-to-many được tách thành các liên kết one-to-many và many-to-one)

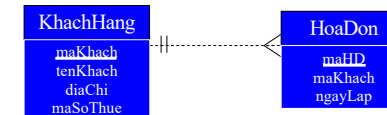
### Các mối quan hệ phụ thuộc

- ❖ Thực thể con chỉ có thể tồn tại khi thực thể cha tồn tại
- ❖ Thực thể con có khóa ngoại tham chiếu đến khóa chính của thực thể cha
- ❖ Khóa ngoại này được bao gồm trong khóa chính của con
- ❖ Biểu diễn bằng đường nét liền



### Các mối quan hệ độc lập

- ❖ Thực thể con có thể tồn tại ngay cả khi thực thể cha không tồn tại
- ❖ Thực thể con có khóa ngoại tham chiếu đến khóa chính của thực thể cha
- ❖ Khóa ngoại này không có trong khóa chính của con
- ❖ Biểu diễn bằng đường nét đứt

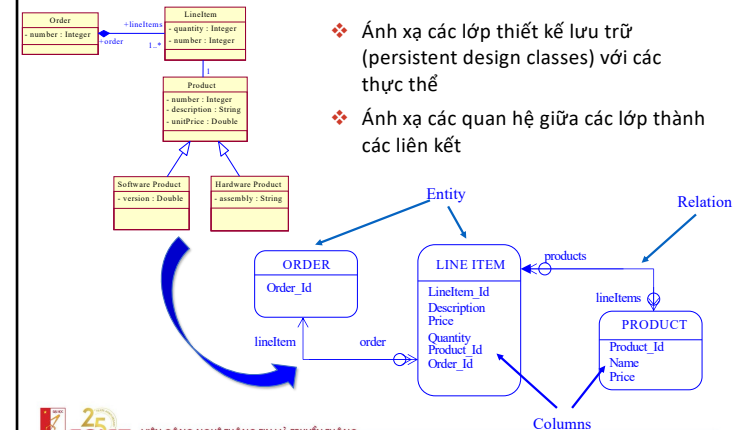


## Nội dung

1. Các mô hình dữ liệu
2. Mô hình đối tượng và mô hình dữ liệu quan hệ
- ⇒ 3. Ánh xạ sơ đồ lớp sang sơ đồ E-R
4. Chuẩn hoá

29

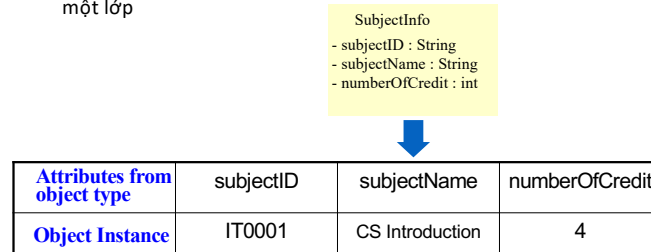
## 3. Ánh xạ sơ đồ lớp sang sơ đồ E-R



30

### 3.1. Ánh xạ các lớp thiết kế lưu trữ với các thực thể

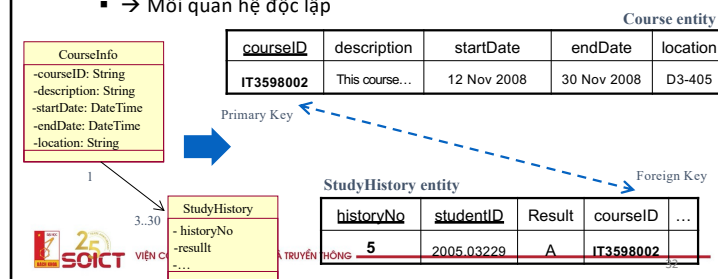
- ❖ Trong một cơ sở dữ liệu quan hệ
  - Mỗi hàng trong bảng được coi là một đối tượng
  - Một cột trong bảng tương đương với một thuộc tính liên tục của một lớp



31

### 3.2. Ánh xạ các liên kết (associations) giữa các Persistent Objects

- ❖ Các liên kết giữa hai đối tượng lưu trữ được thể hiện dưới dạng khóa ngoại cho các đối tượng liên kết.
  - Một khóa ngoại (foreign key) (không nằm trong khóa chính) là một cột trong một bảng có chứa giá trị khóa chính của đối tượng được liên kết
  - → Mỗi quan hệ độc lập

