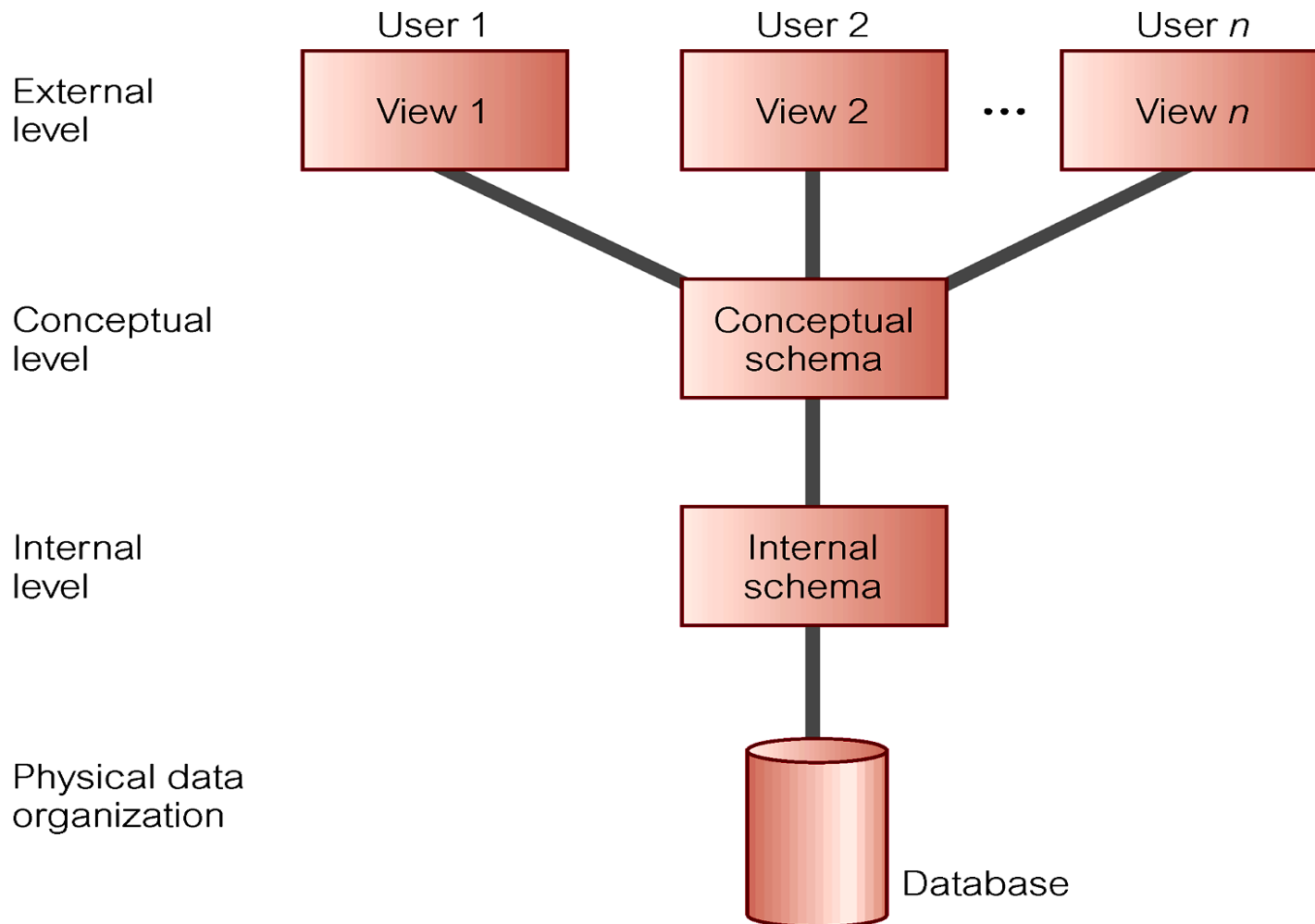


Chương 2: Môi trường của cơ sở dữ liệu

Tham khảo tài liệu [B]:
Chapter 2: Database Environment

Kiến trúc 3 mức ANSI-SPARC

(ANSI-SPARC Three-level Architecture)



