



TRƯỜNG ĐẠI HỌC MỞ  
TP HỒ CHÍ MINH

# BÀI GIẢNG CƠ SỞ DỮ LIỆU

## Chương 2: Môi trường của cơ sở dữ liệu



# Nội dung chương 2 (1)

- Mục đích của kiến trúc CSDL 3 mức
- Nội dung của mức ngoài, mức ý niệm và mức trong (external, conceptual, and internal levels)
- Mục đích của việc ánh xạ mức ngoài/mức ý niệm và mức ý niệm/mức trong
- Ý nghĩa của sự độc lập dữ liệu giữa logic và vật lý
- Phân biệt DDL và DML
- Phân loại các mô hình dữ liệu





## Nội dung chương 2 (2)

- Mục đích và tầm quan trọng của việc mô hình hóa mức ý niệm
- Các chức năng và dịch vụ chính yếu của DBMS
- Các thành phần phần mềm của DBMS
- Ý nghĩa của kiến trúc client–server và sự tiện lợi của kiến trúc này đối với DBMS
- Chức năng và các trường hợp dùng Bộ giám sát việc xử lý giao tác (Transaction Processing Monitor)
- Chức năng và tầm quan trọng của catalog hệ thống (system catalog)



# Mục tiêu của kiến trúc 3 mức

- Tất cả users có thể truy xuất dữ liệu như nhau
- View của một user không bị thay đổi khi các view khác bị thay đổi
- Users không cần phải biết chi tiết ở mức vật lý cơ sở dữ liệu được lưu như thế nào



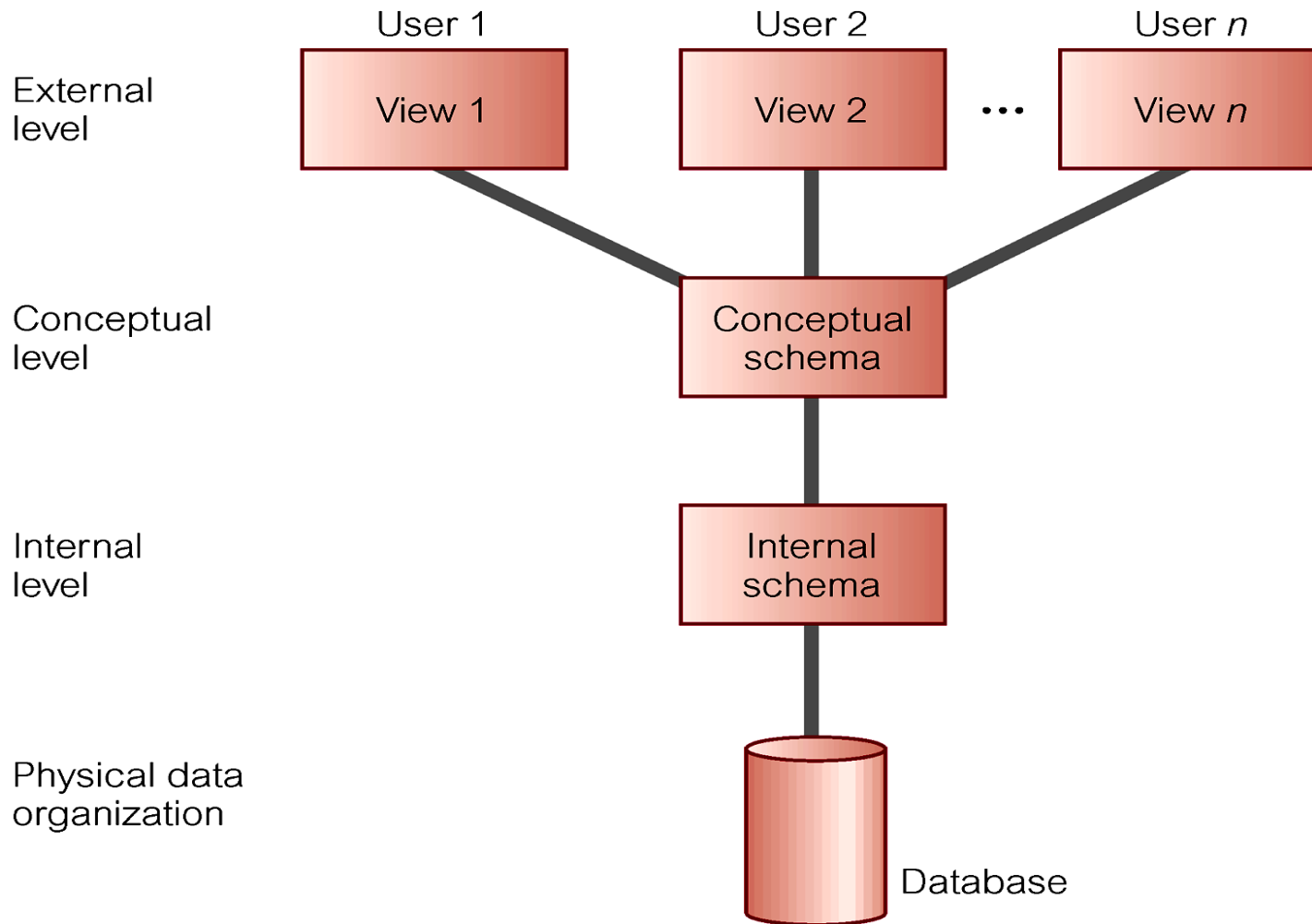


# Mục tiêu của kiến trúc 3 mức

- Người quản trị CSDL (DBA) có thể thay đổi cấu trúc lưu trữ CSDL nhưng không làm ảnh hưởng đến views của các users
- Cấu trúc mức trong của CSDL không bị ảnh hưởng bởi sự thay đổi các yếu tố mức vật lý
- DBA có thể thay đổi cấu trúc mức ý niệm nhưng không làm ảnh hưởng đến tất cả các users



# Kiến trúc 3 mức ANSI-SPARC (ANSI-SPARC Three-level Architecture)





# Kiến trúc 3 mức ANSI-SPARC

- Mức ngoài (External Level)
  - gồm các views của các users
  - miêu tả từng phần của CSDL có liên quan đến từng user
- Mức ý niệm (Conceptual Level)
  - là một view tổng quát cho cả CSDL
  - miêu tả dữ liệu gì được lưu trong CSDL và mối quan hệ giữa các dữ liệu đó ra sao



# Kiến trúc 3 mức ANSI-SPARC

- Mức trong (Internal Level)
  - là sự thể hiện ở mức vật lý của CSDL trên máy tính
  - miêu tả dữ liệu được lưu như thế nào trong CSDL







# Sự khác nhau giữa 3 mức

External view 1

sNo	fName	lName	age	salary
-----	-------	-------	-----	--------

External view 2

staffNo	lName	branchNo
---------	-------	----------

Conceptual level

staffNo	fName	lName	DOB	salary	branchNo
---------	-------	-------	-----	--------	----------

Internal level

```
struct STAFF {  
    int staffNo;  
    int branchNo;  
    char fName [15];  
    char lName [15];  
    struct date dateOf Birth;  
    float salary;  
    struct STAFF *next;  
};  
index staffNo; index branchNo;
```

/\* pointer to next Staff record \*/  
/\* define indexes for staff \*/



# Sự độc lập về dữ liệu (Data Independence)

- Độc lập về mặt logic:
  - Loại trừ sự thay đổi của lược đồ mức ngoài khi lược đồ mức ý niệm thay đổi
  - Cho phép lược đồ mức ý niệm thay đổi (ví dụ khi thêm/bớt thực thể)
  - Không yêu cầu thay đổi của lược đồ mức ngoài hay viết lại chương trình ứng dụng