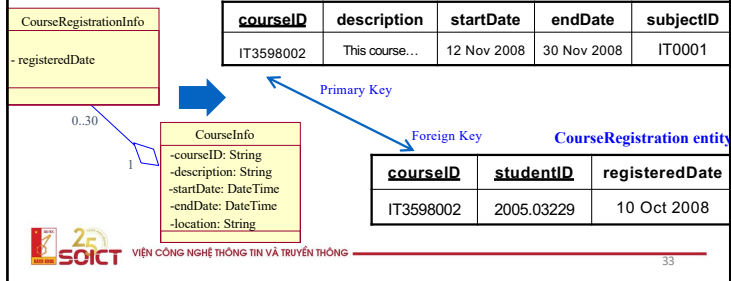


32



### 3.3. Ánh xạ kết tập (aggregation) vào mô hình dữ liệu

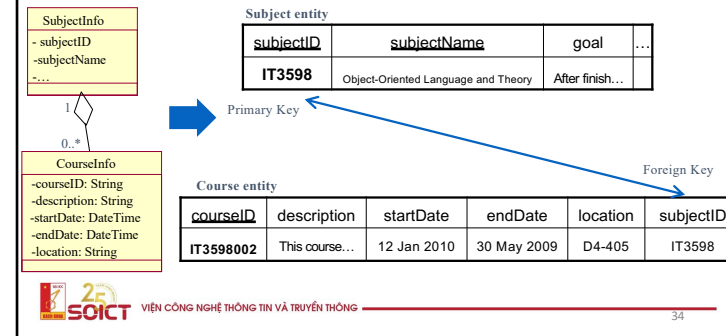
- ❖ Kết tập cũng được mô hình hóa thành mối quan hệ phụ thuộc bằng cách sử dụng quan hệ khóa ngoại
  - Sự phụ thuộc được thực hiện bằng một ràng buộc xóa theo tầng (cascading delete constraint)



33

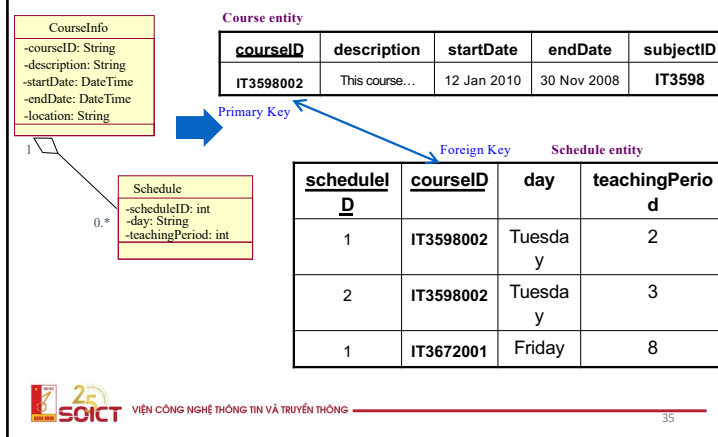
### 3.3. Ánh xạ kết tập (aggregation) vào mô hình dữ liệu (2)

- ❖ Trong một số trường hợp, chúng ta có thể ánh xạ đến mối quan hệ độc lập để đơn giản hóa khóa chính.
- ❖ Ví dụ: CourseID là khóa chính (dựa theo yêu cầu)



