

# CHƯƠNG 1

## TỔNG QUAN VỀ CÁC HỆ THỐNG PHÂN TÁN

### NỘI DUNG

- 1. Hệ thống phân tán là gì
- 2. Phân loại các hệ thống phân tán
- 3. Các đặc trưng và mục tiêu thiết kế cơ bản của các hệ thống phân tán
- 4. Các kiến trúc của các hệ thống phân tán
- 5. Các mô hình căn bản trong các hệ thống phân tán

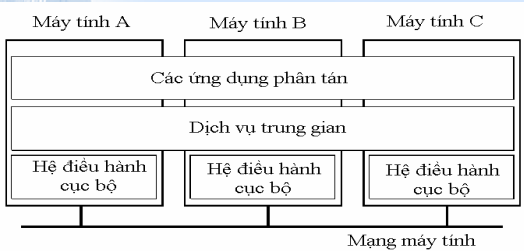
### GIỚI THIỆU HỆ THỐNG PHÂN TÁN

### ĐỊNH NGHĨA HỆ THỐNG PHÂN TÁN

- Khái niệm về hệ thống phân tán có thể hiểu theo nhiều mức độ khác nhau, đơn giản nhất là việc trao đổi thông tin giữa các chương trình sử dụng cơ chế đường ống, cho đến mức độ cao hơn là sự trao đổi thông tin giữa các chương trình trên mạng
  - Hệ thống phân tán là một nhóm các máy tính độc lập nhưng hoạt động gắn kết chặt chẽ với nhau như một hệ thống thống nhất
- Hệ thống phân tán = Mạng + Các dịch vụ hệ thống

### TỔ CHỨC HỆ THỐNG PHÂN TÁN

- Hình thành thêm lớp trung gian cung cấp dịch vụ phân tán:
- Đảm bảo kết nối đến nhiều máy tính
  - Cung cấp giao diện thống nhất cho các ứng dụng lớp trên



### ƯU ĐIỂM HỆ THỐNG PHÂN TÁN

- **Tính kinh tế:** Không cần thiết phải sử dụng các máy tính lớn (mainframes) đắt tiền.
- **Tốc độ:** Tăng thêm sức mạnh tính toán bằng cách phân chia xử lý cho nhiều máy tính.
- **Tin cậy:** nếu một máy gặp sự cố, hệ thống vẫn có thể hoạt động.
- **Chia sẻ dữ liệu:** Nhiều hệ thống dùng chung CSDL.
- **Chia sẻ phần cứng:** Nhiều hệ thống dùng chung một tài nguyên phần cứng.
- **Truyền thông:** giao tiếp giữa các hệ thống dễ dàng hơn.
- **Linh hoạt:** phân chia nhiệm vụ xử lý giữa các máy tính.

NHƯỢC ĐIỂM HỆ THỐNG PHÂN TÁN

- Phần mềm: yêu cầu phần mềm phức tạp hơn
- Mạng: dễ xảy ra nghẽn mạng hoặc có thể phát sinh các vấn đề khác do lỗi mạng
- An toàn và Bảo mật thông tin: Dễ bị mất/sửa thông tin, dễ bị tấn công/đột nhập trên mạng

ĐẶC TRƯNG VÀ MỤC TIÊU THIẾT KẾ  
HỆ THỐNG PHÂN TÁN

MỤC TIÊU XÂY DỰNG HỆ THỐNG PHÂN TÁN

- **Tính thân thiện:** Giao diện người dùng thân thiện. Người dùng ít phải quản lý/kiểm soát tài nguyên hệ thống
- **Hiệu năng:** khắc phục khó khăn do sự chậm trễ thông tin liên lạc
  - Các vấn đề về thiết bị vật lý
  - Phân bổ tải, khắc phục nút cổ chai hoặc tắc nghẽn trong mạng
- **Linh hoạt:** Dễ dàng thay đổi dịch vụ, có khả năng mở rộng và di chuyển, có khả năng module hóa, khả năng tương tác,...
- **Nhất quán:** Thống nhất trong việc sử dụng dịch vụ, dữ liệu trên toàn hệ thống
- **Chịu lỗi:** Tự khôi tạo lại trạng thái tốt nhất và đồng thời đảm bảo mất mát nhỏ nhất, Tự xử lý các trường hợp lỗi
- **An toàn và bảo mật thông tin:** thông tin tin cậy, bảo vệ tốt các điều khiển truy nhập

MONG MUỐN XÂY DỰNG  
HỆ THỐNG PHÂN TÁN

- **Tính trong suốt:** Che dấu việc xử lý
- **Toàn vẹn:** Thống nhất trong toàn bộ hệ thống
- **Hiệu năng:** Tốc độ xử lý dịch vụ nhanh
- **Chịu lỗi:** Vẫn hoạt động ngay cả khi có lỗi
- **Tính mở:** Giao tiếp với các hệ thống khác
- **Bảo trì/nâng cấp:** Dễ dàng nâng cấp/thay thế phần cứng/Phần mềm. Đáp ứng mở rộng hệ thống, dịch vụ

TÍNH TRONG SUỐT

- Tính trong suốt là sự che dấu của hệ thống với người sử dụng
- Tính trong suốt tạo cho người sử dụng khả năng dễ thao tác hơn trên hệ thống. Tạo ra một cách nhìn riêng biệt đối với người dùng
- Tính trong suốt có thể coi như một giao diện chuyển đổi mà ở đó người sử dụng sẽ thao tác với hệ thống bằng bộ lệnh được chuẩn hoá. Các lệnh này qua giao diện chuyển đổi nói trên sẽ thao tác đến các đối tượng phù hợp trong hệ thống.
- Cân bằng giữa hiệu năng và sự đơn giản của hệ thống
- Tính trong suốt là nhân tố chính trong việc thiết kế phần mềm hệ thống phân tán.
- Mục tiêu cần đạt:
  - Hệ thống được nhận biết như một thể thống nhất chứ không phải là tập hợp của nhiều thành phần độc lập
  - Cung cấp một cách nhìn đơn giản đối với hệ thống
  - Giảm ảnh hưởng của cấu trúc vật lý của hệ thống

CÁC ĐẶC TÍNH TRONG SUỐT

Tính trong suốt	Mô tả
Truy nhập	Che dấu cách thể hiện và truy nhập dữ liệu
Vị trí	Che dấu nơi lưu trữ dữ liệu. Che dấu việc di chuyển dữ liệu.
Nhân bản	Che dấu việc nhân bản dữ liệu
Tương tranh	Cho phép nhiều người có thể đồng thời sử dụng dữ liệu, hạn chế tối đa xung đột (dẫn đến deadlock)
Lỗi	Che dấu lỗi xảy ra trong hệ thống
Di chuyển	Che dấu việc di chuyển dữ liệu nhưng vẫn đảm bảo dịch vụ
Hiệu năng	Cho phép cấu hình lại hệ thống để tăng hiệu năng
Đồng nhất	Tài nguyên có thể được lưu trữ trong bộ nhớ trong (RAM) hoặc trên thiết bị nhớ ngoài (đĩa cứng...)
Qui mô	Người dùng không cảm nhận được qui mô của hệ thống