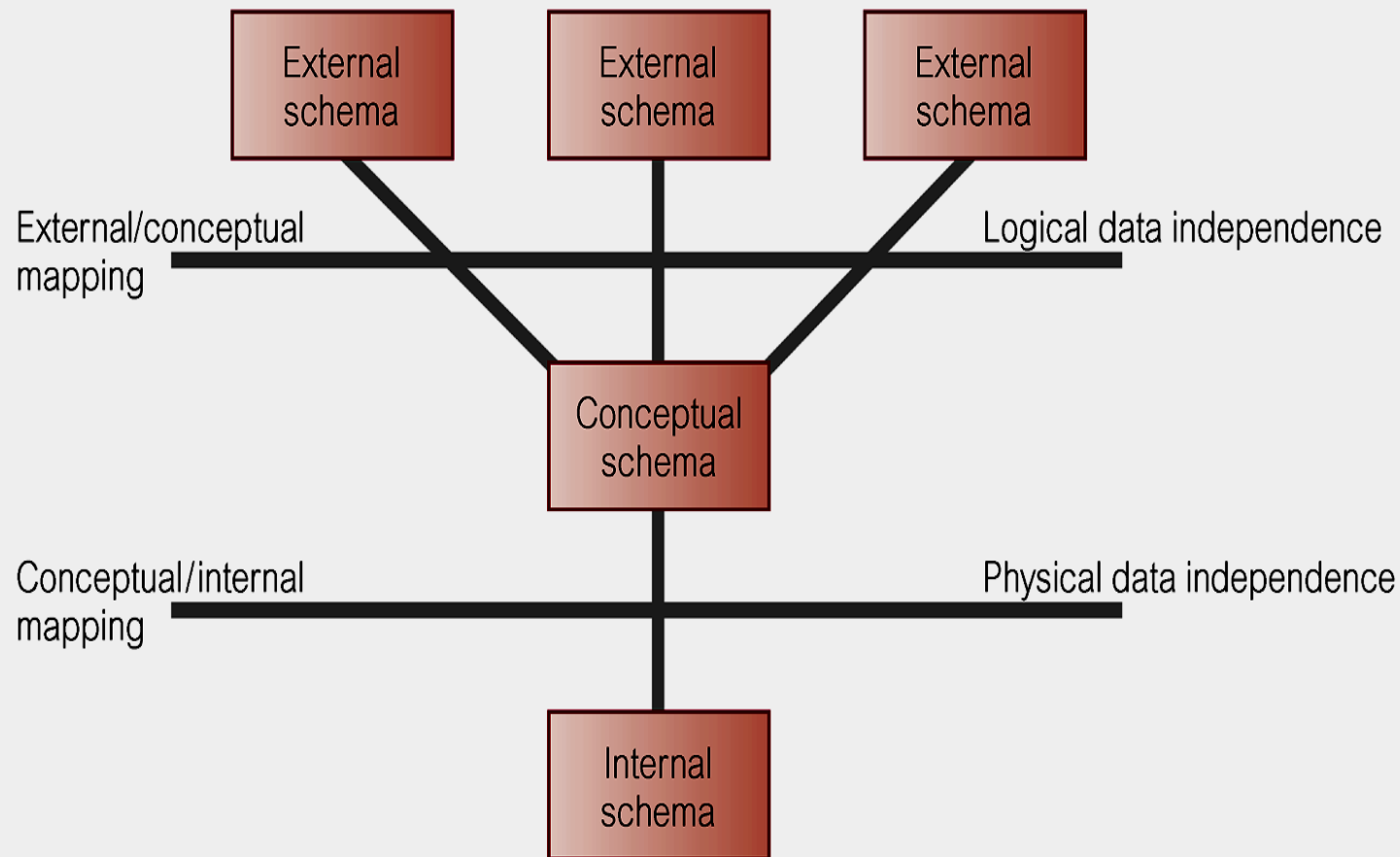


# Sự độc lập về dữ liệu (Data Independence)

- Độc lập về mặt vật lý:
  - Loại trừ sự thay đổi của lược đồ mức ý niệm khi lược đồ mức trong thay đổi
  - Cho phép lược đồ mức trong thay đổi (ví dụ khi dùng cách tổ chức file khác nhau, các thiết bị hay các cấu trúc lưu trữ khác nhau)
  - Không yêu cầu thay đổi lược đồ mức ý niệm hay mức ngoài



# Độc lập dữ liệu của kiến trúc ANSI-SPARC 3 mức





# Ngôn ngữ cho CSDL (Database Languages) (1)

## Ngôn ngữ định nghĩa dữ liệu (DDL)

- Cho phép người quản trị CSDL (DBA) hoặc user miêu tả và đặt tên cho các thực thể, các thuộc tính, và các mối quan hệ cần thiết của ứng dụng
- Ngoài ra còn tạo sự toàn vẹn trong kết hợp DL (associated integrity) và sự ràng buộc an toàn DL (security constraints)



# Ngôn ngữ cho CSDL (Database Languages) (2)

## Ngôn ngữ thao tác dữ liệu DML

- Cung cấp các phép thực hiện thao tác cơ bản trên dữ liệu đang lưu trong CSDL
- Ngôn ngữ DML thủ tục (Procedural DML)
  - Cho phép user ra lệnh một cách chi tiết cho hệ thống phải làm như thế nào (HOW) để thao tác dữ liệu
- Ngôn ngữ DML phi thủ tục (Non-Procedural DML)
  - Cho phép user đưa ra yêu cầu về dữ liệu đang cần (WHAT), chứ không nói cách thức hệ thống phải truy xuất dữ liệu đó như thế nào

## Ngôn ngữ điều khiển dữ liệu DCL

- Là ngôn ngữ điều khiển dữ liệu, sử dụng để phân quyền người dùng
- Bao gồm các lệnh:
  - Lệnh GRANT: Trao một quyền tới người dùng.
  - Lệnh REVOKE: Thu hồi quyền đã trao cho người dùng.



# Ngôn ngữ cho CSDL (Database Languages) (3)

- Ngôn ngữ thế hệ thứ 4 (Fourth Generation Language (4GL))
  - Ngôn ngữ truy vấn (Query Languages) ví dụ ngôn ngữ SQL, ngôn ngữ QBE
  - Công cụ tạo cửa sổ giao diện (Forms Generators)
  - Công cụ tạo bảng báo cáo (Report Generators)
  - Công cụ tạo biểu đồ dữ liệu (Graphics Generators)
  - Công cụ tạo ứng dụng (Application Generators)





# Mô hình dữ liệu (Data Model) (1)

- Là tập hợp các khái niệm có liên quan, dùng để miêu tả dữ liệu, mối quan hệ giữa các dữ liệu, và các ràng buộc trên dữ liệu của một tổ chức
- Mô hình dữ liệu bao gồm:
  - Một phần cấu trúc (A structural part)
  - Một phần thao tác (A manipulative part)
  - Có thể có một tập các qui tắc toàn vẹn (integrity rules)





# Mô hình dữ liệu (Data Model) (2)

- Mục đích
  - Biểu diễn dữ liệu theo cách thức dễ hiểu
  - Các loại mô hình dữ liệu gồm:
    - Mô hình lấy đối tượng làm nền tảng (Object-based)
    - Mô hình lấy record làm nền tảng (Record-based)
    - Mô hình vật lý (Physical)





- **Mô hình dữ liệu** là tập hợp các **khái niệm** trong việc miêu tả, thao tác với dữ liệu.
  - Có thể coi đó là một tập khái niệm và ký hiệu cung cấp cho việc thiết kế CSDL của người thiết kế.
  - Một mô hình dữ liệu gồm các thành phần như: Cấu trúc (các luật của CSDL), Thao tác (cập nhật, thay đổi), Tập các luật về tính toàn vẹn...



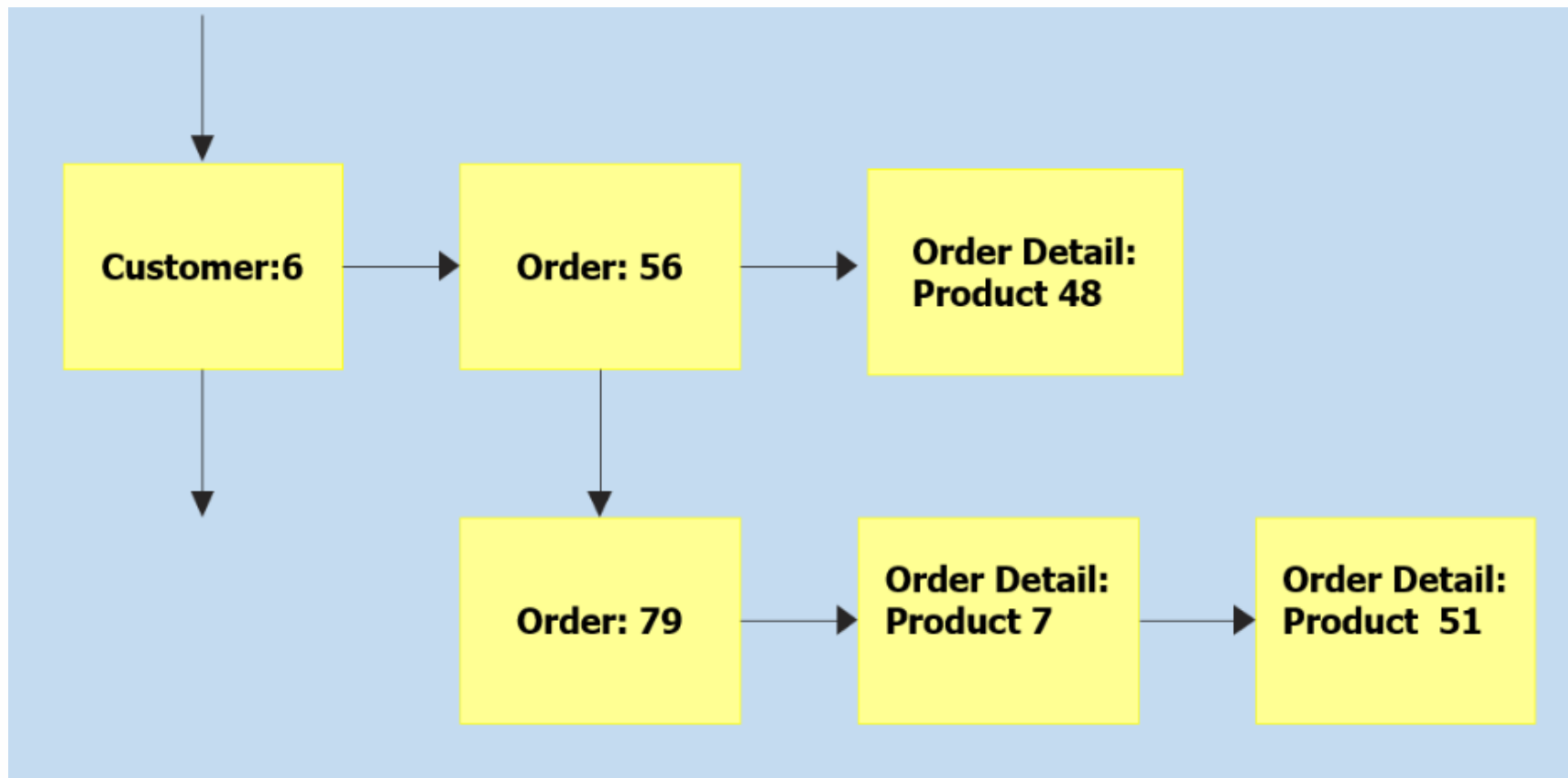
# Các mô hình dữ liệu (Data Models)