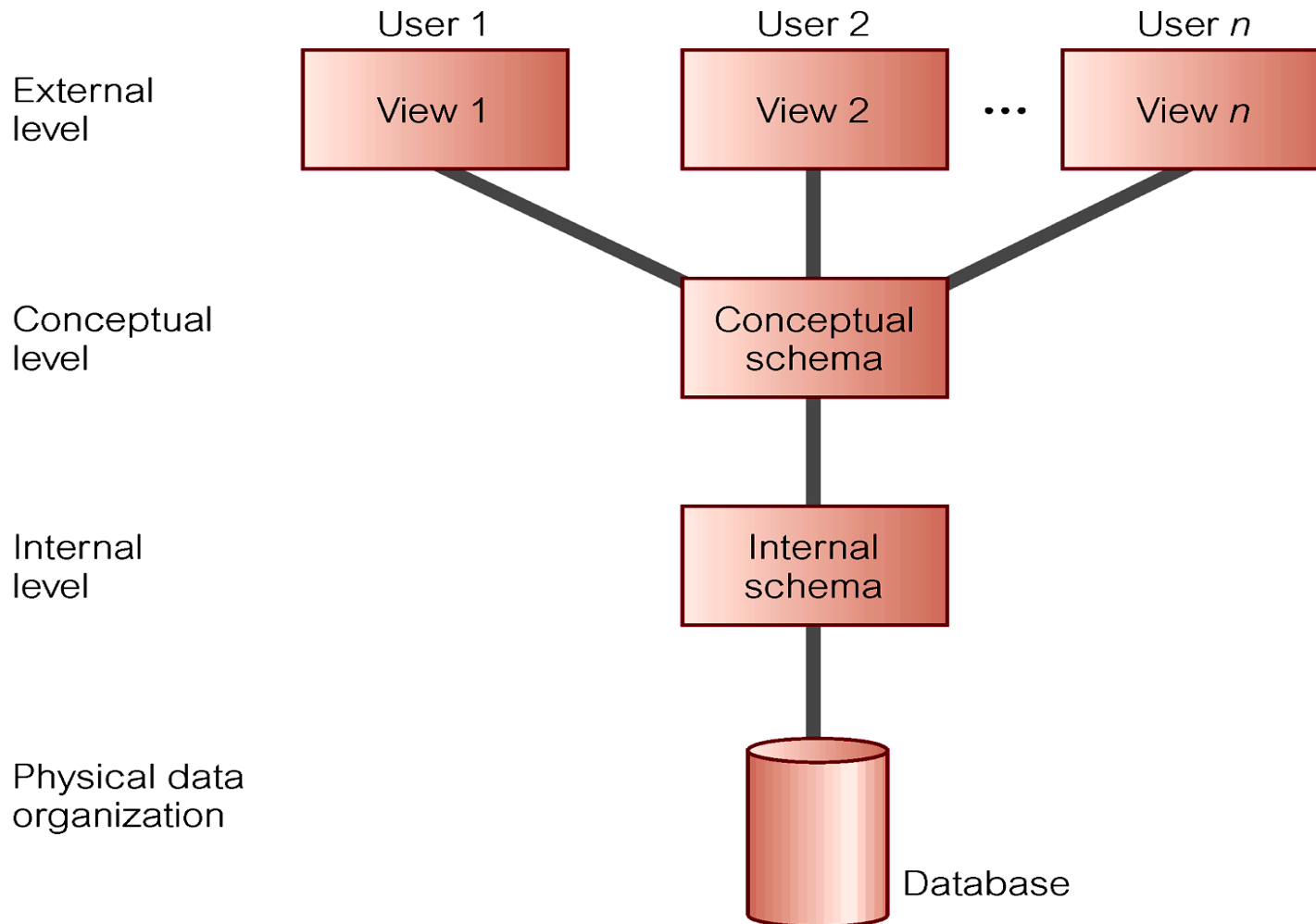
Mục tiêu của kiến trúc 3 mức

- ◆ 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 user
- ◆ Cấu trúc mức trong của CSDL không bị ảnh hưởng bởi sự thay đổi các yếu tố mức vật lý
- ◆ Người thiết kế CSDL có thể thay đổi cấu trúc mức ý niệm nhưng không làm ảnh hưởng đến tất cả các user