### CÁC ĐẶC TÍNH TRONG SUỐT

- *Trong suốt truy cập:* cho phép các đối tượng cục bộ, hay từ xa đều được truy cập bằng các hoạt động giống nhau.
- *Trong suốt vị trí:* các đối tượng sử dụng dịch vụ không cần biết vị trí cài đặt của các đối tượng cung cấp dịch vụ.
- *Trong suốt nhân bản:* cho phép nhiều bản sao của các đối tượng được sử dụng để tăng độ tin cậy và hiệu năng của hệ thống mà không cho người sử dụng thấy được sự nhân bản đó.
- *Trong suốt tương tranh:* cho phép nhiều xử lý được thực hiện đồng thời sử dụng thông tin chia sẻ mà không xảy ra xung đột.
- *Trong suốt lỗi:* cung cấp khả năng chịu lỗi để người sử dụng có thể hoàn thành được công việc ngay cả khi hệ thống gặp lỗi về phần cứng hoặc phần mềm.

### CÁC ĐẶC TÍNH TRONG SUỐT

- *Trong suốt di chuyển:* cho phép di chuyển các đối tượng trong hệ thống mà không ảnh hưởng đến hoạt động của người sử dụng. Khả năng di chuyển thường được thể hiện khi các đối tượng thành phần trong hệ thống chuyển dịch vị trí khi chuyển sang cơ chế dự phòng để khắc phục lỗi tại máy tính chính, hoặc nhằm mục đích cân bằng tải.
- *Trong suốt hiệu năng:* cho phép cấu hình lại hệ thống để tăng hiệu năng.
- *Đồng nhất:* Dữ liệu được đồng nhất giữa bộ nhớ và thiết bị lưu trữ ngoài
- *Trong suốt quy mô:* cho phép các thành phần trong hệ thống có thể thay đổi quy mô mà không ảnh hưởng đến cấu trúc của hệ thống cũng như hoạt động của hệ thống.

### TOÀN VỆN

- Truy nhập hệ thống
- Nhân bản dữ liệu
- Lưu trữ dữ liệu

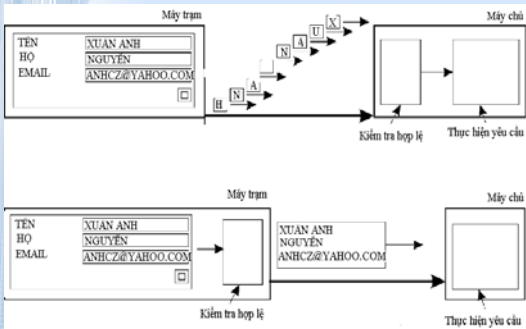
### HIỆU NĂNG

- Các vấn đề phát sinh khi qui mô tăng lên
  - Dịch vụ tập trung: một máy chủ cho tất cả người sử dụng
  - Dữ liệu tập trung: sử dụng duy nhất một cổng truy nhập
  - Thuật toán tập trung: Kết quả thực hiện thuật toán trên một máy chủ dựa trên quá nhiều thông tin từ các máy chủ khác
- Giải pháp xử lý:
  - Xử lý tương tranh
  - Xử lý song song
  - Tìm ra điểm tắc nghẽn (nút cổ chai): Xem xét thời gian thực hiện của từng thủ tục/dịch vụ
  - Phân tải xử lý

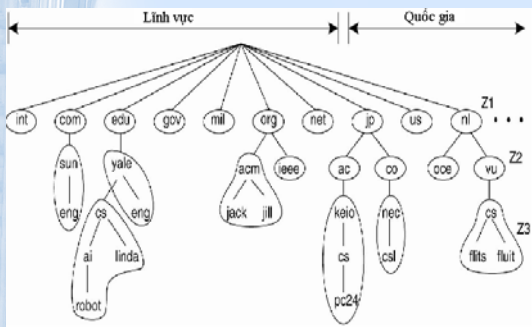
### ĐẶC ĐIỂM THUẬT TOÁN PHI TẬP TRUNG

- Không một máy tính nào có toàn bộ thông tin về trạng thái của hệ thống
- Các máy ra quyết định chỉ dựa trên thông tin cục bộ.
- Nếu có lỗi xảy ra trên một máy nào đó sẽ không ảnh hưởng đến thuật toán.
- Không thể giả thiết có sự tồn tại của đồng hồ thời gian chung cho toàn bộ hệ thống.

### KỸ THUẬT XỬ LÝ QUI MÔ LỚN



KỸ THUẬT XỬ LÝ QUI MÔ LỚN



TÍNH CHỊU LỖI  
(Fault tolerance)

- Lỗi: Kết quả ngoài mong muốn của người dùng.
- Nguyên nhân:
  - Phần cứng
  - Phần mềm
  - Mạng
- Tính chịu lỗi của hệ thống phân tán luôn là vấn đề được quan tâm khi thiết kế, xây dựng và triển khai hệ thống

TIÊU CHÍ ĐÁNH GIÁ TÍNH CHỊU LỖI

- Khả năng phát hiện lỗi: Bao gồm việc phát hiện ra lỗi, xác định hoặc chẩn đoán lỗi và gửi bản tin cảnh báo đến quản trị hệ thống.
- Lưu vết: Trong mọi trường hợp phải lưu lịch sử các giao dịch, các lỗi xảy ra trong hệ thống
- Khả năng tự sửa lỗi:
  - Các lỗi được phát hiện, và được định danh sẽ được sửa chữa.
  - Các lỗi được chẩn đoán (có thể không chính xác) sẽ được tìm kiếm các giải pháp có thể để giúp người sử dụng lựa chọn phương pháp phục hồi.
  - Các lỗi không thể chẩn đoán sẽ được cảnh báo để người sử dụng có biện pháp xử lý, đồng thời phục hồi về trạng thái ban đầu như trước khi tiến hành xử lý.

GIẢI PHÁP HẠN CHẾ LỖI

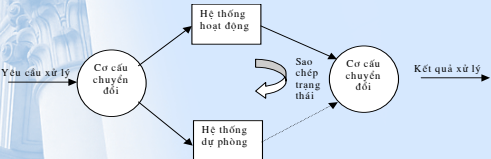
- *Tăng độ dự thừa tài nguyên phần cứng*: Hệ thống sẽ dự phòng nhiều thiết bị phần cứng ngoài các thiết bị yêu cầu cần có. Khi gặp lỗi, hệ thống sẽ tự động chuyển các xử lý tại module đang hoạt động trên các trạm có lỗi sang các module dự phòng trên các trạm dự phòng. Khi đó, hệ thống vẫn đảm bảo hoạt động bình thường.
- *Tăng khả năng phục hồi cho hệ thống phần mềm*: Hệ thống phần mềm phải đảm bảo có thể phát hiện lỗi, sửa chữa lỗi hoặc ít nhất tự phục hồi về trạng thái ban đầu trước khi tính toán. Khả năng phục hồi phải đảm bảo về tốc độ để hệ thống có thể vận hành bình thường.