## Object-based Data Models

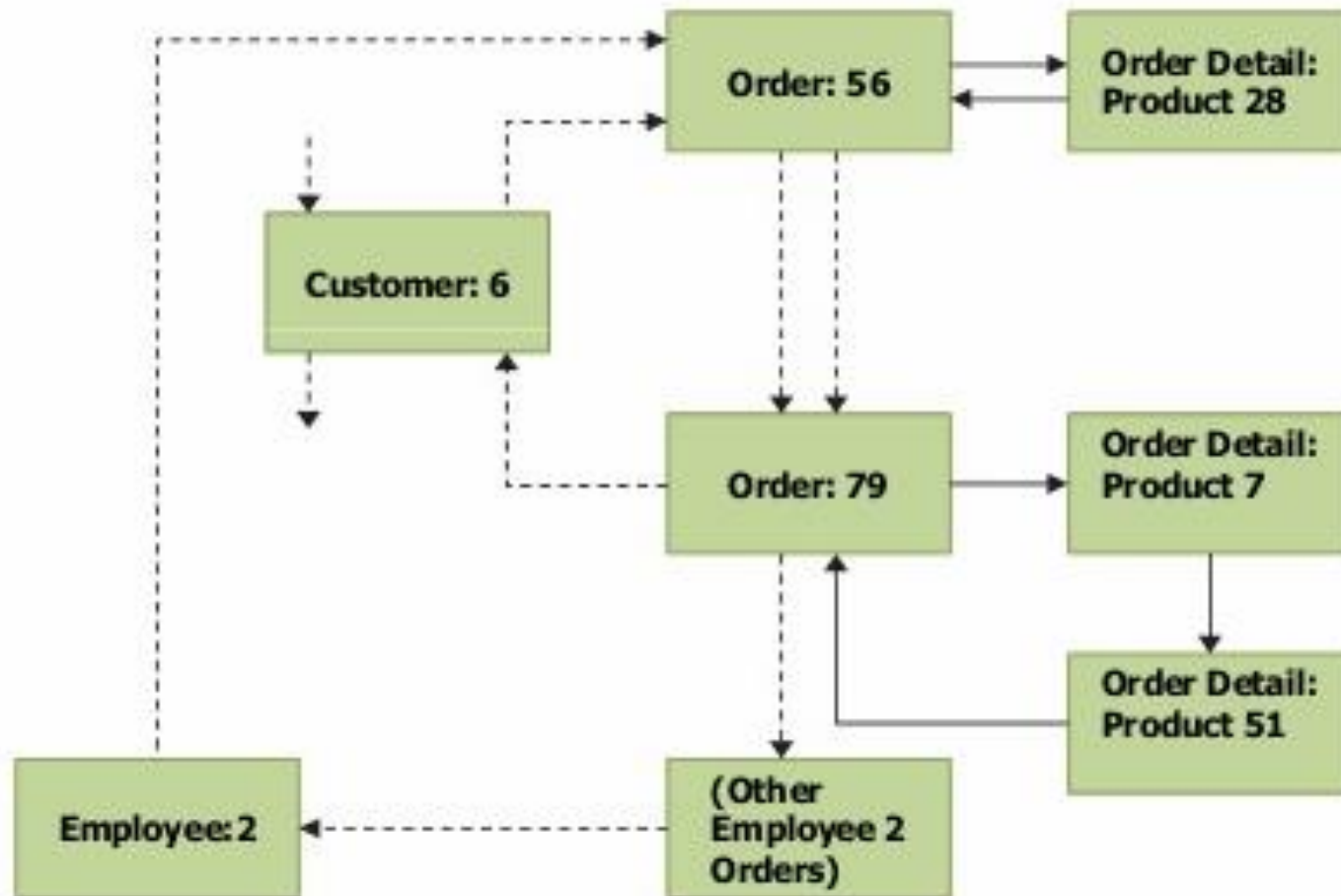
- Mô hình thực thể-mối kết hợp (Entity-Relationship)
- Mô hình ngữ nghĩa (Semantic)
- Mô hình chức năng (Functional)
- Mô hình hướng đối tượng (Object-Oriented)
- Record-based Data Models
- Mô hình quan hệ (Relational Data Model)
- Mô hình mạng (Network Data Model)
- Mô hình phân cấp (Hierarchical Data Model)
- Physical Data Models