34

### Các ví dụ khác trong Course Registration

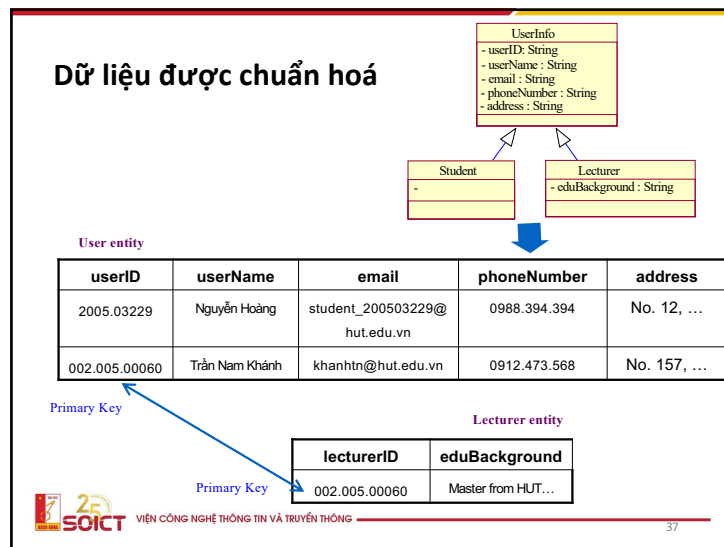


35

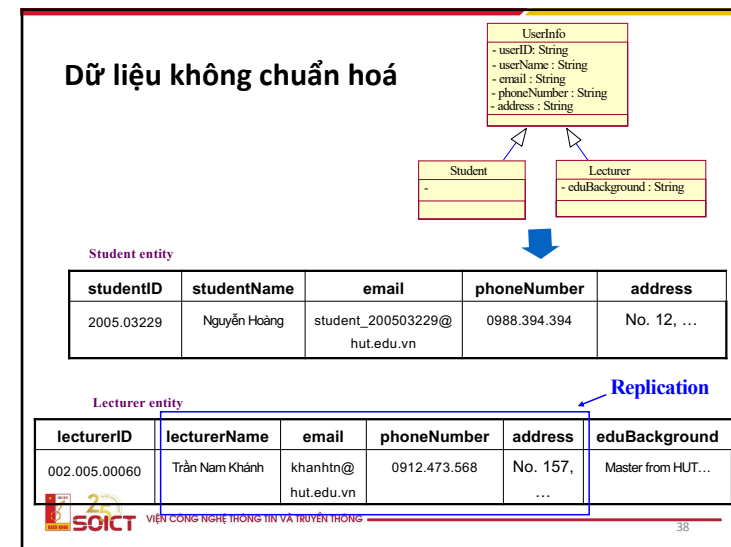
### 3.4. Mô hình hoá kế thừa (inheritance) trong mô hình dữ liệu

- ❖ Mô hình dữ liệu không hỗ trợ kế thừa mô hình hóa theo cách trực tiếp
- ❖ Hai lựa chọn:
  - Sử dụng các bảng riêng biệt (dữ liệu chuẩn hóa)
  - Tạo bản sao tất cả các liên kết và thuộc tính được kế thừa (dữ liệu không chuẩn hóa)