Chi tiết về 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

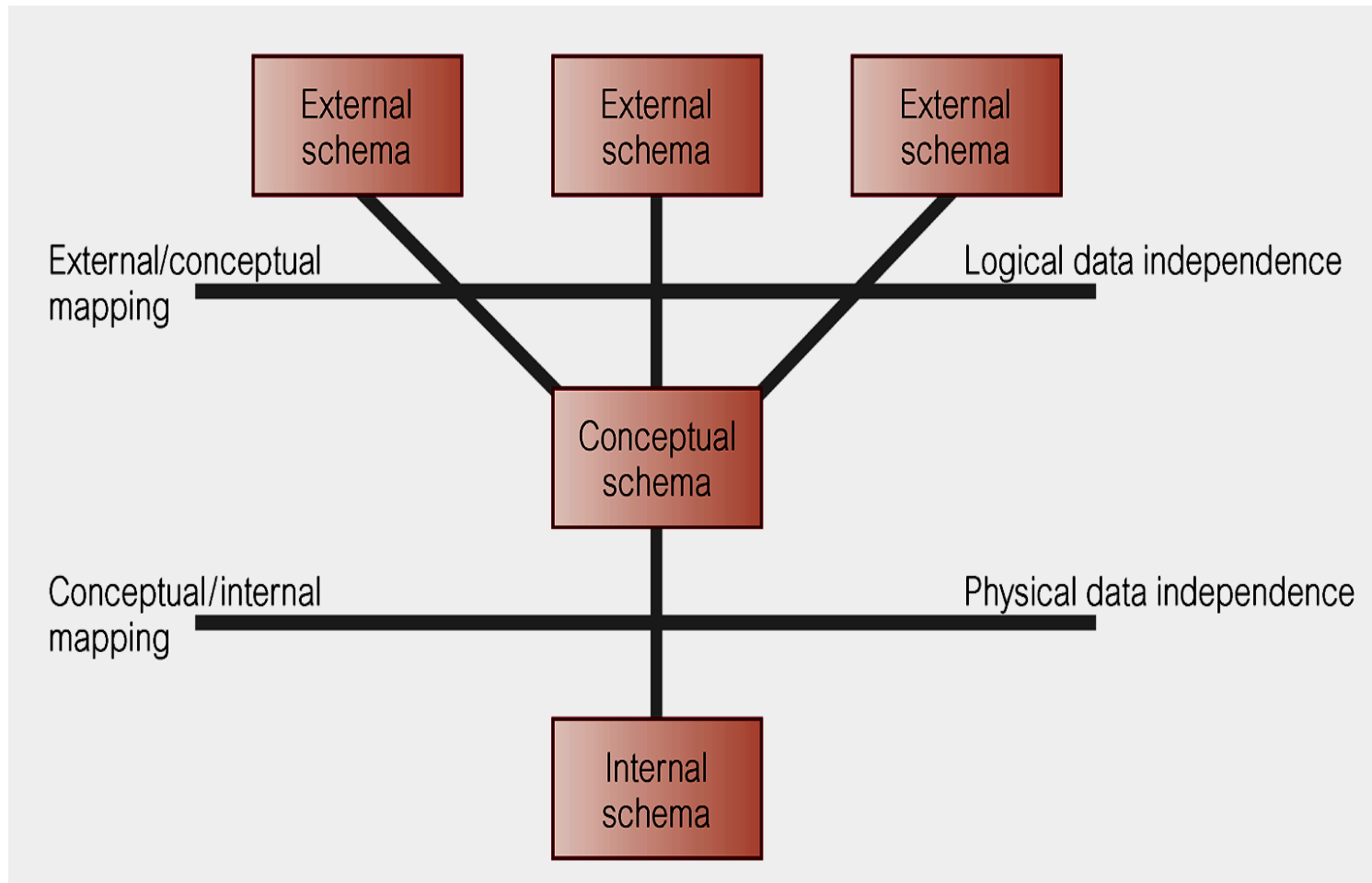
staffNo	fName	lName	DOB	salary	branchNo
---------	-------	-------	-----	--------	----------

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 */

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



Sự độc lập về dữ liệu (Data Independence)

(1)

◆ 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)

(2)

◆ 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

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

Mô hình dữ liệu (Data Model)