# Mô hình dữ liệu phân cấp trong CSDL Northwind

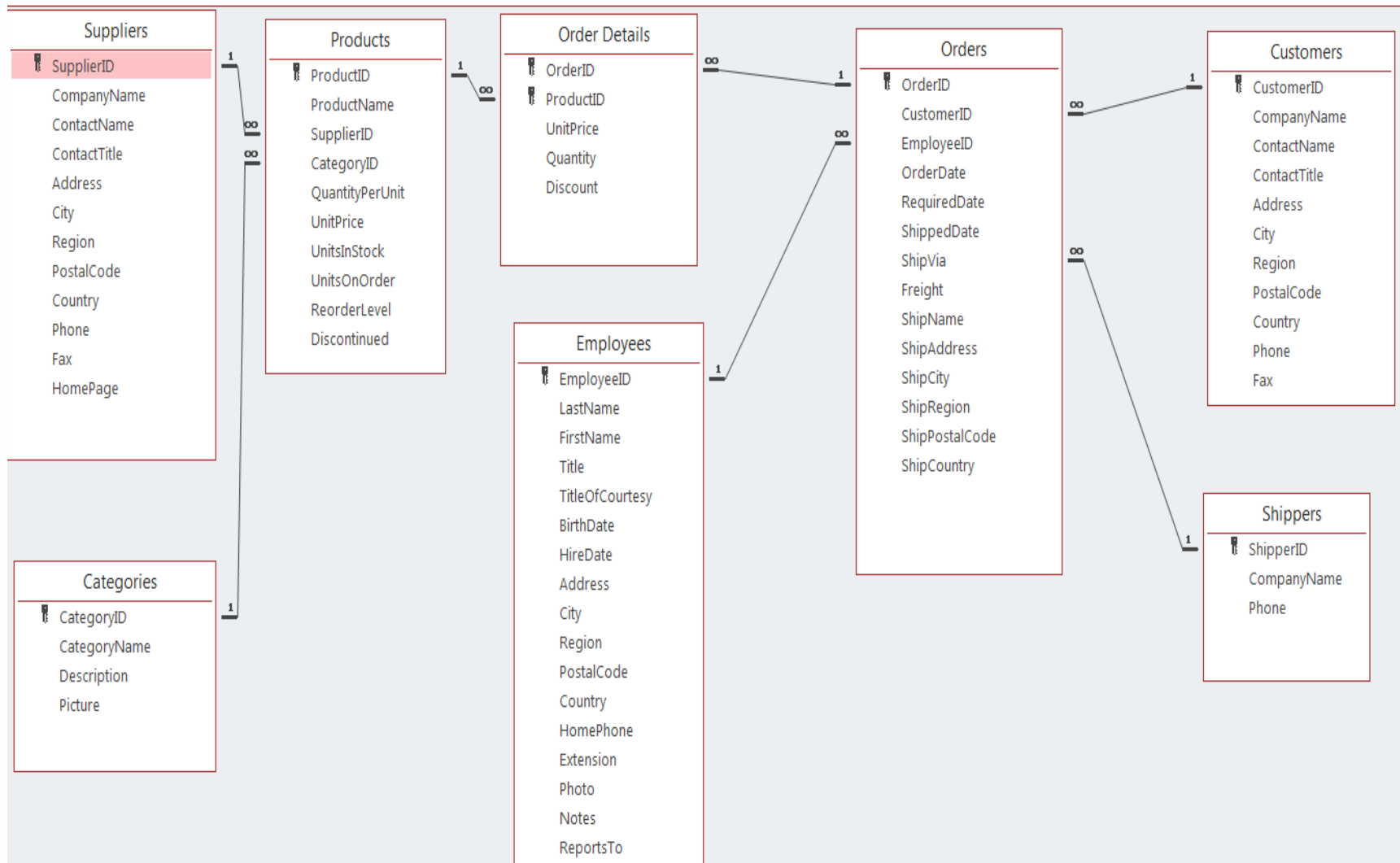


# Mô hình mạng trong CSDL Northwind



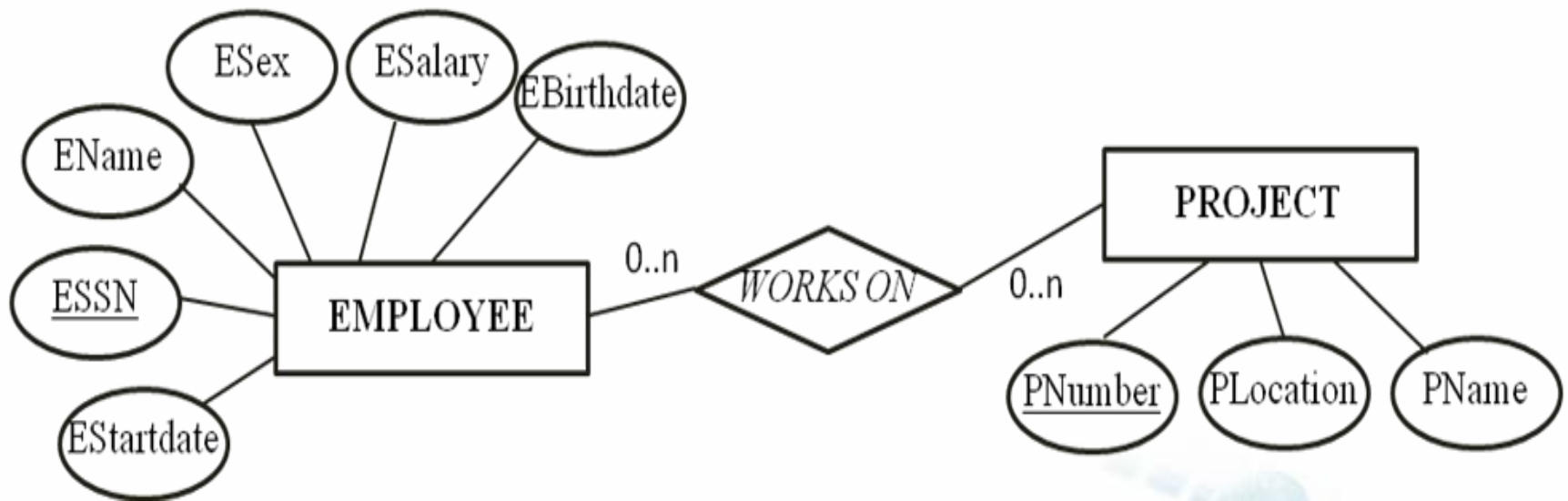


# Mô hình dữ liệu quan hệ trong CSDL Northwind





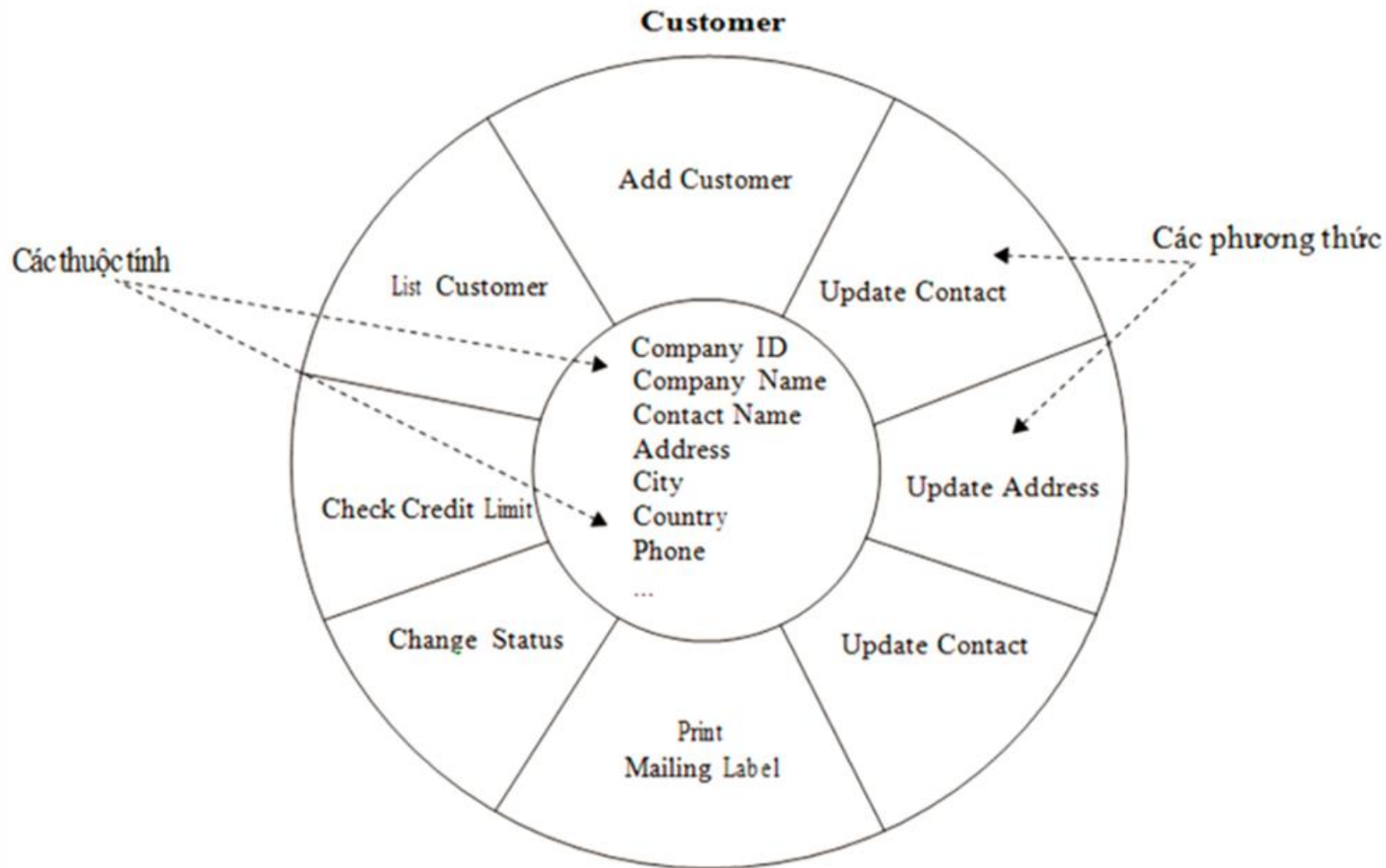
# Mô hình thực thể mối kết hợp







# Mô hình dữ liệu đối tượng “Customer”



# Mô hình mạng ngữ nghĩa: Ontology



Template Slots			
Name	Cardinality	Type	
Cấp	single	String	
Khả năng di chuyển	single	Boolean	default=true
Nơi sống	single	String	
			Ràng buộc



# Mô hình mạng ngữ nghĩa: Ontology

Cơ sở dữ liệu - Môi trường của cơ sở dữ liệu

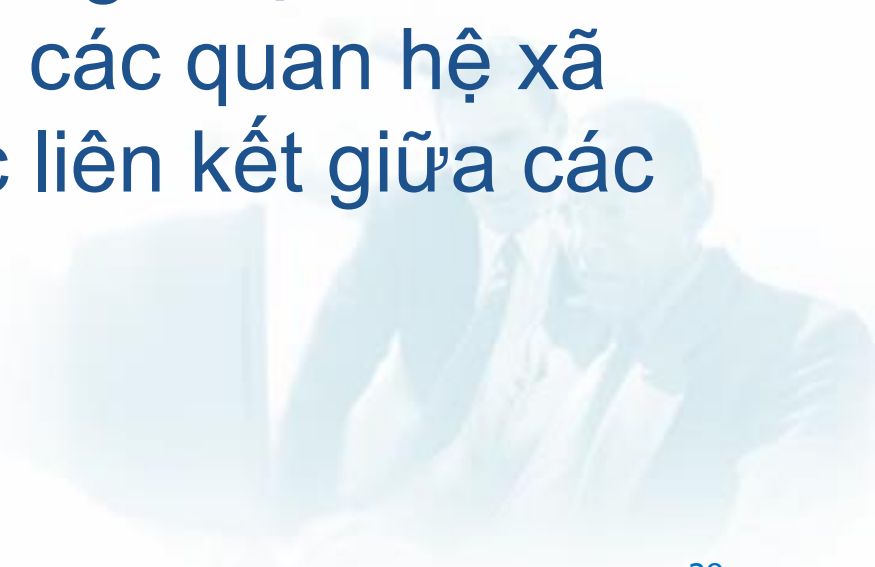
The screenshot displays the Protégé ontology editor with the following components:

- Class hierarchy (Left):** A tree view showing the ontology structure. The **Storm** class is highlighted, with its subclasses **S1**, **S2**, and **S3** listed below it. A red label **Tập C** is placed next to this section.
- General class axioms (Top Center):** Displays logical axioms such as **Rain and (hasPrecipitation some rs5-7.5) SubClassOf HeavyRain**. A red label **Tập Z** is placed next to this section.
- Individuals: WindEvent (Top Right):** A list of individuals including **Speed\_Racer\_Sr**, **Thunderstorm\_Ice\_Pellets**, **Uchi\_Kuro\_Bushi**, **Under**, **WATER\_VAPOR**, **Webbing\_of\_the\_Hand**, **Wind1**, **WindEvent1**, and **X\_Racer**. A red label **Tập I** is placed next to this section.
- Property assertions: WindEvent (Middle Right):** Shows assertions for the **hasWindSpeed** property, with a value range of **{(40-50);{(0-10)}}**.
- Data property hierarchy (Bottom Center):** A list of data properties including **hasDate**, **hasDegree**, **hasHumidity**, **hasPrecipitation**, **hasPressure**, **hasProtons**, **hasReportModifier**, **hasSaltPercent**, **hasSnow**, **hasSpeedValue**, **hasSunshine**, **hasSymbol**, **hasTemperature**, **hasTime**, **hasUnits**, **hasWindDirection**, and **hasWindSpeed**. A red label **Tập R** is placed next to this section.