CÁC GIẢI PHÁP DỰ PHÒNG

- *Dự phòng nguội*:
  - Sử dụng thiết bị thay thế
  - Bình thường không làm gì hoặc thực hiện một xử lý khác. Khi sự cố xảy ra thì chuyển sang hệ thống dự phòng.
  - Hệ thống sẽ tạm ngừng hoạt động trong khoảng thời gian thay thế nên sẽ gây ra tổn thất không thể lường trước.
- *Dự phòng nóng*: đảm bảo thời gian cho phép làm hệ thống không bị gián đoạn.
  - Dự phòng kép đồng bộ
  - Dự phòng ghép song song

DỰ PHÒNG KÉP ĐỒNG BỘ

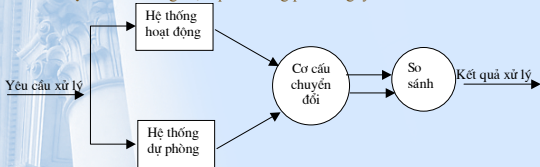
- **Nguyên lý**:
  - Hai hệ thống hoạt động độc lập với nhau và thường xuyên trao đổi các thông tin điều khiển hệ thống nhưng tại một thời điểm chỉ có một hệ thống cung cấp dịch vụ cho các đối tượng trên mạng
  - Khi sự cố xuất hiện thì chuyển từ hệ thống hoạt động sang hệ thống dự phòng và trạng thái cuối cùng được sao chép sang
- **Ưu điểm**: Không cần phải thay thế trạng thái khi có sự cố do đó việc chuyển hệ thống được thực hiện trong thời gian rất ngắn. Thuật toán điều khiển (phần mềm) khá đơn giản.
- **Nhược điểm**: Hiệu suất sử dụng thấp và giá thành cao.





## DỰ PHÒNG GHEP SONG SONG

- **Nguyên lý:**
  - Các quá trình được thực hiện song song ở cả hệ thống hoạt động và hệ thống dự phòng.
  - Khi có sự cố thì không cần phải thay thế trạng thái do đó việc chuyển hệ thống được thực hiện trong thời gian rất ngắn.
  - Thường thì loại hệ thống dự phòng này rất ít khi được sử dụng trong hệ thống phân tán
- **Ưu điểm:** Hệ thống không phức tạp vì hai hệ thống hoạt động song song.
- **Nhược điểm:** không hiệu quả và lãng phí tài nguyên



## CÁC HOẠT ĐỘNG DỰ PHÒNG NÓNG

- **Phát hiện lỗi:** Tiến trình phát hiện kịp thời sự xuất hiện lỗi.
- **Chuẩn đoán lỗi:** Tiến trình phát hiện nguyên nhân gây ra lỗi hoặc chính xác phân hệ hoặc thành phần nào lỗi.
- **Chính sách ngăn chặn lỗi:** Tiến trình bảo vệ sự lan truyền lỗi từ lỗi ban đầu tại một điểm trong hệ thống tới một điểm khác mà ở đó nó có thể có ảnh hưởng tới dịch vụ cung cấp cho người sử dụng.
- **Che lỗi:** Tiến trình đảm bảo rằng chỉ các giá trị đúng nhận truyền tới ranh giới cho phép.
- **Bù lỗi:** Nếu lỗi xuất hiện và liên quan với phần hệ thì nó cần thiết cho hệ thống cung cấp một phúc đáp để bù lại đầu ra của phần hệ lỗi.
- **Sửa lỗi:** Tiến trình mà các lỗi được di chuyển trong hệ thống. Trong hệ thống dự phòng nóng thiết kế tốt thì các lỗi được lưu trữ trước khi chúng truyền tới một phạm vi phân chia dịch vụ hệ thống tác động.

## TÍNH MỞ

- Tính mở của một hệ thống máy tính là đặc điểm để xác định xem liệu hệ thống có thể mở rộng bằng nhiều cách được không.
- Khi xây dựng các hệ thống theo chuẩn công nghiệp, và nhất là các hệ thống phân tán, tính mở luôn được đề cập như một yếu tố không thể thiếu khi thiết kế hệ thống.
- Một hệ thống có mở rộng được hay không thường do hệ thống phần mềm hỗ trợ, quản lý và điều khiển (phần mềm ở đây bao gồm cả software và firmware).
- Trong hệ phân tán, người ta đưa ra một giao diện chuẩn và coi như giao diện phần mềm chia khoá để các nhà phát triển phần mềm tuân thủ khi thiết kế. Tính mở còn được thực hiện thông qua việc cung cấp cơ chế truyền thông chuẩn giữa các tiến trình. Cơ chế này đảm bảo các ứng dụng khác nhau có thể tuân thủ để giao tiếp với nhau mà không phụ thuộc nền, hay phụ thuộc môi trường phát triển ứng dụng.

## TÍNH MỞ

Trong công nghệ phần mềm, một hệ thống được xem là có tính mở nếu hệ thống đó đáp ứng được 2 tính chất sau:

- **Tính dễ tương tác:** Một hệ thống được gọi là dễ tương tác với hệ thống khác nếu hệ thống đó có thể làm việc chung trong một môi trường với các hệ thống khác và hệ thống đó có thể xử lý được các đối tượng do hệ thống khác phát sinh và ngược lại
- **Tính dễ chuyển mạng:** Một hệ thống được gọi là dễ chuyển mạng nếu nó có thể làm việc trong nhiều môi trường khác nhau và trên nhiều chủng loại thiết bị khác nhau, nói cách khác: hệ thống không kén chọn môi trường, thiết bị.

## BẢO TRÌ HỆ THỐNG

- Phân loại theo tần suất và độ quan trọng của sự truy nhập, xác định người chịu trách nhiệm và chu kỳ làm việc.
- Nếu dữ liệu được phân tán, phương pháp điều khiển sẽ phức tạp, vì vậy những quy tắc trong công việc phải rõ ràng.
- Sao lưu dữ liệu qua kênh liên lạc sẽ làm tăng lưu lượng (mạng + RAM + CPU), do vậy cần hạn chế tối đa các thao tác này (lựa chọn thời điểm sao lưu, chiến lược sao lưu)
- Định kỳ dọn dẹp hệ thống: Loại bỏ những dữ liệu không cần thiết làm giảm hiệu năng hoạt động của hệ thống

## SỰ CẦN THIẾT XÂY DỰNG HỆ THỐNG PHÂN TÁN

- **Yêu cầu tính toán phân tán:** tận dụng khả năng tính toán song song hoặc sử dụng khả năng tính toán của các máy tính chuyên dụng.
- **Yêu cầu xử lý phân tán:** Ứng dụng chạy trên nhiều máy tính khác nhau mới có thể đáp ứng được yêu cầu bài toán.
- **Yêu cầu tính chính xác và an toàn dữ liệu:** các hệ thống cần phải đảm bảo tính chính xác cao và an toàn dữ liệu lớn. Các hệ thống này thường được thiết kế để có thể đáp ứng được yêu cầu tính toán ngay cả khi có sự cố xảy ra, điều này được thực hiện bằng cách tăng số lần tính toán cho cùng một nhiệm vụ, tăng số ứng dụng chạy dự phòng nhằm mục đích kịp thời phát hiện và xử lý lỗi.
- **Chia sẻ tài nguyên:** Các ứng dụng chạy trên mạng thực hiện trao đổi thông tin với nhau, chia sẻ tài nguyên của mình cho các ứng dụng khác. Một số ứng dụng phải chạy trên nhiều máy tính vì dữ liệu được đặt phân tán trên mạng liên quan đến quyền quản lý và quyền sở hữu dữ liệu: cho phép truy nhập và sử dụng theo quyền được cho phép.