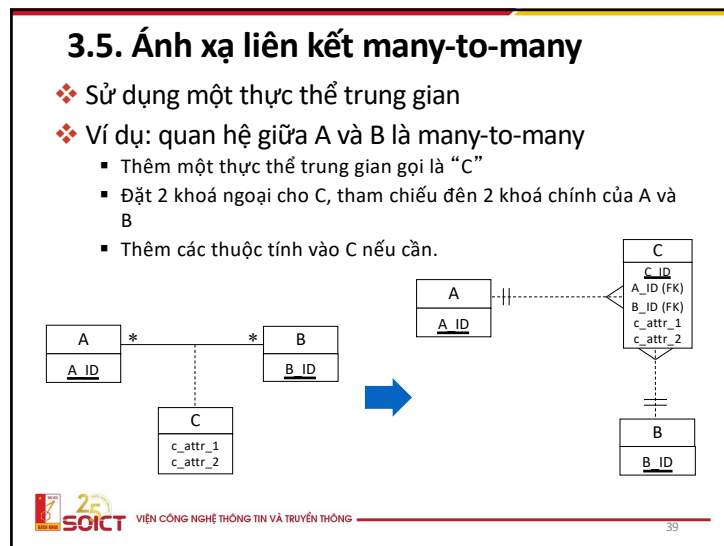
36



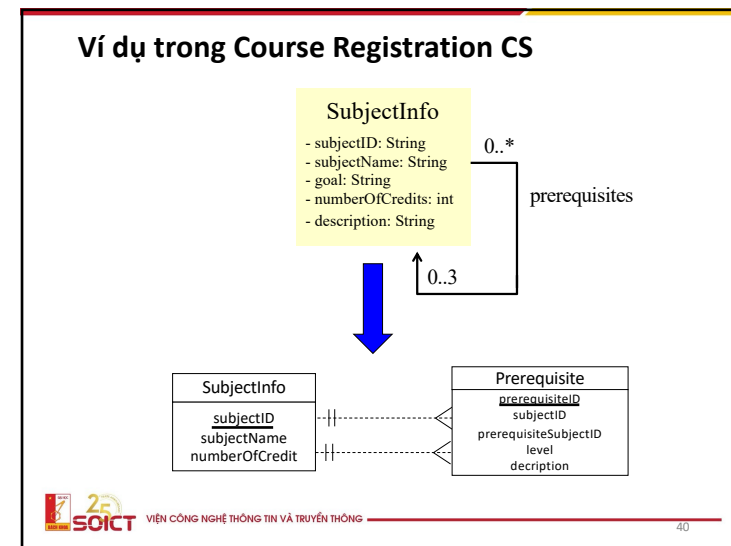
37



38

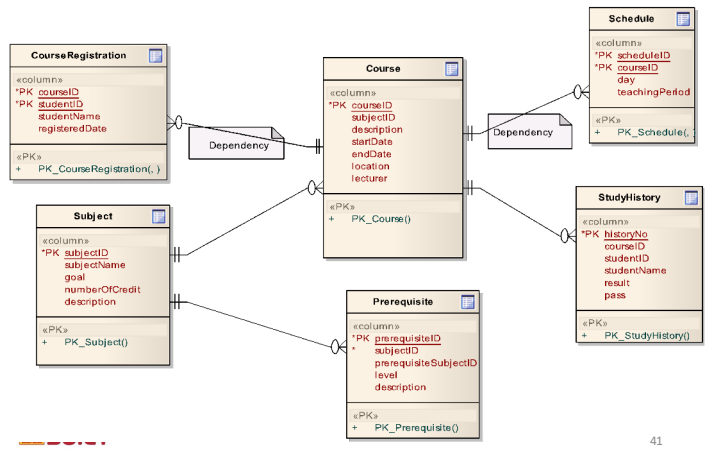


39



40

## Biểu đồ E-R (E-R diagram)



41

## Nội dung

1. Các mô hình dữ liệu
2. Mô hình đối tượng và mô hình dữ liệu quan hệ
3. Ánh xạ sơ đồ lớp sang sơ đồ E-R
- ⇒ 4. Chuẩn hoá

42

42

### 4.1. Tổng quan về chuẩn hoá

- ❖ Chuẩn hoá (Normalization): quy trình các bước sẽ xác định và loại bỏ các dư thừa trong thiết kế cơ sở dữ liệu.
- ❖ Mục đích của chuẩn hoá: để cải thiện
  - hiệu quả lưu trữ
  - toàn vẹn dữ liệu
  - và khả năng mở rộng

43

43

### 4.1. Tổng quan về chuẩn hoá (2)

- ❖ Trong mô hình quan hệ, tồn tại các phương pháp để định lượng mức độ hiệu quả của cơ sở dữ liệu.
- ❖ Các phân loại này được gọi là **các dạng chuẩn (normal forms hoặc NF)**, và có các thuật toán để chuyển đổi một cơ sở dữ liệu đã cho giữa chúng.
- ❖ Chuẩn hóa thường liên quan đến việc tách các bảng hiện có thành nhiều bảng, các bảng này phải được nối lại hoặc liên kết mỗi khi truy vấn được đưa ra.

44

44

## 4.2. Lịch sử



- ❖ Edgar F. Codd lần đầu tiên đề xuất quá trình chuẩn hóa và cái được gọi là dạng chuẩn 1 trong bài báo của ông ***A Relational Model of Data for Large Shared Data Banks*** Codd phát biểu rằng:

*“There is, in fact, a very simple elimination procedure which we shall call normalization. Through decomposition nonsimple domains are replaced by ‘domains whose elements are atomic (nondecomposable) values’.”*

## 4.3. Các dạng chuẩn (Normal Forms)

- ❖ Edgar F. Codd ban đầu thiết lập ba dạng chuẩn: 1NF, 2NF và 3NF.
- ❖ Hiện nay có một số dạng chuẩn khác thường được chấp nhận, nhưng 3NF được coi là đủ cho hầu hết các ứng dụng.
- ❖ Hầu hết các bảng khi đạt 3NF cũng đều ở dạng BCNF (Boyce-Codd Normal Form).



## Functionally determines

- ❖ Trong một bảng, một tập các cột X, quyết định hàm (functionally determines) một cột Y khác...