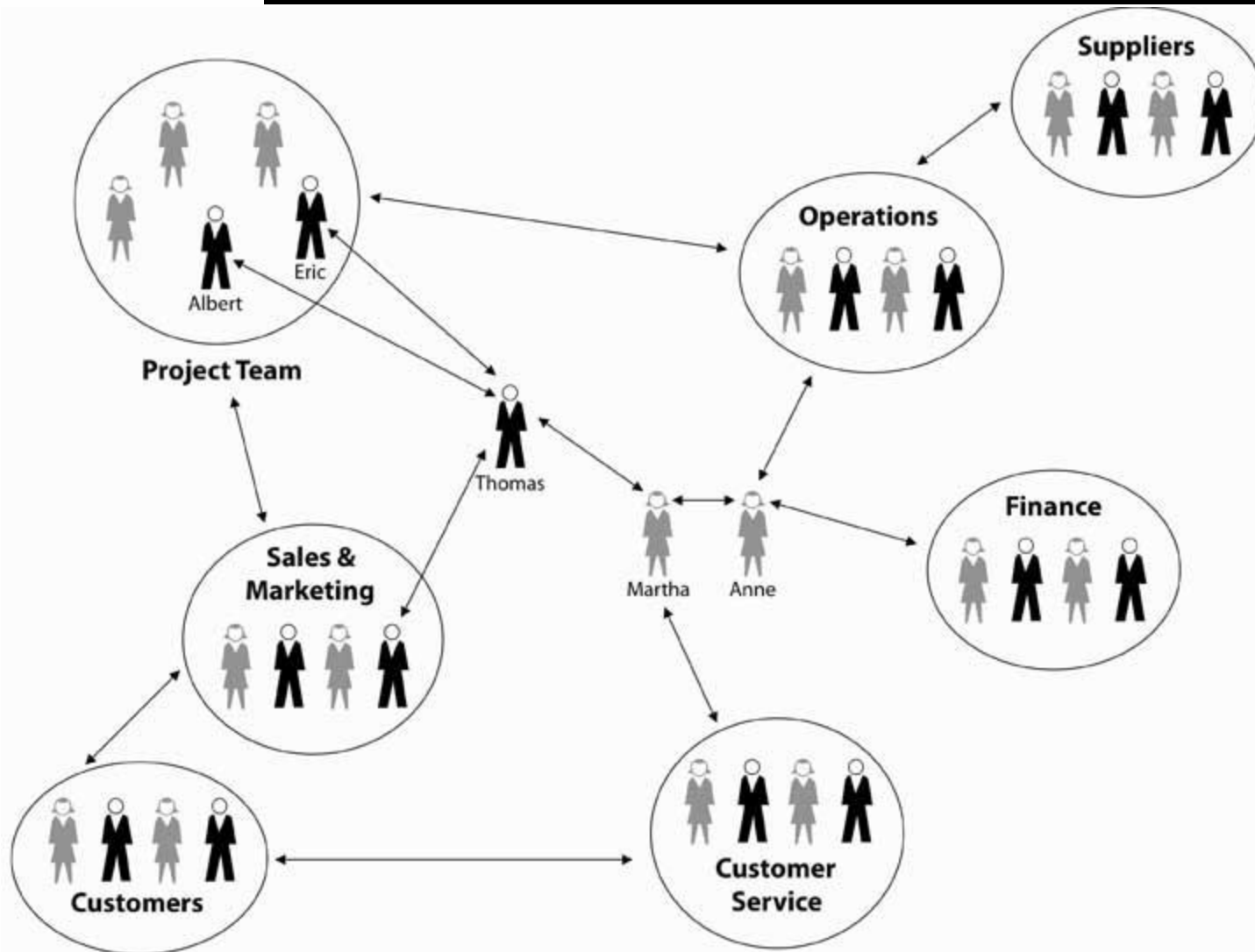
# Mạng xã hội - social network

- Mạng xã hội: nút là tác nhân xã hội và cạnh là quan hệ/tương tác giữa các tác nhân đó.
- Là một cấu trúc xã hội bao gồm các cá nhân hay tổ chức, thường được biểu diễn bằng các nút, cùng với các quan hệ xã hội, tương ứng với các liên kết giữa các nút.