NHỮNG QUAN NIỆM CẦN TRÁNH

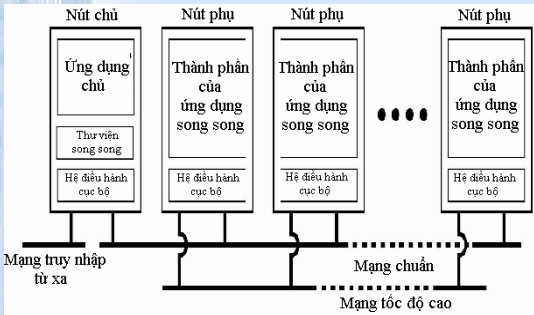
- Mạng đáng tin cậy
- Mạng đã được bảo mật
- Mạng là hệ thống đồng nhất
- Hình trạng mạng không bao giờ thay đổi
- Độ trễ bằng không
- Băng thông vô hạn
- Chi phí vận chuyển bằng không
- Không có người quản trị.

PHÂN LOẠI HỆ THỐNG PHÂN TÁN

PHÂN LOẠI HỆ THỐNG PHÂN TÁN

- Các hệ thống tính toán phân tán
  - Hệ thống tính toán cụm
  - Hệ thống tính toán lưới
- Các hệ thống thông tin phân tán:
  - Cơ sở dữ liệu phân tán
  - Các hệ tính toán phân tán
  - Các ứng dụng phân tán, ...

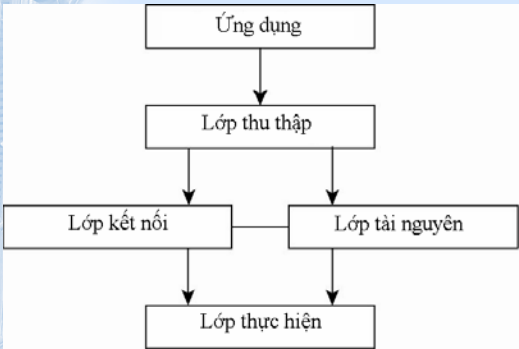
CÁC HỆ THỐNG TÍNH TOÁN CỤM



ĐẶC ĐIỂM HỆ THỐNG TÍNH TOÁN CỤM

- Hiệu quả kinh tế (thay cho việc đầu tư 01 máy tính lớn có giá thành cao)
- Sử dụng kỹ thuật lập trình song song
- Thường đòi hỏi hệ thống đồng nhất (Hệ điều hành, mạng...)

HỆ THỐNG TÍNH TOÁN LƯỚI



ĐẶC ĐIỂM HỆ THỐNG TÍNH TOÁN LƯỚI

- Không yêu cầu tính đồng nhất của hệ thống (Mạng, hệ điều hành, chính sách quản lý)
- Tổ chức ảo: Tài nguyên của nhiều tổ chức khác nhau được đưa về phục vụ cho việc cộng tác giữa các nhóm.
- Phần mềm điều khiển tổ chức ảo được phân thành 4 lớp:
  - Lớp thực hiện: Cung cấp giao diện đến tài nguyên cục bộ, cho phép chia sẻ tài nguyên bên trong tổ chức ảo
  - Lớp kết nối: Cung cấp các giao thức hỗ trợ giao tác lưới để có thể bao trùm việc sử dụng nhiều loại tài nguyên
  - Lớp tài nguyên: Quản lý từng tài nguyên, sử dụng thông tin do lớp kết nối cung cấp để truy nhập vào lớp thực hiện.
  - Lớp thu thập: Quản lý truy nhập đến nhiều tài nguyên: Khám phá, định vị, lập lịch thực hiện, nhân bản...
  - Lớp ứng dụng: Cung cấp tiện ích vận hành trong tổ chức ảo
- Trong hệ thống phân tán, các lớp (Kết nối, tài nguyên, Thu thập) được gọi là phần mềm trung gian (Middleware)

HỆ THỐNG THÔNG TIN PHÂN TÁN

- Dữ liệu được lưu trữ tại nhiều nơi
- Tại mỗi vị trí, dữ liệu được quản lý (thường sử dụng hệ quản trị CSDL) và sẵn sàng cung cấp cho các máy tính khác (máy trạm hoặc máy chủ).
- Mỗi thao tác thay đổi dữ liệu gọi là giao tác.
- Các yêu cầu đối với hệ thống thông tin phân tán:
  - Bao quát các thay đổi theo ngữ cảnh
  - Khuyến khích bố cục cơ sở
  - Thừa nhận mặc nhiên phải chia sẻ dữ liệu

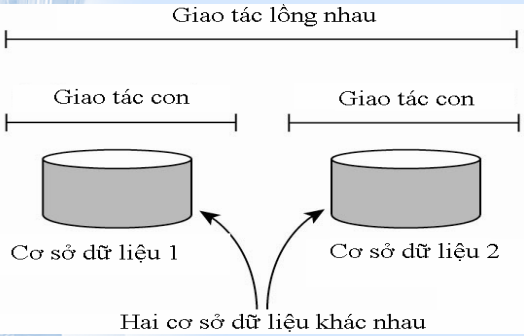
CÁC HÀM NGUYÊN THỦY TRONG GIAO TÁC

- BEGIN TRANSACTION: Đánh dấu bắt đầu phiên giao tác
- END TRANSACTION: Kết thúc phiên giao tác và cam kết những lệnh thực hiện trước đó là đúng (COMMIT), hệ thống sẽ cập nhật dữ liệu mới nhất theo kết quả của các lệnh đã thực hiện.
- ABORT TRANSACTION: Kết thúc phiên giao tác nhưng hủy bỏ các lệnh đã thực hiện trước đó (ROLLBACK), phục hồi lại dữ liệu như chưa từng thực thi lệnh nào
- READ: Đọc dữ liệu từ tập tin, bảng...
- WRITE : Ghi dữ liệu vào tập tin, bảng...