$X \rightarrow Y$

... khi và chỉ khi mỗi giá trị của X được liên kết với nhiều nhất một giá trị của Y trong bảng.

- ví dụ. Nếu bạn biết X thì khi đó chỉ có một khả năng cho Y.

## Các dạng chuẩn phổ biến...

- ◆ Dạng chuẩn thứ nhất (First normal form)
  - Tất cả các giá trị dữ liệu là nguyên tử và vì vậy mọi thứ đều phù hợp với một quan hệ toán học.
- ◆ Dạng chuẩn thứ hai (Second normal form)
  - Đã ở dạng chuẩn 1NF thêm với không có thuộc tính không phải khóa chính phụ thuộc bắc cầu vào khóa chính
- ◆ Dạng chuẩn thứ ba (Third normal form)
  - Đã ở dạng chuẩn 2NF thêm với không có thuộc tính không phải khóa chính phụ thuộc bắc cầu vào khóa chính

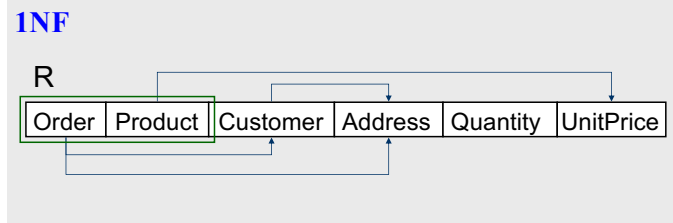
## Ví dụ về dạng chuẩn

- ◆ Hãy xem xét một bảng đại diện cho các đơn đặt hàng trong một cửa hàng trực tuyến
- ◆ Mỗi mục trong bảng đại diện cho một mục trên một đơn đặt hàng cụ thể.
- ◆ Các cột
  - Order
  - Product
  - Customer
  - Address
  - Quantity
  - UnitPrice
- ◆ Khoá chính là {Order, Product}

## Các phụ thuộc hàm

- Mỗi đơn hàng dành cho **một** khách hàng {Order} → {Customer} hàng
- Mỗi khách hàng có **một** địa chỉ duy nhất {Customer} → {Address} nhất
- Mỗi sản phẩm có **một** giá duy nhất {Product} → {UnitPrice} nhất
- FD's 1 và 2 có tính bắc cầu {Order} → {Address}

## Ví dụ – Biểu đồ phụ thuộc hàm (FD Diagram)

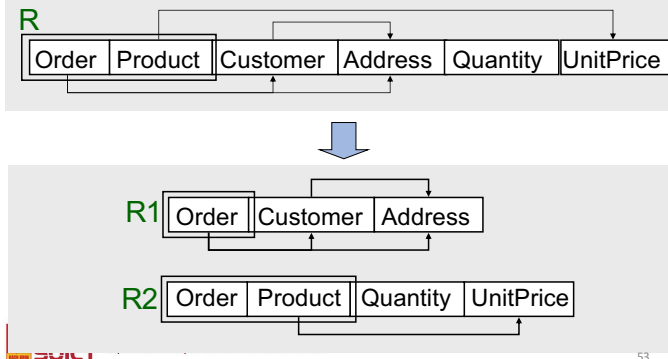


## Chuẩn hoá về 2NF

- ◆ Hãy nhớ dạng chuẩn thứ 2 có nghĩa là không có phụ thuộc một phần vào khóa. Nhưng chúng ta có:
  - {Order} → {Customer, Address}
  - {Product} → {UnitPrice}
- Và một khoá chính: {Order, Product}
- Vì vậy, để loại bỏ FD đầu tiên, chúng ta *tách*:
  - {Order, Customer, Address}
- và
  - {Order, Product, Quantity and UnitPrice}

## Chuẩn hoá về 2NF

1NF

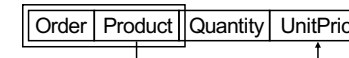


53

## Chuẩn hoá về 2NF

- R1 hiện tại đã ở dạng chuẩn 2NF, nhưng vẫn còn một phụ thuộc hàm bộ phận trong R2:

