

CHƯƠNG II

CÁC VẤN ĐỀ VÀ GIẢI PHÁP CƠ BẢN TRONG CÁC HỆ PHÂN TÁN

NỘI DUNG

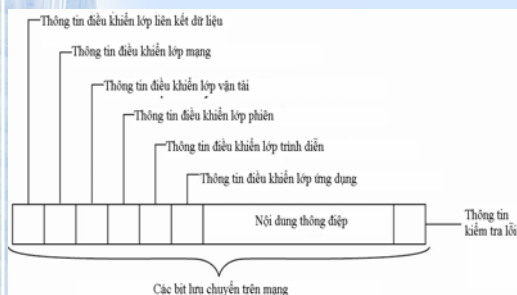
- Truyền thông
- Định danh
- Đồng bộ
- Tiến trình trong các hệ thống phân tán
- Quản trị giao dịch và điều khiển tương tranh
- Phục hồi và chịu lỗi
- Bảo mật
- Tính nhất quán và vấn đề nhân bản

TRUYỀN THÔNG

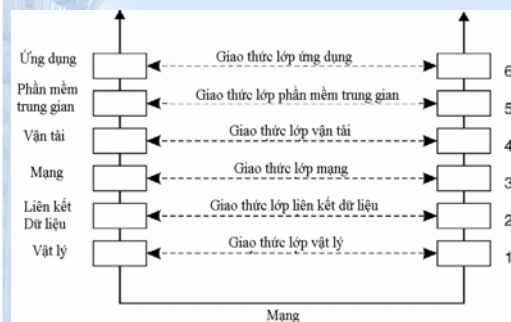
MÔ HÌNH OSI

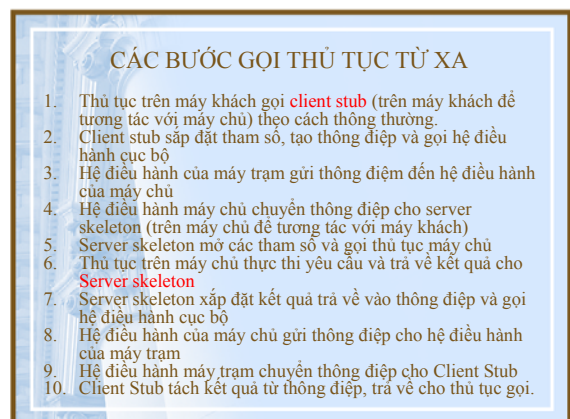
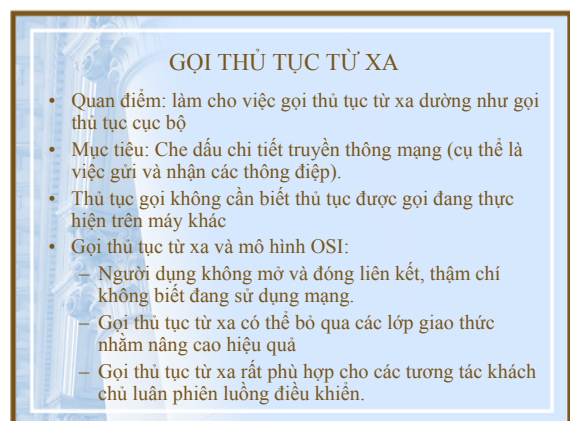
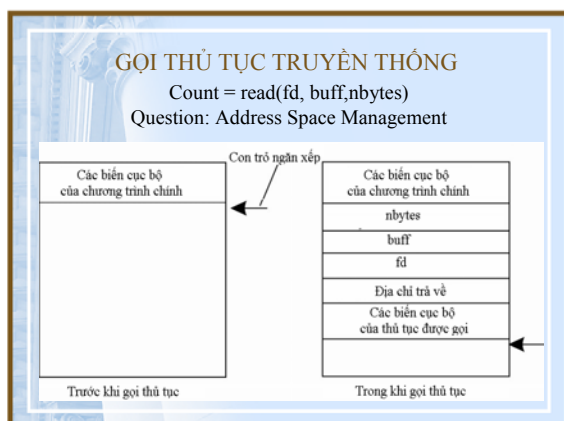
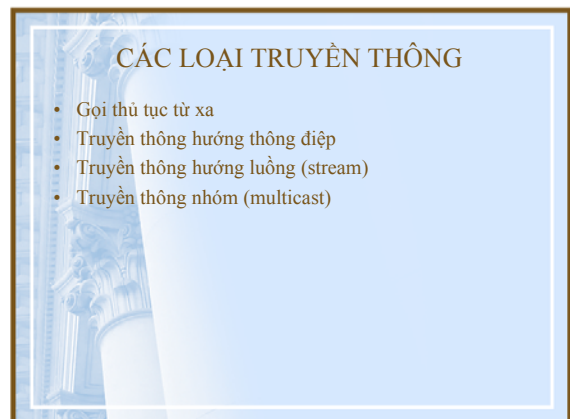
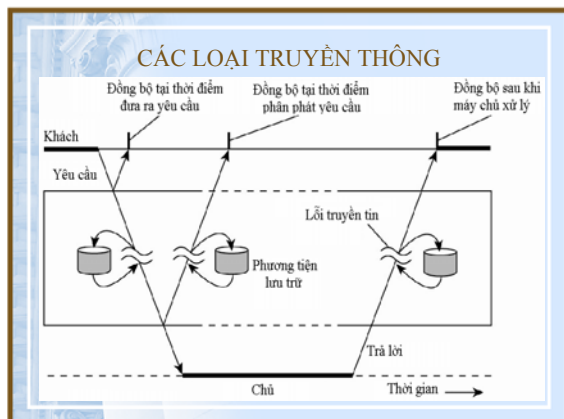
7. Tầng ứng dụng	7. Tầng ứng dụng
Giao diện 7-6	Giao diện 7-6
6. Tầng trình diễn	6. Tầng trình diễn
Giao diện 6-5	Giao diện 6-5
5. Tầng phiên	5. Tầng phiên
Giao diện 5-4	Giao diện 5-4
4. Tầng vận tải	4. Tầng vận tải
Giao diện 4-3	Giao diện 4-3
3. Tầng mạng	3. Tầng mạng
Giao diện 3-2	Giao diện 3-2
2. Tầng liên kết	2. Tầng liên kết
Giao diện 2-1	Giao diện 2-1
1. Tầng vật lý	1. Tầng vật lý