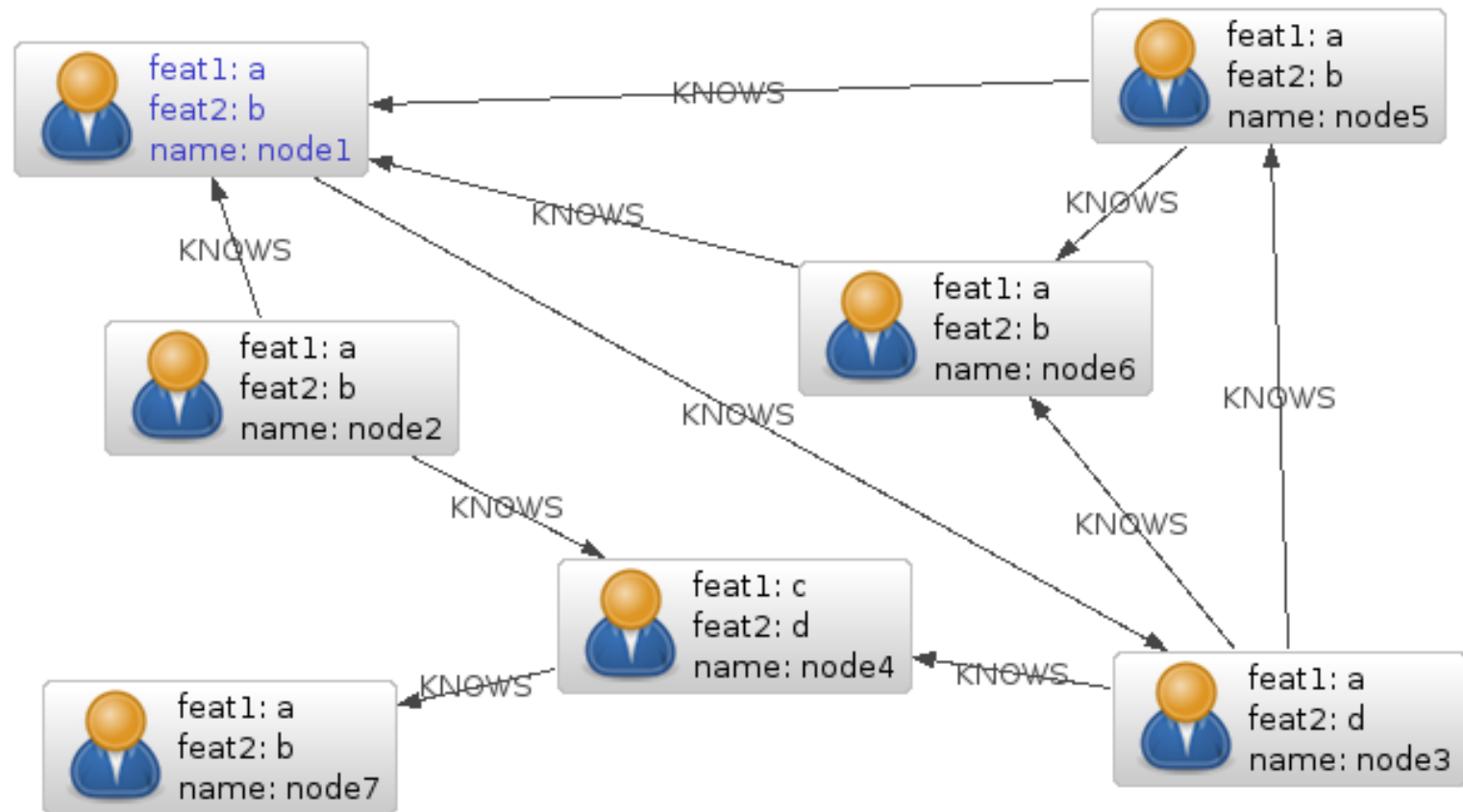
# Mạng xã hội - social network

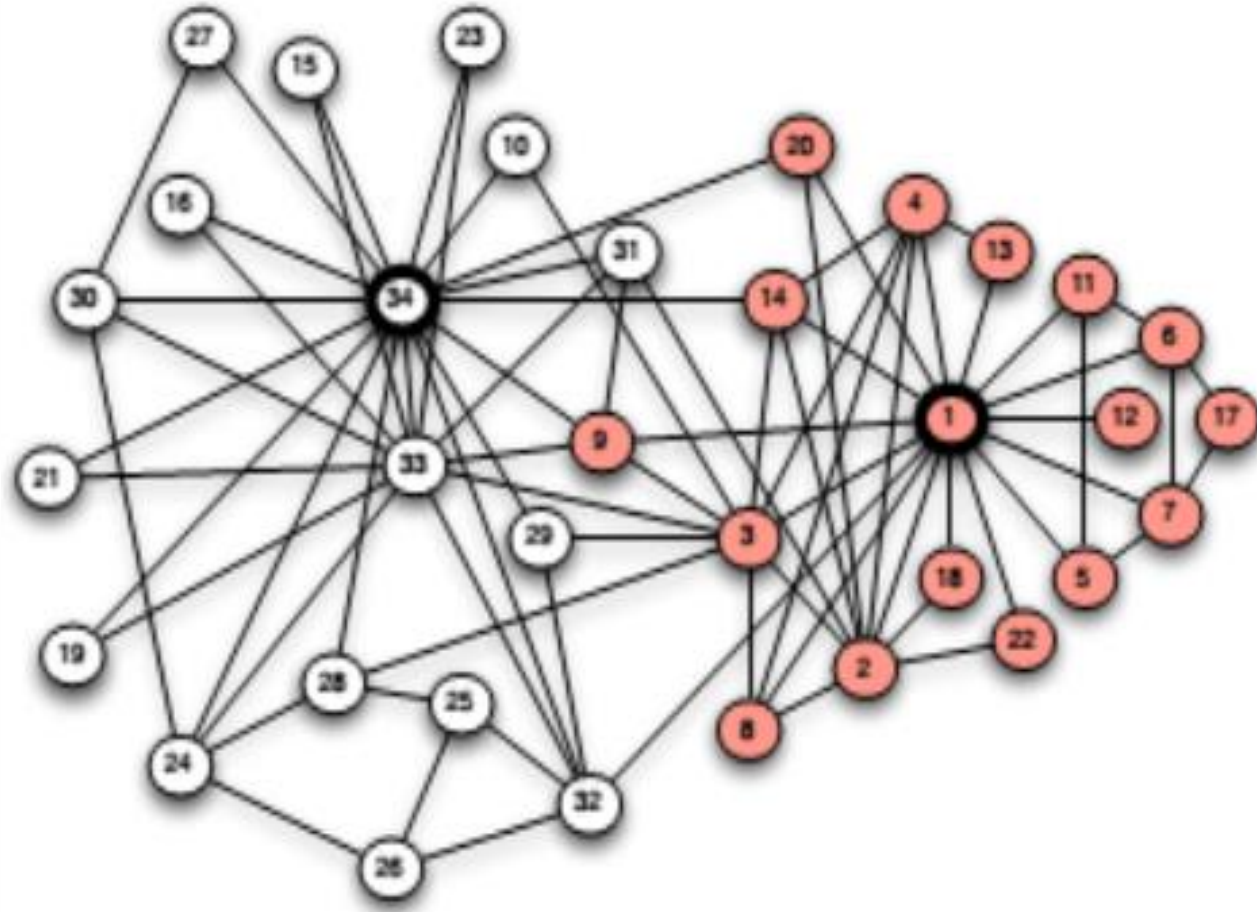






# Mạng xã hội - social network





[Easley10] David Easley and Jon Kleinberg (2010). Networks, Crowds, and Markets: Reasoning about a Highly Connected World, Cambridge University Press, 2010



# NHIỆM VỤ CỦA HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

## Định nghĩa cơ sở dữ liệu

- Đặc tả kiểu dữ liệu, các cấu trúc, các ràng buộc toàn vẹn, ...

## Xây dựng cơ sở dữ liệu

- Lưu trữ các dữ liệu trên các phương tiện lưu trữ, ...

## Thao tác cơ sở dữ liệu

- Truy vấn, cập nhật dữ liệu, ...





# PHÂN LOẠI HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

## Dựa trên mô hình dữ liệu

- Mô hình dữ liệu quan hệ, mô hình dữ liệu phân cấp, mô hình dữ liệu đối tượng, mô hình dữ liệu đối tượng – quan hệ

## Dựa trên số người dùng

- Hệ thống đơn người dùng, hệ thống đa người dùng

## Dựa trên vị trí đặt hệ thống

- Hệ thống tập trung, hệ thống phân tán





# Các chức năng của DBMS (1)

1. Lưu, truy xuất và cập nhật dữ liệu
2. Có phân quyền truy xuất cho các user
3. Hỗ trợ giao tác (giao dịch)
4. Có dịch vụ kiểm soát đồng thời
5. Có dịch vụ phục hồi dữ liệu



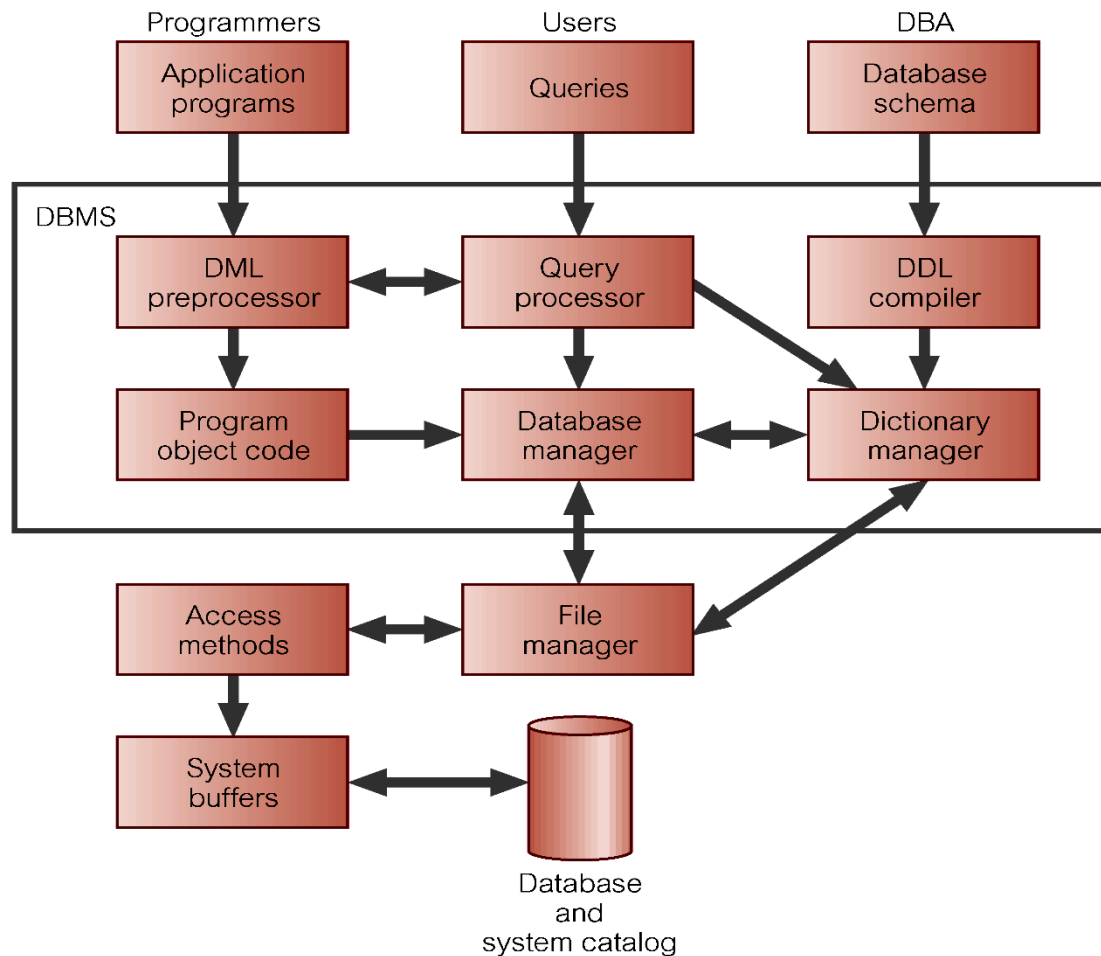


## Các chức năng của DBMS (2)

6. Có dịch vụ kiểm tra quyền truy xuất dữ liệu
7. Hỗ trợ việc giao tiếp dữ liệu (kết hợp được với các software khác để thực hiện)
8. Có dịch vụ toàn vẹn dữ liệu
9. Hỗ trợ việc độc lập dữ liệu giữa CSDL và chương trình
10. Có các dịch vụ tiện ích khác: import, export, thống kê, giám sát, ...