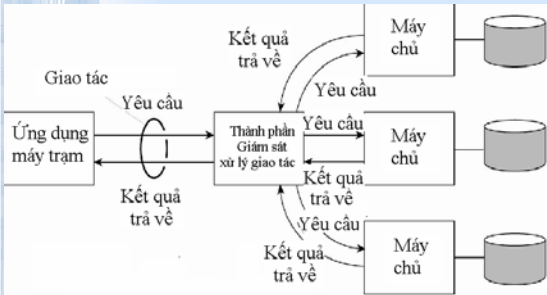
CÁC THUỘC TÍNH ĐẶC TRƯNG CỦA GIAO TÁC

- Nguyên tử (Atomic): Đối với thế giới bên ngoài thì giao tác không thể chia nhỏ hơn được
- Nhất quán (Consistent): Giao tác không vi phạm tính bất biến hệ thống.
- Cách ly (Isolated): Các giao tác đồng thời thực hiện cùng một lúc không ảnh hưởng lẫn nhau.
- Có thời hạn (Durable): Khi đã xác nhận thay đổi thì những thay đổi đó là vĩnh cửu (Bền vững).

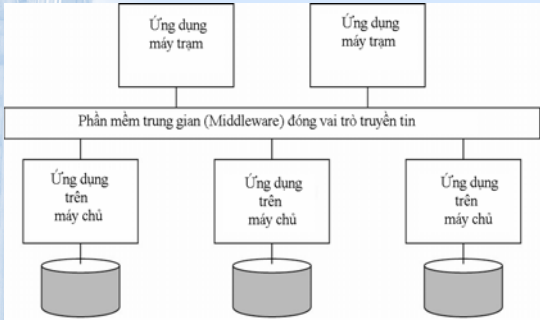
GIAO TÁC LỒNG NHAU



VAI TRÒ CỦA GIÁM SÁT GIAO TÁC TRONG CÁC HỆ THỐNG PHÂN TÁN



### SỬ DỤNG PHẦN MỀM TRUNG GIAN



### LỰA CHỌN TIÊU CHÍ PHÂN TÁN

- **Phân tán theo chức năng:** Các máy tính cung cấp các chức năng cụ thể (CSDL, Xác thực, chuyển tiếp, chuyên dụng...)
- **Phân tán theo nhu cầu phân tải xử lý:** tải được phân chia tới các máy tính được kết nối với nhau nhằm phản ứng linh hoạt với sự tăng/giảm tải.
- **Phân tán để đề phòng các sự cố:**
  - Rủi ro được phân tán ra cho nhiều máy tính.
  - Nếu lỗi xảy ra trọng một máy tính, dịch vụ sẽ kèm đi, nhưng quá trình xử lý có thể tiếp tục mà không dừng toàn bộ hệ thống.
  - Hệ thống phân tán sẽ có cơ chế phòng ngừa sự xâm nhập bên ngoài dễ hơn.
- **Phân tán dữ liệu:**
  - Phân quyền truy nhập dữ liệu theo loại ứng dụng, đối tượng sử dụng và các đặc tính dữ liệu.
  - Dữ liệu được phân tán cho từng địa điểm mà ở đó nó sẽ được sử dụng, như vậy hiệu suất đáp ứng sẽ tốt và chi phí truyền dẫn sẽ giảm.. Việc quản lý bảo dưỡng dữ liệu sẽ phức tạp.

### PHÂN TÁN THEO CHỨC NĂNG

- Mạng có hiệu suất như một máy tính lớn nhưng đạt được bởi việc kết hợp các máy tính nhỏ, giá thành thấp, như vậy đạt hiệu suất cao về chi phí.
- Các chức năng cụ thể được xác định rõ ràng cho từng máy tính đơn lẻ, như vậy tổng chi phí cho hệ thống sẽ giảm và khả năng xử lý sẽ cao như ở một hệ thống trọn vẹn.
- Khả năng mở rộng và độ linh hoạt của hệ thống cao.
- Tải tính toán được chia đều cho từng máy tính để làm cho việc phân chia chức năng trở nên phù hợp nhất.

### PHÂN TÁN THEO NHU CẦU PHÂN TẢI

- **Hệ thống phân tải tĩnh:**
  - Các quá trình xử lý được phân chia tới các máy tính mà chức năng của chúng đã được xác định trước.
  - Hệ thống phân tải loại này không thể phản ứng linh hoạt đối với những thay đổi tăng, giảm số lượng các giao dịch.
- **Hệ thống phân tải động:** Tải trên tài nguyên của từng hệ thống được xác định và việc phân chia tải là động. Kiểu phân tải này cần có sự phân chia điều khiển nhiệm vụ tùy theo trạng thái của tải. Hệ thống phân tải này có đặc điểm:
  - Các máy tính có thể được thêm vào tại một thời điểm nhất định để thỏa mãn độ tăng của tải, như vậy khả năng mở rộng của hệ thống là cao.
  - Tải có thể phân chia cho việc xử lý song song như vậy khả năng lưu thông của toàn bộ hệ thống được tăng.

### KIẾN TRÚC HỆ THỐNG PHÂN TÁN

### CÁC KIẾN TRÚC PHÂN TÁN

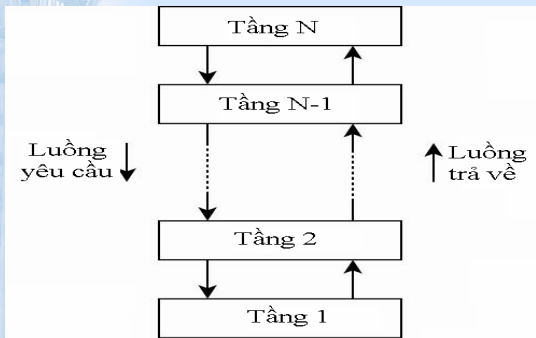
- **Kiến trúc phân tán dọc:** các công việc xử lý được thực hiện bằng cách đặt các máy tính lớn theo cấu trúc lớp. Các tiến trình xử lý được phân cho các lớp thấp hơn tương ứng với cấu trúc tổ chức và loại nhiệm vụ.
- **Kiến trúc phân tán ngang:** Nhiều máy tính được kết nối ngang hàng vào mạng để xử lý công việc, có thể thêm máy tính nhằm nâng cao độ linh hoạt và nâng cấp hệ thống. Các công việc trước kia được tập trung trên 01 máy tính thì có thể chia tính toán với các máy tính khác. Có thể sử dụng các thư viện được cung cấp từ các máy tính khác, điều này đảm bảo được sự phân tán chức năng và sử dụng chung các nguồn tài nguyên.
- **Kiến trúc phân tán kết hợp:** Kết hợp cả hai kiểu phân tán dọc và ngang.