THÔNG điệp ĐIỂN HÌNH XUẤT HIỆN TRÊN MẠNG

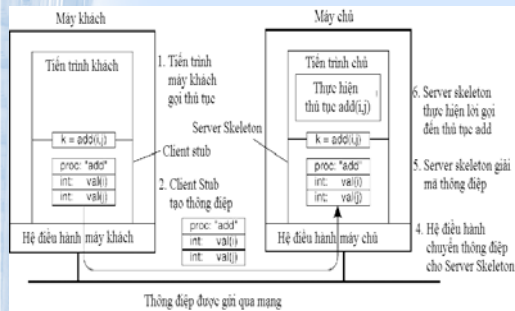


GIAO THỨC PHẦN MỀM TRUNG GIAN





CÁC BƯỚC GỌI THỦ TỤC TỪ XA



TRUYỀN THAM TRỊ

3	2	1	0	0	1	2	3	0	1	2	3
0	0	0	5	5	0	0	0	0	0	0	5
L	L	I	J	J	I	L	L	L	L	I	J

Thông điệp ban đầu trên máy Pentium Thông điệp nhận được trên máy SPARC Thông điệp sau khi đảo ngược

SẮP XẾP THAM SỐ

- Tham số phải được sắp xếp dễ thể hiện theo chuẩn
- Tham số có thể là kiểu đơn (char, int) hoặc kiểu phức hợp (struct) nhưng phân hệ chuyển đổi phải biết kiểu này.
- Không thể thực hiện truyền tham chiếu, có thể mô phỏng bằng cơ chế sao chép-phục hồi. Bản sao cấu trúc dữ liệu tham chiếu được gửi đến máy chủ và khi trả về trả về cho client stub bản sao của cấu trúc sẽ được thay bằng cấu trúc mà máy chủ đã thay đổi.
- Tuy nhiên, việc sắp xếp không thể xử lý trường hợp con trỏ đến bất kỳ cấu trúc dữ liệu nào.