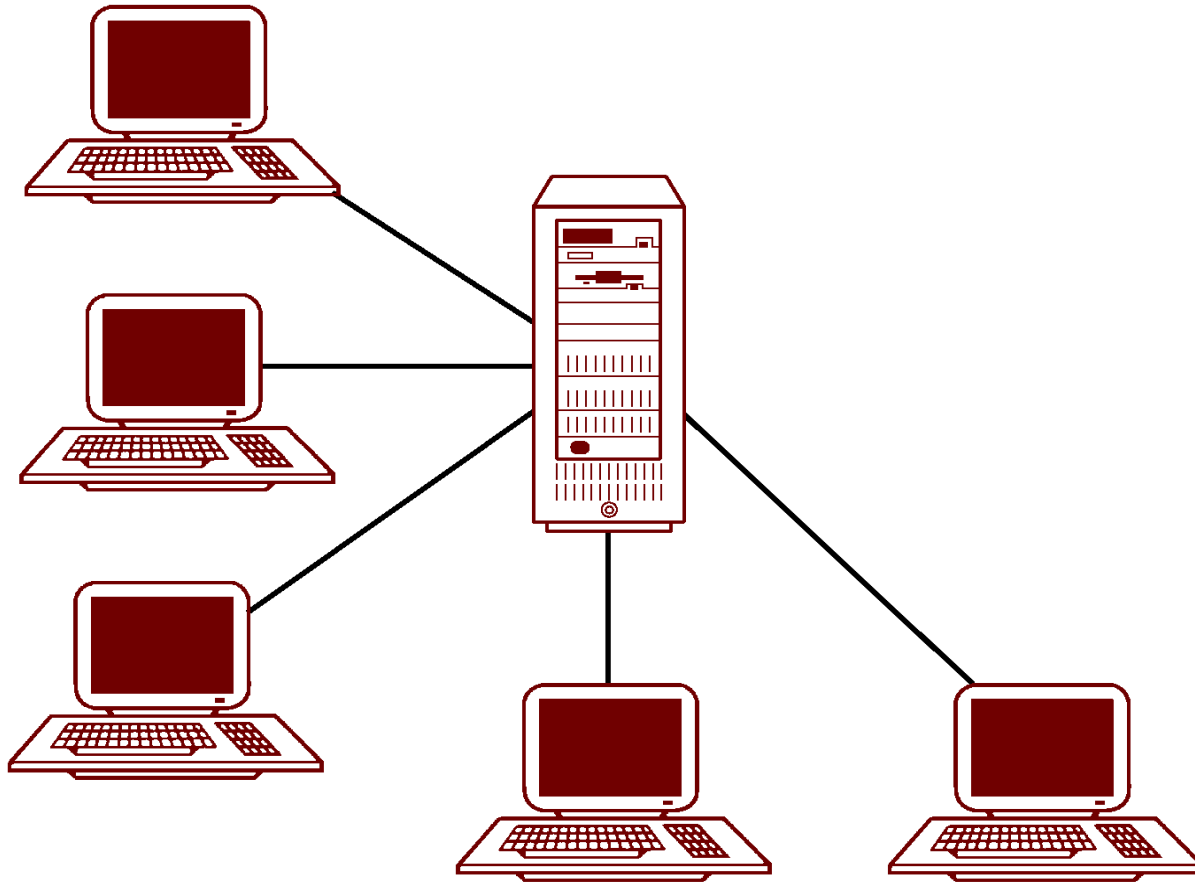
◆ Mục đích

- Biểu diễn dữ liệu theo cách thức dễ hiểu

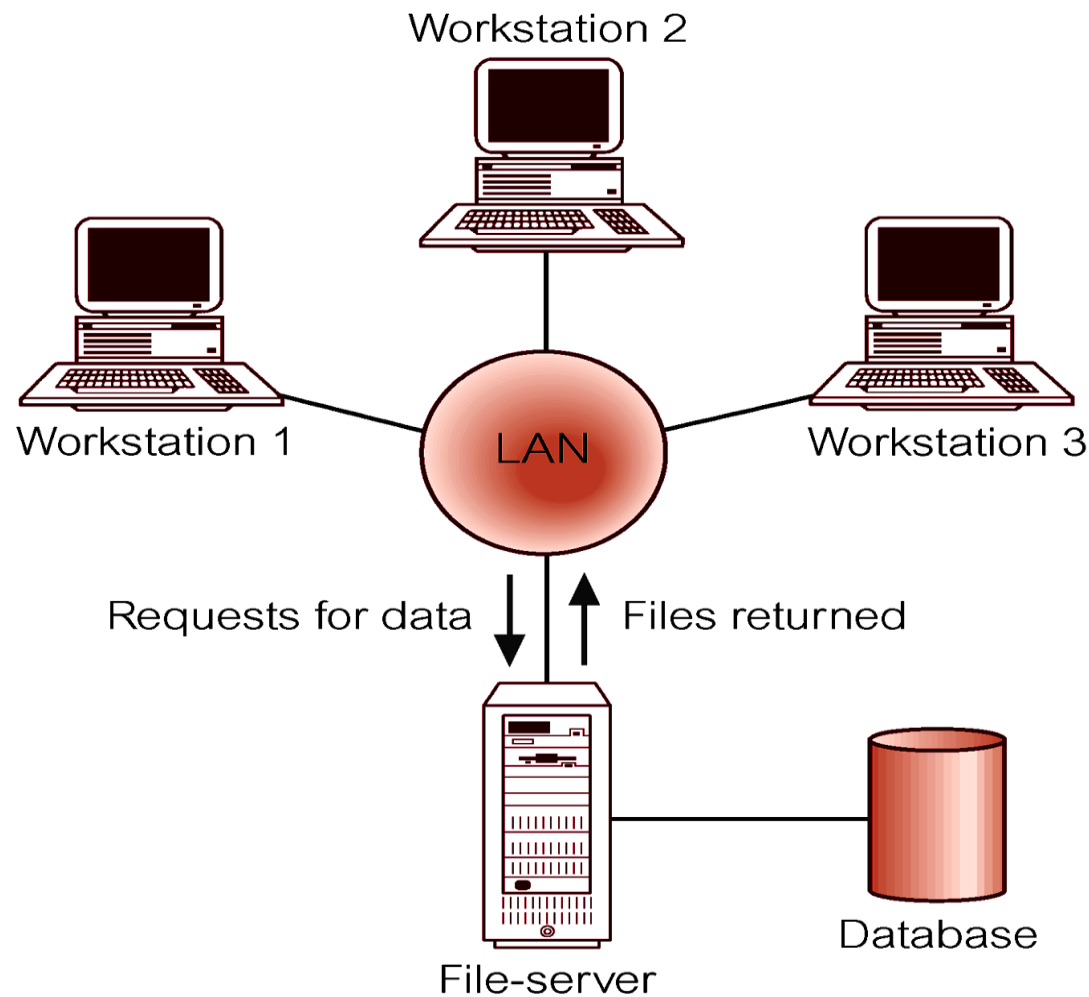
◆ Các loại mô hình dữ liệu gồm:

- Mô hình lấy đối tượng làm nền tảng (Object-based)
- Mô hình lấy record làm nền tảng (Record-based)
- Mô hình vật lý (Physical)