CÁC KIỂU KIẾN TRÚC (MÔ HÌNH)  
HỆ THỐNG PHÂN TÁN

- Khái niệm: Là tổ chức logic của phần mềm hệ thống phân tán được chia thành nhiều thành phần (kiến trúc phần mềm của hệ thống phân tán)
- Phân loại:
  - Mô hình phân tầng
  - Mô hình dựa trên đối tượng
  - Mô hình dựa trên sự kiện
  - Mô hình dữ liệu tập trung

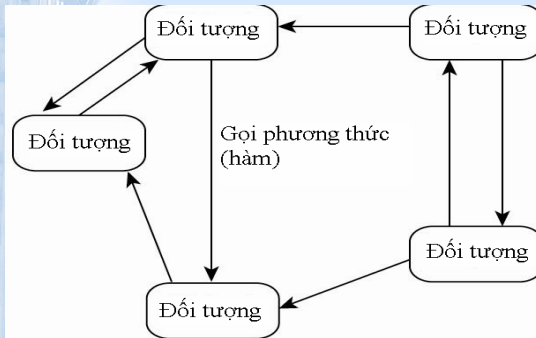
MÔ HÌNH PHÂN TẦNG



ĐẶC ĐIỂM MÔ HÌNH PHÂN TẦNG

- Các thành phần được tổ chức thành từng lớp có sự ràng buộc chặt chẽ.
- Lớp trên gọi các thành phần lớp dưới liên kế
- Bên yêu cầu: thông tin yêu cầu được lưu chuyển từ lớp trên xuống lớp dưới, kết quả trả về được chuyển từ lớp dưới lên lớp trên
- Bên thực hiện yêu cầu: thông tin yêu cầu được lưu chuyển từ lớp dưới lên lớp trên, kết quả trả về được chuyển từ lớp trên xuống lớp dưới
- Hiệu năng phụ thuộc vào số lượng tầng

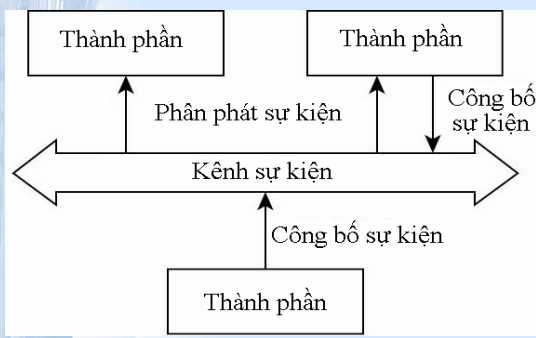
MÔ HÌNH DỰA TRÊN ĐỐI TƯỢNG



ĐẶC ĐIỂM MÔ HÌNH  
DỰA TRÊN ĐỐI TƯỢNG

- Ràng buộc lỏng hơn mô hình phân lớp
- Mỗi đối tượng được coi là một thành phần và được gọi bằng cơ chế gọi thủ tục từ xa
- Các đối tượng hoạt động tương đối độc lập, dễ dàng thay đổi và nâng cấp
- Phù hợp với kiến trúc hệ thống khách/chủ

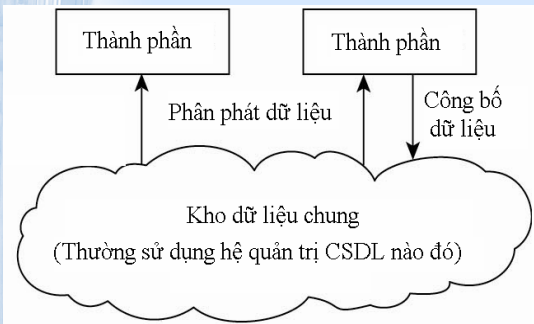
MÔ HÌNH DỰA TRÊN SỰ KIỆN



ĐẶC ĐIỂM MÔ HÌNH DỮ TRÊN SỰ KIỆN

- Về cơ bản, các tiến trình trao đổi thông tin dựa trên việc phát tán các sự kiện
- Các sự kiện thường được gắn với các luật phân phát sự kiện
- Các tiến trình phát tán sự kiện sau khi đã được phần mềm trung gian đảm bảo chỉ những tiến trình đã đăng ký mới nhận được sự kiện.
- Mức độ ràng buộc giữa các tiến trình thấp

MÔ HÌNH DỮ LIỆU TẬP TRUNG



ĐẶC ĐIỂM MÔ HÌNH DỮ LIỆU TẬP TRUNG

- Các tiến trình trao đổi thông tin với nhau qua kho dữ liệu chung ( chủ động hoặc thụ động)
- Độc lập giữa các thành phần trong hệ thống
- Tiềm lợi cho việc chia sẻ dữ liệu lớn

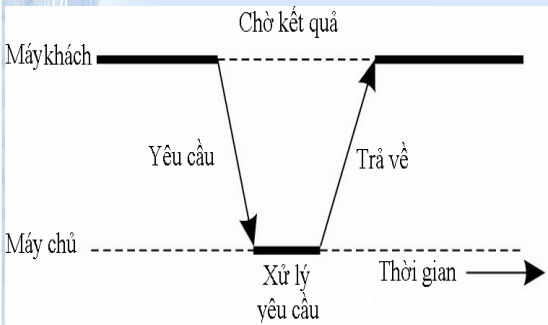
KIẾN TRÚC HỆ THỐNG

- Kiến trúc tập trung
- Kiến trúc không tập trung
- Kiến trúc lai ghép

KIẾN TRÚC TẬP TRUNG

- Máy chủ (Server): Cung cấp dịch vụ
- Khách (Client): Sử dụng dịch vụ
- Kiến trúc khách/chủ: tiến trình trên máy chủ thực hiện dịch vụ nào đó, tiến trình trên máy khách đưa ra các yêu cầu dịch vụ và chờ kết quả xử lý của tiến trình trên máy chủ
- Tương tác giữa máy khách và máy chủ có thể thực hiện trên kênh truyền không liên kết hoặc có liên kết

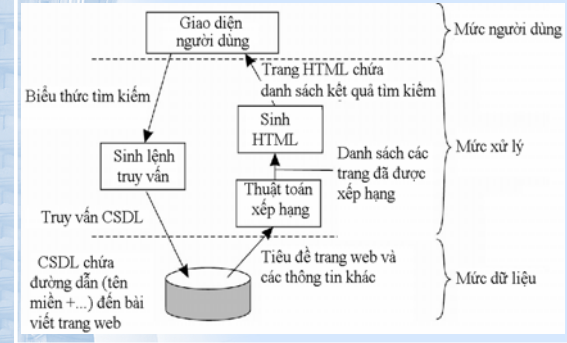
TƯƠNG TÁC GIỮA MÁY KHÁCH VÀ MÁY CHỦ TRONG KIẾN TRÚC TẬP TRUNG



PHÂN LỚP ỨNG DỤNG

- Nguyên nhân: Cần phân định rõ nhiệm vụ xử lý giữa tiến trình trên máy khách và tiến trình trên máy chủ
- Các mức xử lý:
  - Mức giao diện người dùng: Tất cả các công việc liên quan trực tiếp đến giao diện người dùng
  - Mức xử lý: Chứa các ứng dụng xử lý nghiệp vụ
  - Mức dữ liệu: Thao tác trực tiếp trên dữ liệu