ĐẶC TẢ THAM SỐ VÀ SINH Stub

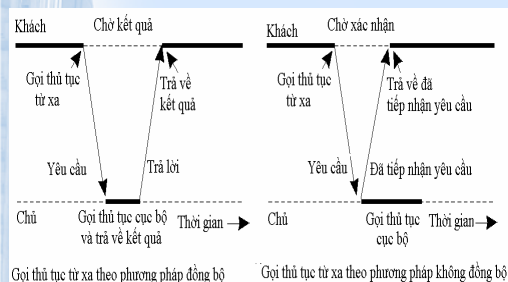
Bên gọi và bên được gọi phải thỏa thuận định dạng của thông điệp sẽ trao đổi và chúng phải tuân thủ các bước giống nhau khi truyền các cấu trúc dữ liệu phức hợp

```

foobar( char x; float y; int z[5] )
{
    ....
}
    
```

Các biến cục bộ của thủ tục foobar(...)	
	x
y	
5	
z[0]	
z[1]	
z[2]	
z[3]	
z[4]	

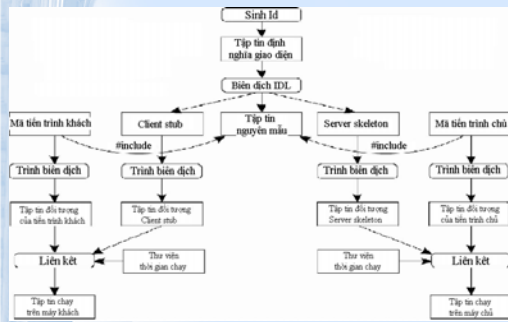
GỌI THỦ TỤC TỪ XA BẰNG PHƯƠNG PHÁP KHÔNG ĐỒNG BỘ



TƯƠNG TÁC GIỮA KHÁCH VÀ CHỦ BẰNG PHƯƠNG PHÁP KHÔNG ĐỒNG BỘ



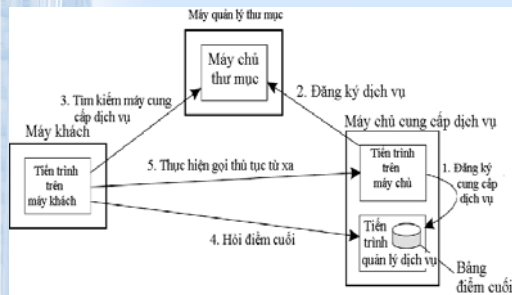
QUI TRÌNH VIẾT ỨNG DỤNG KHÁCH/CHỦ



NHỮNG TIẾN TRÌNH TRÊN MÁY KHÁCH VÀO TIẾN TRÌNH TRÊN MÁY CHỦ

- Đăng ký tiến trình phục vụ của máy chủ để các máy khách có thể xác định được tiến trình phục vụ nhằm mục đích thực hiện thao tác nhúng.
- Việc xác định tiến trình phục vụ được thực hiện qua hai bước:
 - Xác định máy chủ
 - Xác định tiến trình phục vụ trên máy chủ đó