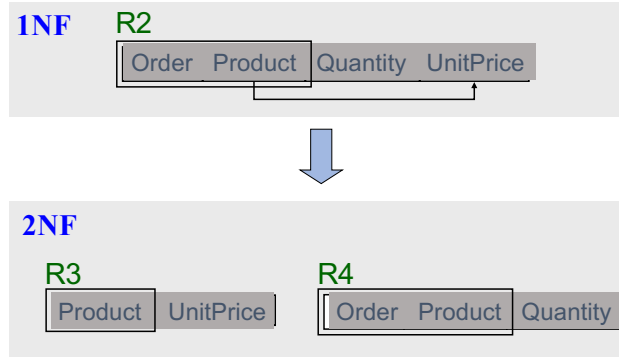
$\{Product\} \rightarrow \{UnitPrice\}$



- Để loại bỏ phụ thuộc hàm này chúng ta tiếp tục *tách*:  
 $\{Product, UnitPrice\}$  và  $\{Order, Product, Quantity\}$

54

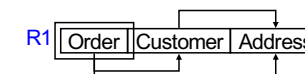
## Chuẩn hoá về 2NF



55

## Hãy tiếp tục đến dạng chuẩn 3NF...

- R hiện đã được tách thành 3 quan hệ - R1, R3 và R4... nhưng R1 có một FD bắc cầu trên khóa của nó...

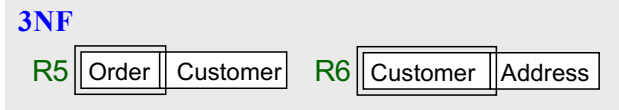
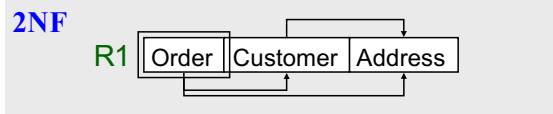


$\{Order\} \rightarrow \{Customer\} \rightarrow \{Address\}$

- Để loại bỏ vấn đề này chúng ta *tách* R1 thành:  
 $\{Order, Customer\}$  và  $\{Customer, Address\}$

56

Vì vậy, cắt nhiều hơn...



57

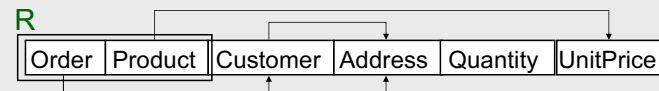
Hãy tổng kết lại:

- ❖ 1NF:  
{Order, Product, Customer, Address, Quantity, UnitPrice}
- ❖ 2NF:  
{Order, Customer, Address}  
{Product, UnitPrice}  
{Order, Product, Quantity}
- ❖ 3NF:  
{Product, UnitPrice}  
{Order, Product, Quantity}  
{Order, Customer}  
{Customer, Address}

58

Từ ban đầu...

0NF



59

Đã trở thành...