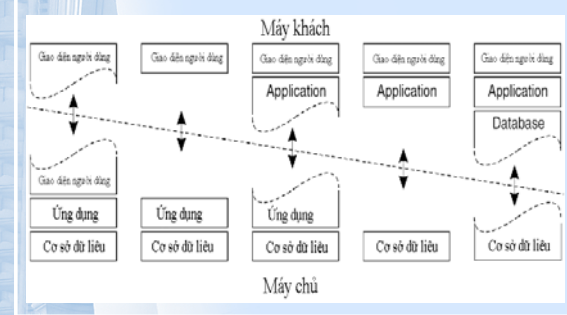
TỔ CHỨC TÌM KIẾM TRÊN INTERNET



KIẾN TRÚC NHIỀU BÊN

- Kết quả của việc phân chia ứng dụng vào các mức logic theo nguyên lý:
  - Máy trạm chỉ chứa chương trình thực hiện các chức năng thuộc mức giao diện người dùng
  - Máy chủ chứa phần còn lại: các chương trình thực hiện chức năng mức xử lý và mức dữ liệu
- Kiến trúc điển hình của kiểu phân tán đọc
- Phân chia mức độ xử lý giữa máy khách và máy chủ
- Ví dụ điển hình: Trang web

PHÂN CHIA XỬ LÝ GIỮA MÁY KHÁCH VÀ MÁY CHỦ



TƯƠNG TÁC GIỮA MÁY KHÁCH VÀ MÁY CHỦ TRONG KIẾN TRÚC NHIỀU BÊN

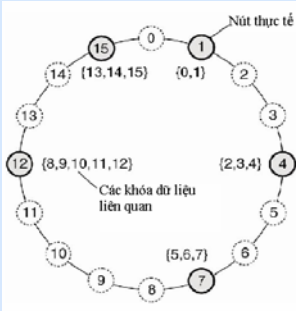


KIẾN TRÚC KHÔNG TẬP TRUNG

- Dựa trên kiểu phân tán ngang
- Tận dụng tài nguyên trong hệ thống phân tán
- Phân loại:
  - Mạng ngang hàng có cấu trúc: Sử dụng bảng băm phân tán (DHT - Distributed Hash Table)
  - Mạng ngang hàng phi cấu trúc: Sử dụng thuật toán ngẫu nhiên

KIẾN TRÚC NGANG HÀNG CÓ CẤU TRÚC SỬ DỤNG MẠCH VÒNG

- Thuật toán Chord: Tìm kiếm dịch vụ ngang hàng qui mô lớn cho các ứng dụng Internet
- Dễ mở rộng
- Phân tải xử lý

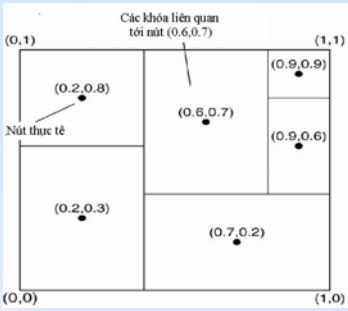


THUẬT TOÁN CHORD

- Hệ thống phân tán được tổ chức logic dưới dạng vòng N nút.
- Một nút Chord với ID là u có một con trỏ tới nút đầu tiên đứng sau nó trong không gian ID theo chiều kim đồng hồ, ký hiệu là Succ(u) và một con trỏ tới node đứng trước nó trong không gian ID, ký hiệu là Pred(u).
- Các nút tạo thành một danh sách liên kết hai chiều. Bên cạnh đó, một nút Chord lưu  $M = \log_2(N)$  con trỏ gọi là các finger.
- Tập các finger của node ID u được xác định  $F(u) = \{Succ(u + 2^{i-1})\}$ ,  $1 \leq i \leq M$ . Với cách lựa chọn finger thế này, trong mạng Chord, các nút quan sát không gian ID của vòng tròn bắt đầu từ ID của chúng. Đồng thời với cách lựa chọn finger của Chord, không gian ID sẽ được chia đôi, nửa thứ nhất cũng được chia đôi, rồi phần tư thứ nhất lại được chia đôi ... Quá trình tìm kiếm là kết quả tự nhiên theo cách chia không gian ID dựa trên việc tìm ID successor.
- Trong điều kiện thông thường, độ phức tạp của thuật toán là  $O(\log^2(N))$ .

KIẾN TRÚC NGANG HÀNG CÓ CẤU TRÚC SỬ DỤNG KHÔNG GIAN ĐA CHIỀU

Ánh xạ của các mục dữ liệu vào các nút trong thuật toán CAN.



THUẬT TOÁN CAN

- CAN sử dụng không gian đa chiều.
- Không gian khóa được chia thành các vùng nhỏ cho các nút trong mạng quản lý, mỗi nút sẽ chịu trách nhiệm toàn bộ các khóa trong vùng quản lý nắm giữ.
- Mỗi khóa là một cặp (khóa, giá trị), cặp (khóa, giá trị) này được ánh xạ vào nút tương ứng.
- Các nút sẽ duy trì thông tin định tuyến với các nút hàng xóm của nó.
- Các nút hàng xóm sẽ liên lạc với nhau để định tuyến tìm tới các nút lưu giữ khóa ở xa.

KIẾN TRÚC NGANG HÀNG PHI CẤU TRÚC

- Liên kết giữa các nút trong hệ thống được thiết lập ngẫu nhiên
- Các mục dữ liệu được đặt ngẫu nhiên tại mỗi nút.
- Tìm kiếm được thực hiện bằng cách gửi truy vấn tìm kiếm trên toàn mạng
- Một trong những mục tiêu của hệ thống ngang hàng phi cấu trúc là xây dựng mạng bao phủ tương tự đồ thị ngẫu nhiên
- Mỗi nút chứa danh sách ngẫu nhiên các nút hàng xóm
- Mỗi nút định kỳ trao đổi các mục dữ liệu với các nút hàng xóm bằng hai luồng (thread) chủ động và bị động
- Luồng chủ động khởi sướng việc trao đổi thông tin

LUỒNG (THREAD) CHỦ ĐỘNG

```
Actions by active thread (periodically repeated):
select a peer P from the current partial view;
if PUSH_MODE {
  mybuffer = [(MyAddress, 0)];
  permute partial view;
  move H oldest entries to the end;
  append first c/2 entries to mybuffer;
  send mybuffer to P;
} else {
  send trigger to P;
}
if PULL_MODE {
  receive P's buffer;
}
construct a new partial view from the current one and P's buffer;
increment the age of every entry in the new partial view;
(a)
```