NHỮNG TIẾN TRÌNH TRÊN MÁY KHÁCH VÀO TIẾN TRÌNH TRÊN MÁY CHỦ



TRUYỀN THÔNG HƯỚNG THÔNG ĐIỆP

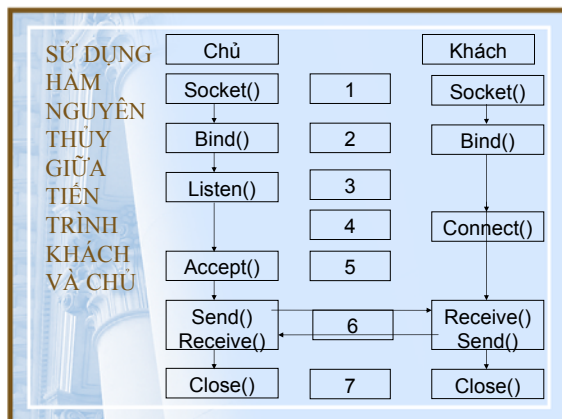
- Gọi thủ tục từ xa che dấu toàn bộ quá trình truyền thông, nhưng sẽ có vấn đề nếu yêu cầu được gửi đến khi bên nhận đang thực thi nhiệm vụ khác. Trong phương pháp đồng bộ, bên gọi bị phong tỏa cho đến khi nhận được kết quả trả về từ phía chủ.
- Hai phương pháp truyền thông hướng thông điệp
 - Truyền thông hướng thông điệp ngắn (transient): Dựa trên mô hình truyền thông hướng thông điệp do lớp vận tải cung cấp
 - Truyền thông hướng thông điệp liên tục (persistent)

BERKELEY SOCKET

- Lần đầu tiên được giới thiệu trong hệ điều hành Berkeley Unix
- Socket là điểm cuối truyền thông mà ở đó các ứng dụng có thể ghi/đọc dữ liệu, phân công việc còn lại do mạng thực hiện hoàn toàn trong suốt với ứng dụng.
- Sử dụng 08 hàm nguyên thủy.
- Mức trừu tượng thấp, được thiết kế để chạy trên các mạng dựa trên ngăn xếp giao thức đa dụng.

CÁC HÀM NGUYÊN THỦY CHO SOCKET

Hàm	Ý nghĩa
Socket	Tạo một điểm cuối truyền thông
Bind	Gán địa chỉ cục bộ vào socket
Listen	Thông báo sẵn sàng tiếp nhận kết nối
Accept	Phong tỏa bên gọi cho đến khi yêu cầu kết nối đến nơi
Connect	Gửi yêu cầu thiết lập kết nối
Send	Gửi dữ liệu trên kết nối đã được thiết lập
Receive	Nhận dữ liệu trên kết nối đã được thiết lập
Close	Đóng và hủy bỏ kết nối

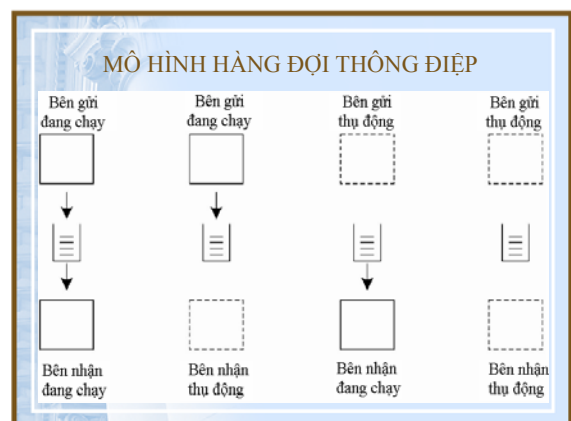


GIAO DIỆN TRUYỀN THÔNG ĐIỆN

- Khắc phục nhược điểm hiệu suất và mức độ trừu tượng thấp bằng cách thêm thư viện truyền thông riêng vào các thiết bị phần cứng.
- Nảy sinh vấn đề không tương thích giữa các phần cứng, do đó cần thiết phải xây dựng giao diện độc lập với phần cứng.
- Dễ dàng viết các ứng dụng hiệu suất cao: mức trừu tượng cao, ít phải gánh thông tin điều khiển
- Sử dụng cho các ứng dụng song song
- Coi lỗi là đương nhiên (mạng hoặc tiến trình), do đó không cần cơ chế phục hồi tự động
- Trao đổi thông tin được thực hiện trong các nhóm của các tiến trình, nguồn và đích được xác định bằng cặp định danh của nhóm và định danh của tiến trình