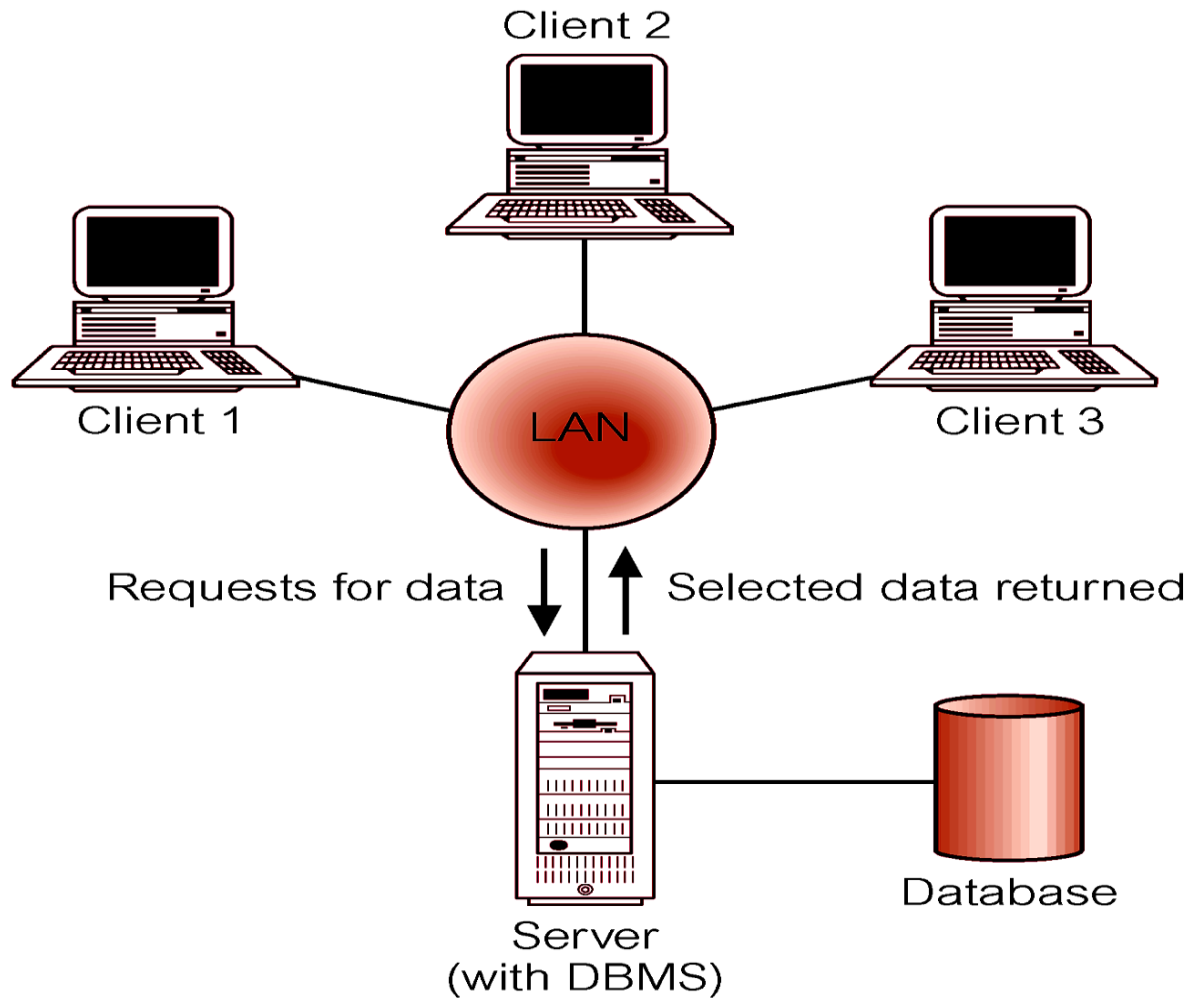
Kiến trúc xử lý từ xa (Teleprocessing)



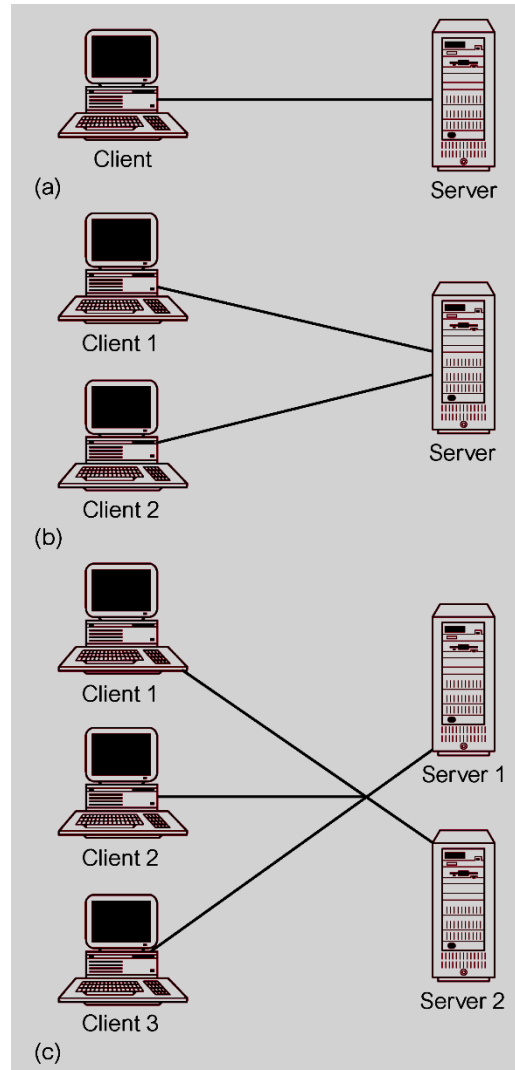
Kiến trúc File-Server



Kiến trúc Client-server



Các sơ đồ kết nối của kiến trúc Client-server



HẾT CHƯƠNG 2

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

Chúc các anh chị học tốt.