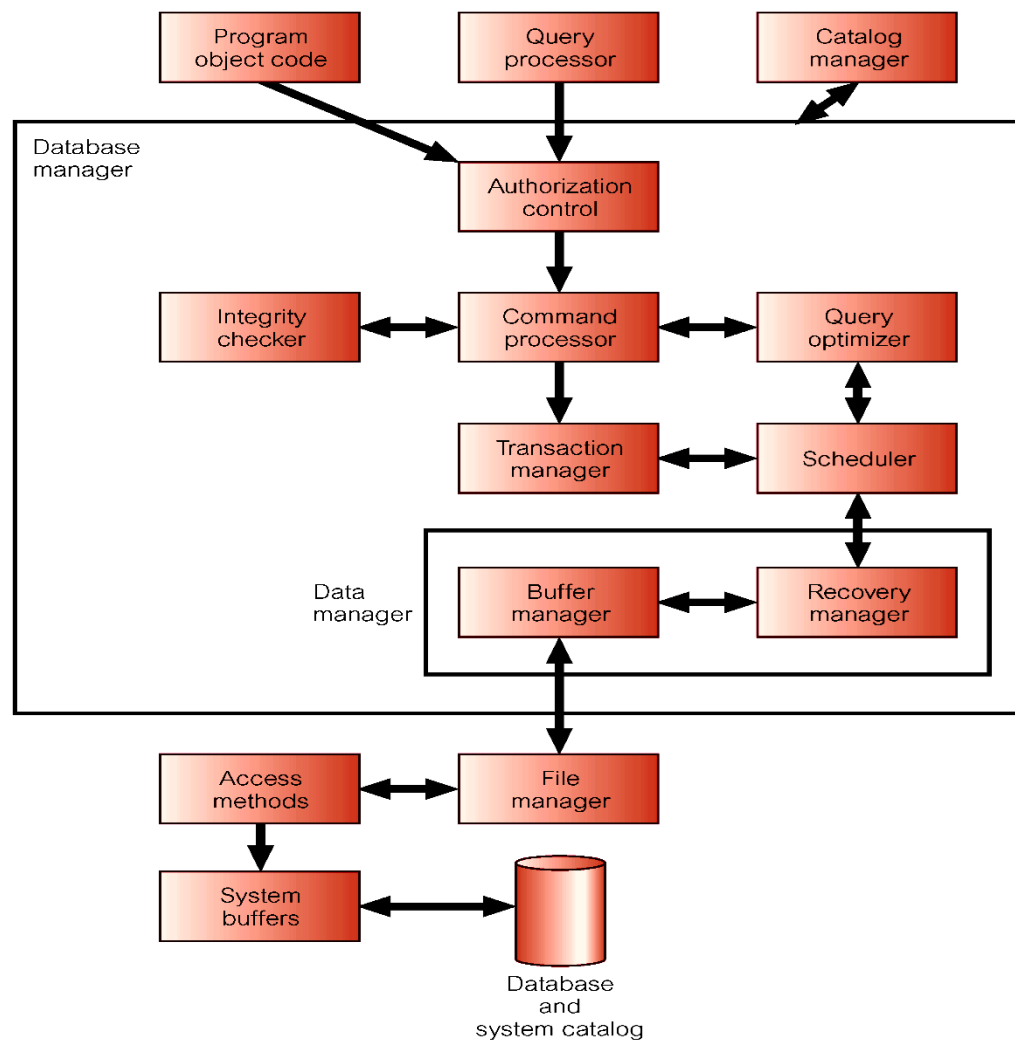


# Các thành phần của DBMS





# Các thành phần của Database Manager (DM)





# Các kiến trúc DBMS đa người dùng

- Xử lý từ xa (Teleprocessing)
- Máy phục vụ file (File-server)
- Máy khách - máy phục vụ (Client-server)





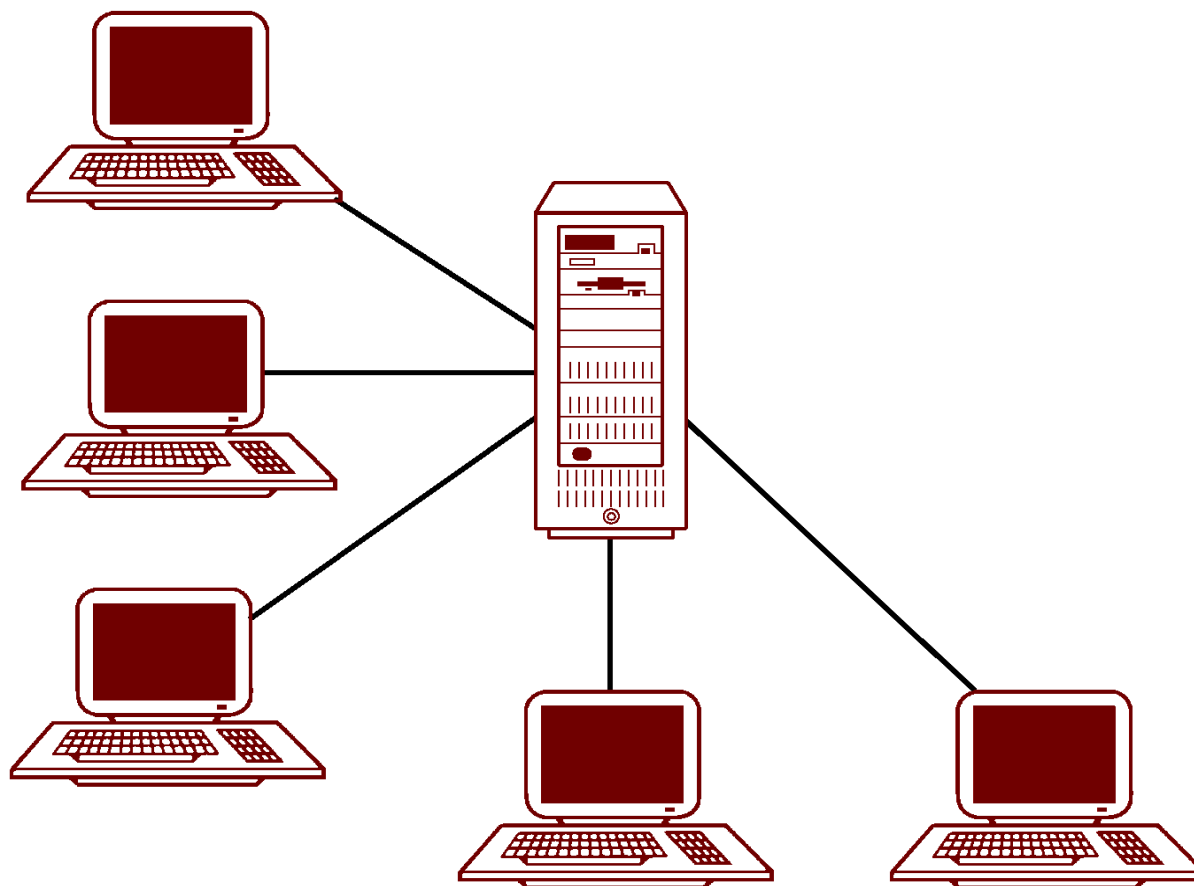
# Kiến trúc xử lý từ xa (Teleprocessing) (1)

- Là kiến trúc truyền thống
- Một máy chính mạnh nối với một số máy đầu cuối (Single mainframe with a number of terminals attached)
  - Tất cả xử lý tập trung trên cùng 1 máy tính.
  - Các trạm đầu cuối gửi yêu cầu về máy trung tâm
- Khuynh hướng hiện nay ít dùng kiến trúc này, vì máy mainframe rất đắt tiền và các máy đầu cuối ngày càng mạnh hơn