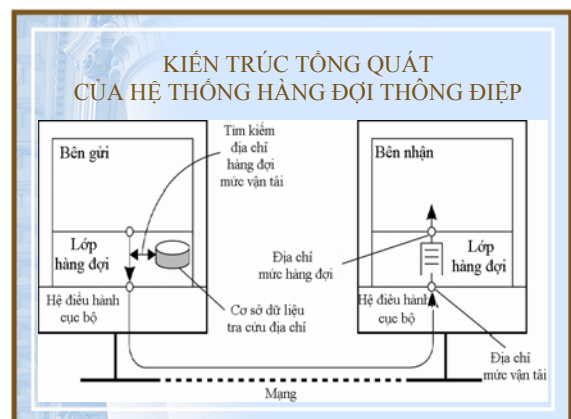
CÁC HÀM NGUYÊN THỦY TRUYỀN THÔNG ĐIỆN

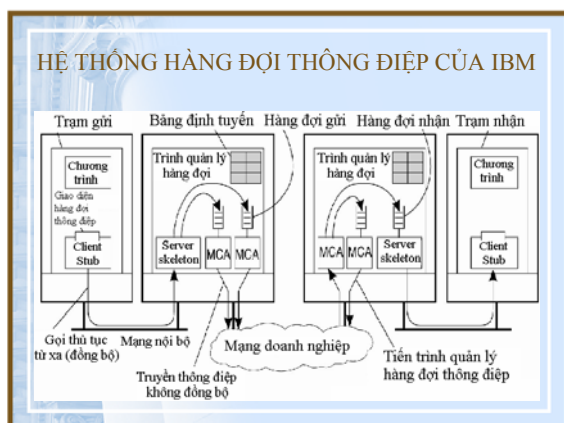
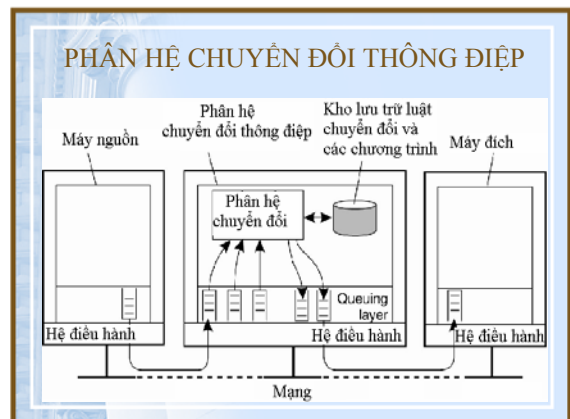
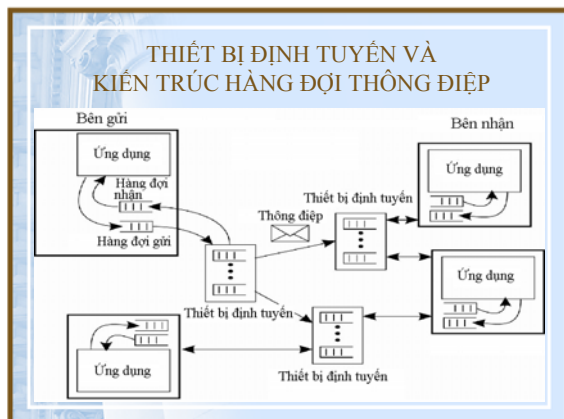
Hàm	Ý nghĩa
MPI_bsend	Thêm thông điệp cần gửi vào vùng đệm gửi cục bộ
MPI_send	Gửi thông điệp và chờ đến khi toàn bộ thông điệp đã ghi vào vùng đệm nhận
MPI_ssend	Gửi thông điệp và chờ đến khi bên nhận thông báo bắt đầu nhận
MPI_sendrecv	Gửi thông điệp và chờ trả lời
MPI_issend	Truyền tham chiếu đến thông điệp cần gửi và tiếp tục
MPI_irecv	Truyền tham chiếu đến thông điệp cần gửi và chờ đến khi bên nhận xác nhận
MPI_recv	Nhận thông điệp, nếu không có thì phóng tỏa
MPI_irecv	Kiểm tra xem có thông điệp đến không, nếu không có thì không phóng tỏa