3NF

Prices

Amounts

Purchase

Details

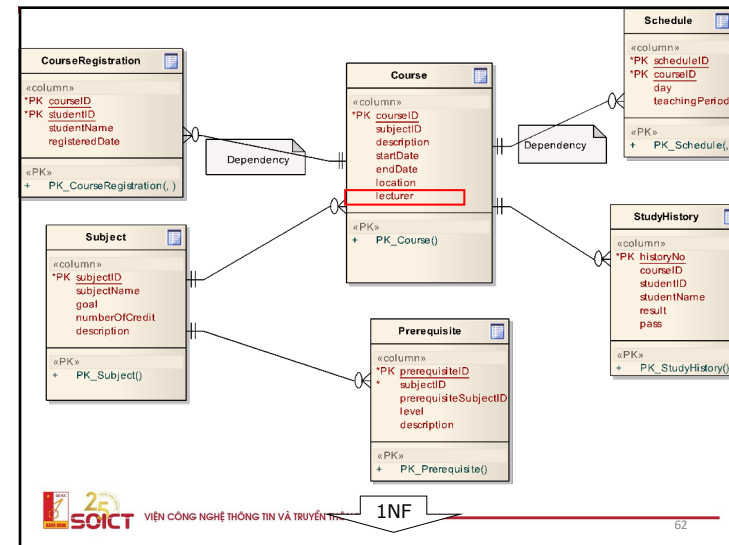
60

## “Register for course” use case

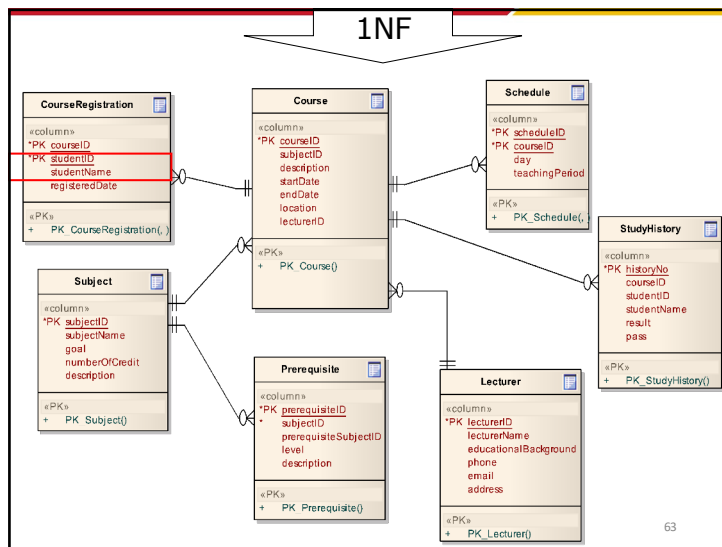
❖ Lập sơ đồ E-R từ bước trước cho trường hợp sử dụng “Đăng ký khóa học” trở thành:

- The first normal form
- The second normal form
- The third normal form

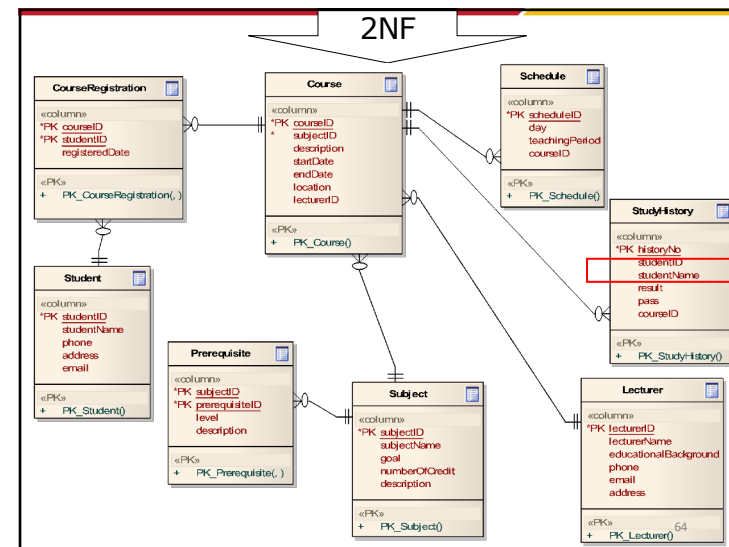
61



62

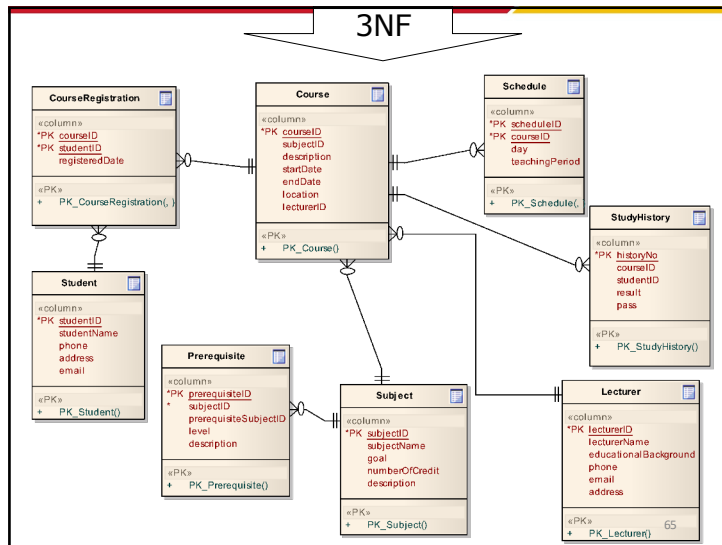


63



64





65

**Question?**


 VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

66

66



**Thank you  
for your  
attentions!**

[soict.hust.edu.vn/](http://soict.hust.edu.vn/)
[fb.com/groups/soict](https://fb.com/groups/soict)



67