# Kiến trúc xử lý từ xa (Teleprocessing) (2)



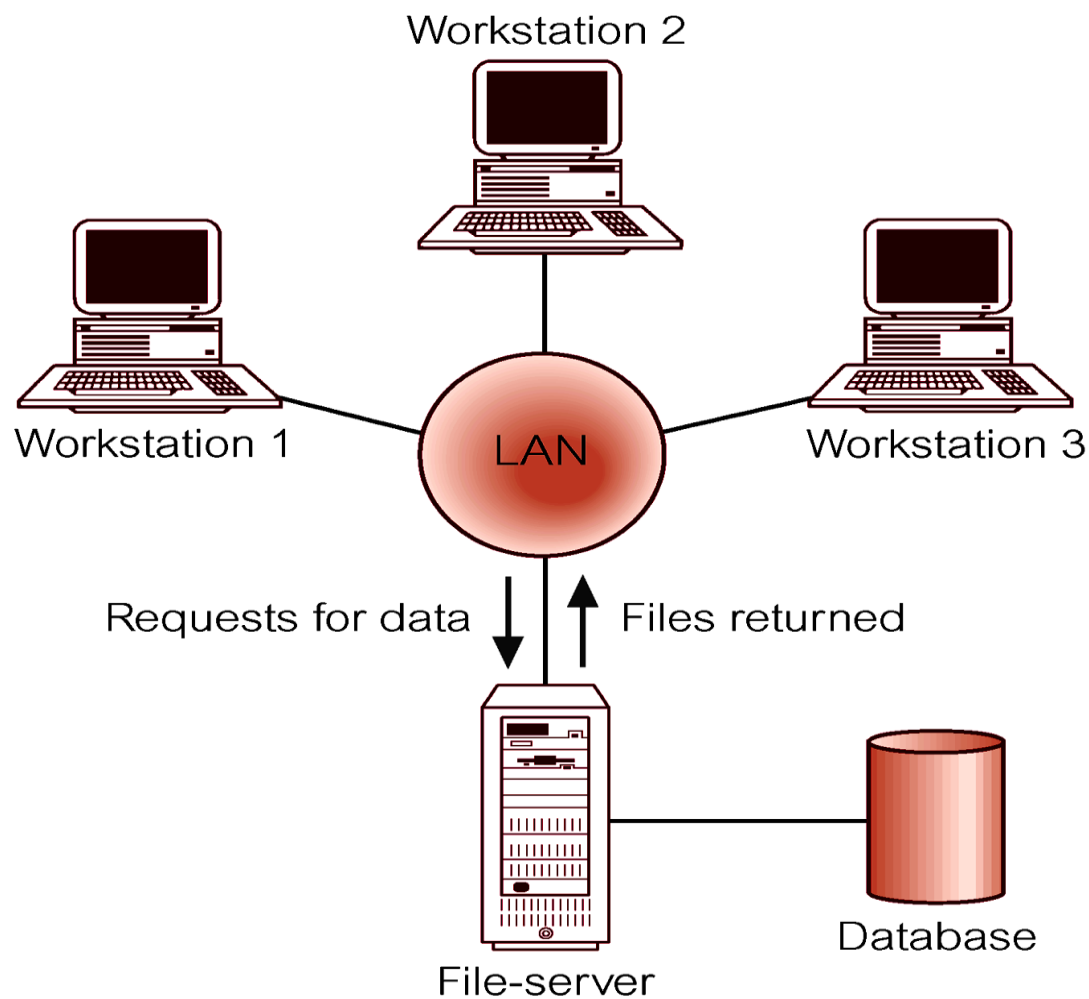


# Kiến trúc File-Server (1)

- Máy File-server được kết nối với nhiều máy trạm (workstation) nhờ mạng máy tính
- Database nằm ở máy file-server.
- DBMS và applications chạy trên các máy workstation.
- Các nhược điểm:
  - Chiếm đường truyền trên mạng nhiều
  - Nhiều bản sao DBMS trên các máy workstation
  - Phức tạp trong việc kiểm soát đồng thời, phục hồi dữ liệu và toàn vẹn dữ liệu



# Kiến trúc File-Server (2)





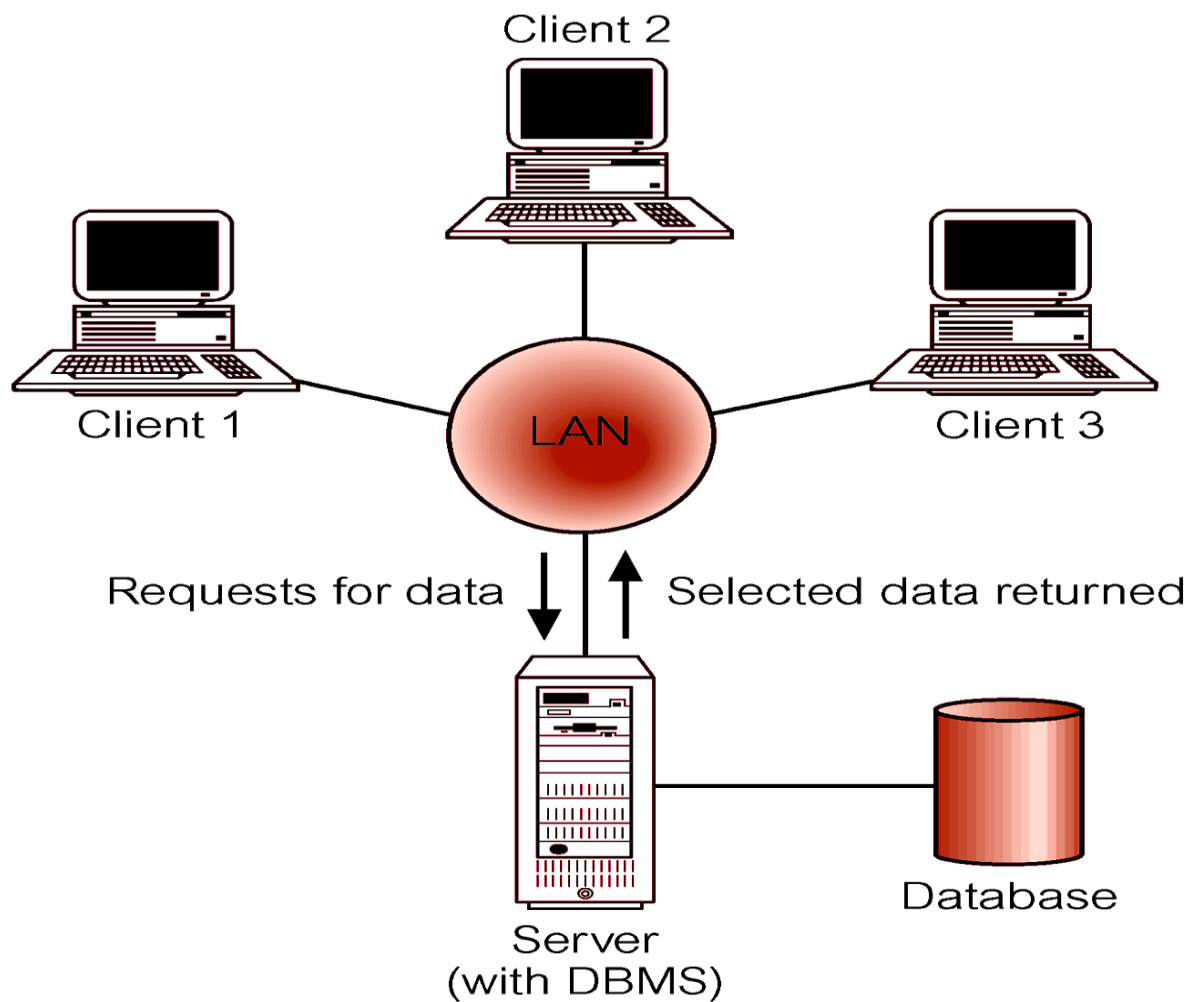
# Kiến trúc Client-server (1)

- Máy Server chứa database và chạy DBMS.
- Máy Client quản lý giao diện với user và chạy các ứng dụng
- Các ưu điểm:
  - Truy xuất rộng hơn đến các database
  - Tăng tốc độ
  - Có thể giảm giá thành phần cứng
  - Giảm chi phí truyền dữ liệu
  - Tăng tính nhất quán dữ liệu



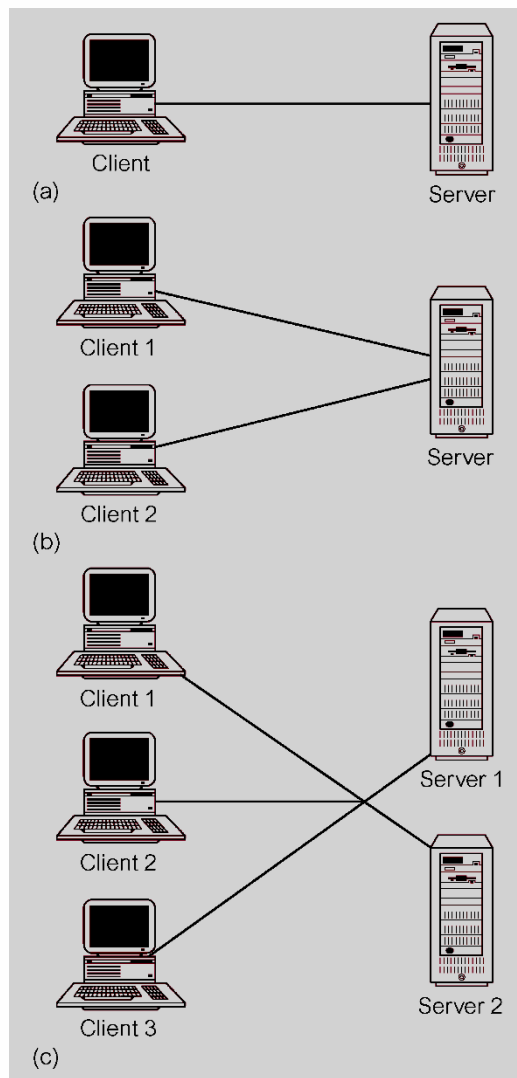


# Kiến trúc Client-server (2)





# Các sơ đồ kết nối của kiến trúc Client-server



- Tìm hiểu CSDL Northwind
- Tìm hiểu Hệ quản trị CSDL Access, Hệ quản trị CSDL SQL Server.





# Mô hình dữ liệu quan hệ trong CSDL Northwind