CÁC HÀM TRONG MÔ HÌNH HÀNG ĐỢI THÔNG ĐIỆN

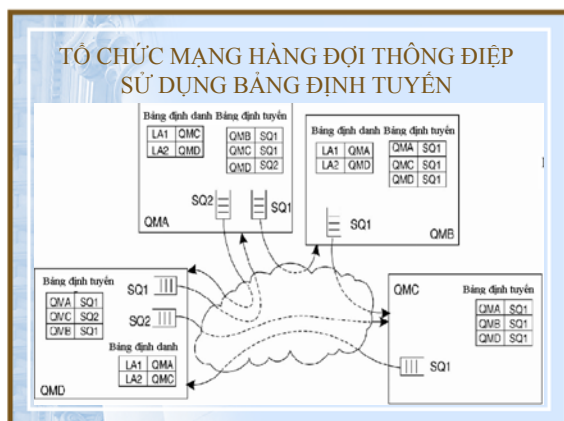
Hàm	Ý nghĩa
Put	Thêm thông điệp vào hàng đợi đã xác định
Get	Phóng tỏa khi hàng đợi còn dữ liệu và loại bỏ thông điệp đầu tiên
Poll	Kiểm tra xem hàng đợi có thông điệp hay không, nếu có thì xóa thông điệp đầu tiên nhưng không phóng tỏa hàng đợi
Notify	Thông báo khi có thông điệp đưa vào hàng đợi





CÁC THUỘC TÍNH LIÊN QUAN TỚI KÊNH THÔNG ĐIỆP

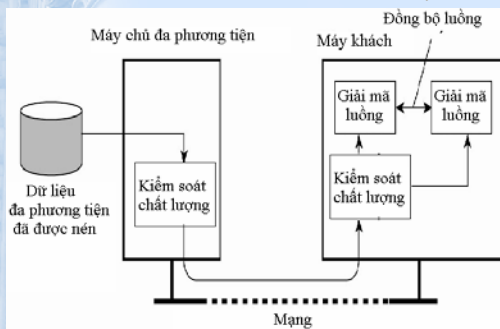
Thuộc tính	Mô tả
Transport type	Quyết định giao thức sử dụng tại lớp vận tải
FIFO delivery	Thông điệp được phân phát theo thứ tự gửi
Message Length	Độ dài tối đa của mỗi thông điệp
Setup retry count	Số lần tối đa kích hoạt lại tác tử kênh thông điệp (MCA)
Delivery retries	Số lần tối đa MCA đặt lại thông điệp vào hàng đợi



CÁC HÀM NGUYÊN THỦY TRONG GIAO DIỆN HÀNG ĐỢI THÔNG ĐIỆP

Hàm	Mô tả
MQopen	Mở hàng đợi
MQclose	Đóng hàng đợi
MQinput	Đưa thông điệp vào hàng đợi
MQget	Lấy thông điệp từ hàng đợi

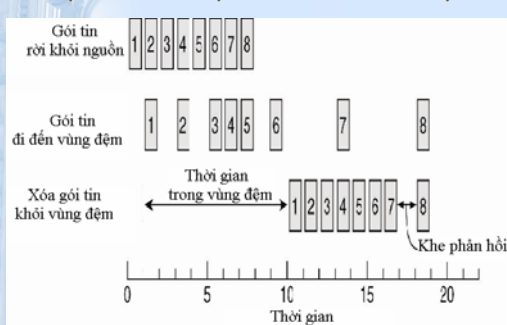
KIẾN TRÚC XỬ LÝ LƯỠNG DỮ LIỆU



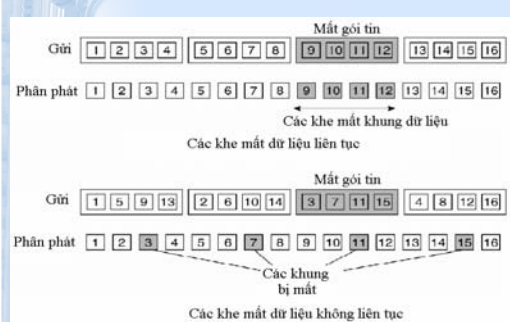
LƯỠNG VÀ CHẤT LƯỢNG DỊCH VỤ

- Tốc độ chuyển dữ liệu (thông lượng)
- Độ trễ tối đa để thiết lập liên kết
- Độ trễ tối đa từ đầu cuối đến đầu cuối
- Biên độ tối đa thay đổi độ trễ
- Độ trễ toàn bộ (từ lúc gửi đi đến khi nhận được phản hồi)

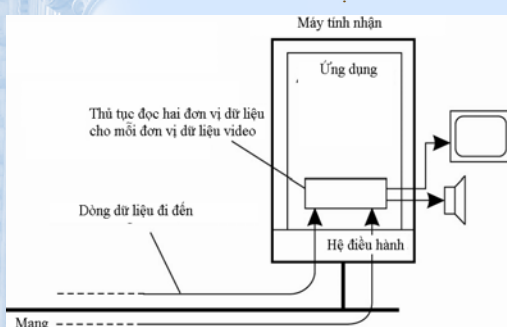
SỬ DỤNG VÙNG ĐỆM ĐỂ GIẢM BIÊN ĐỘ TRỄ



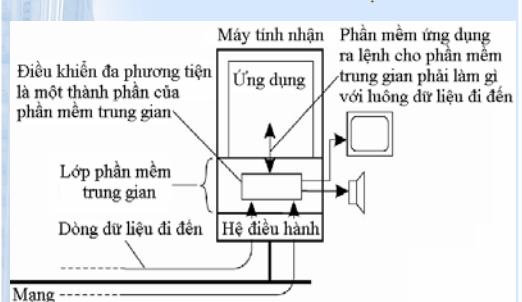
ẢNH HƯỞNG MẤT GÓI TIN



CƠ CHẾ ĐỒNG BỘ



NGUYÊN LÝ ĐỒNG BỘ CÓ SỰ HỖ TRỢ CỦA PHẦN MỀM ỨNG DỤNG



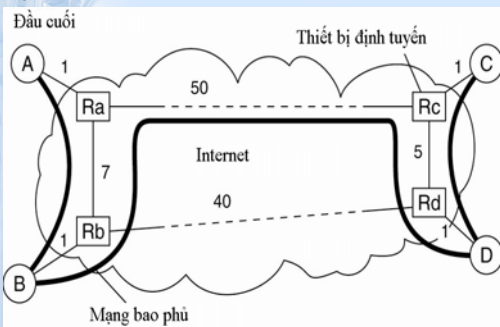
TRUYỀN THÔNG MULTICAST

- Khái niệm: Dữ liệu từ một trạm được gửi đến nhiều trạm
- Vấn đề chính: thiết lập đường truyền thông để phổ biến thông tin
- Các phương pháp:
 - Phổ biến mức ứng dụng
 - Phổ biến dựa trên lan truyền (gossip)

PHỔ BIẾN MỨC ỨNG DỤNG

- Ý tưởng: Các nút được tổ chức thành mạng phủ và dùng để phổ biến thông tin đến các thành viên của mình

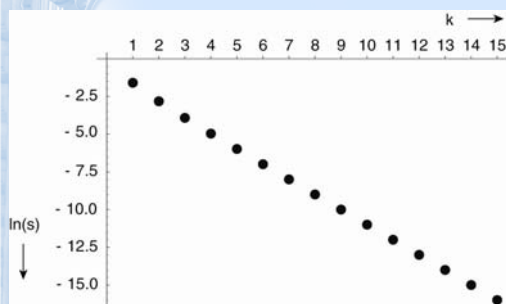
QUAN HỆ GIỮA CÁC KÊNH TRONG MẠNG



MÔ HÌNH TRUYỀN BẮ THÔNG TIN

- Mô hình truyền bá phi entropy
 - Nút P lựa chọn nút khác Q theo thứ tự ngẫu nhiên
 - Sau đó trao đổi cập nhật với Q
- Cách tiếp cận trao đổi cập nhật
 - P chỉ chuyển vào thông tin cập nhật cho Q
 - P chỉ lấy thông tin cập nhật từ Q
 - P và Q trao đổi với nhau

MÔ HÌNH TRUYỀN BẮ THÔNG TIN



BÀI TẬP

- Thiết kế mô hình trao đổi thông tin giữa các thành phần trong bản thiết kế hệ thống quản lý mạng
- Nêu rõ lý do, phân tích vì sao lựa chọn giải pháp trao đổi thông tin