LUỒNG (THREAD) BỊ ĐỘNG

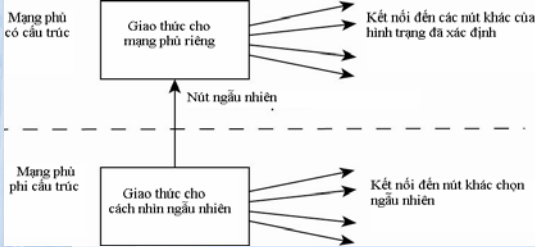
Actions by passive thread:

```
receive buffer from any process Q;  
if PULL_MODE {  
  mybuffer = [(MyAddress, 0)];  
  permute partial view;  
  move H oldest entries to the end;  
  append first c/2 entries to mybuffer;  
  send mybuffer to P;  
}  
construct a new partial view from the current one and P's buffer;  
increment the age of every entry in the new partial view;
```

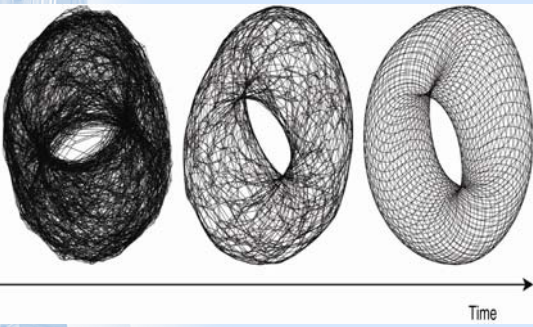
(b)

QUẢN LÝ HÌNH TRẠNG CÁC MẠNG PHỦ

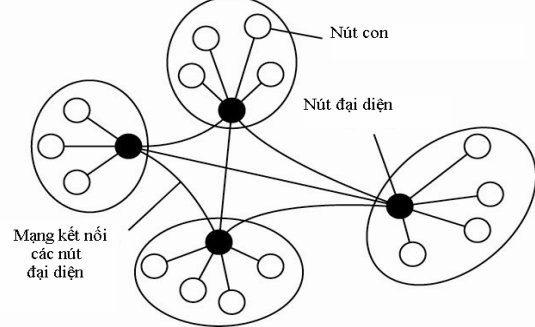
- Có thể xây dựng và duy trì mạng phủ riêng bằng cách trao đổi và lựa chọn các mục cần thận từ các nút hàng xóm
- Lớp dưới duy trì độ chính xác của đồ thị ngẫu nhiên



QUẢN LÝ HÌNH TRẠNG CÁC MẠNG PHỦ



MẠNG CỦA CÁC MẠNG NGANG HÀNG

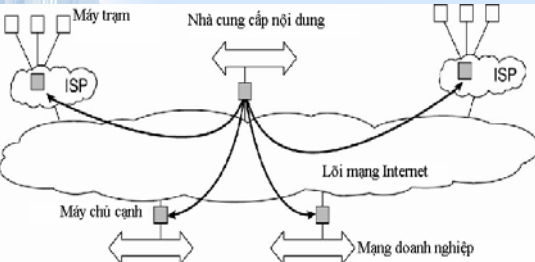


KIẾN TRÚC LAI GHÉP

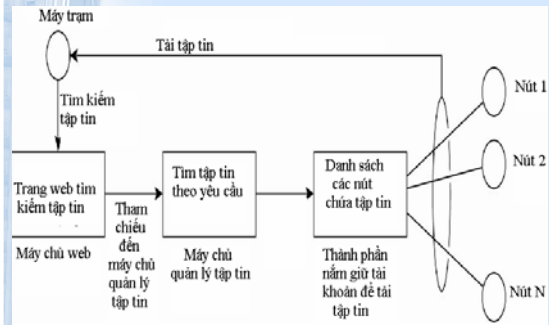
- Nhiều hệ thống phân tán sử dụng kết hợp nhiều kiến trúc khác nhau
- Mục tiêu: Khắc phục nhược điểm của mỗi loại kiến trúc

CÁC HỆ THỐNG MÁY CHỦ CẠNH (Edge)

- Tối ưu hóa nội dung và ứng dụng
- Bảo mật



CÁC HỆ THỐNG PHÂN TÁN CỘNG TÁC



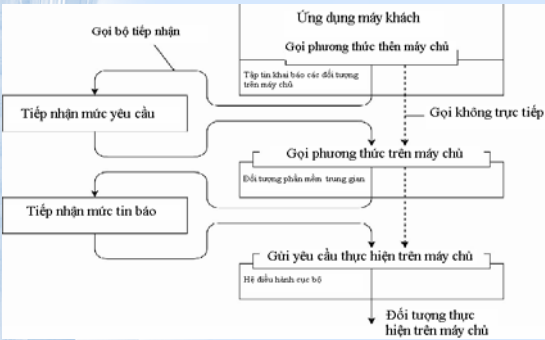
CÁC THÀNH PHẦN CỦA MẠNG PHÂN BỐ NỘI DUNG CỘNG TÁC

- Tồn tại thành phần có thể chuyển hướng yêu cầu của máy khách đến máy chủ khác
- Tồn tại thành phần phân tích mẫu truy nhập
- Tồn tại thành phần quản lý việc nhân bản các trang web

PHẦN MỀM TRUNG GIAN

- Phần mềm trung gian nằm ở đầu trong kiến trúc hệ thống phân tán?
- Hầu hết các phần mềm trung gian dựa trên mô hình đối tượng?
- Sử dụng phần mềm trung gian sẽ đơn giản hóa việc thiết kế và cài đặt ứng dụng phân tán, tuy nhiên hệ thống phần mềm sẽ phụ thuộc vào phần mềm trung gian này

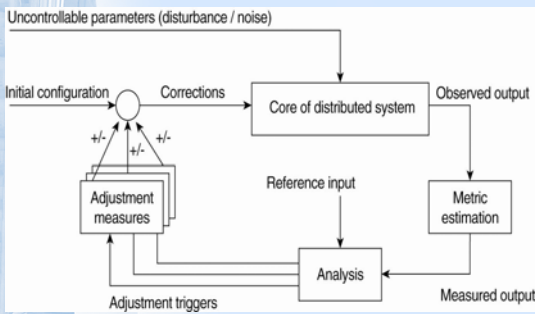
BỘ TIẾP NHẬN TRUNG GIAN



CÁCH TIẾP CẬN CHUNG ĐỐI VỚI CÁC PHẦN MỀM THÍCH NGHI

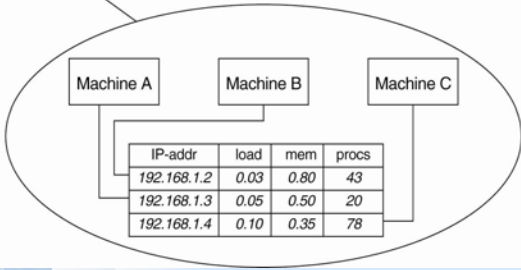
- Tách riêng các mối liên quan: Dùng phương pháp cổ điển mô đun hóa hệ thống
- Tương phản tính toán: Khả năng tự kiểm tra, trong trường hợp cần thiết tự thích nghi. Cung cấp các phương tiện cho phép thay đổi trong thời gian chạy
- Thiết kế dựa trên thành phần: Hệ thống có thể cấu hình tĩnh hay động

MÔ HÌNH ĐIỀU KHIỂN HỒI QUY

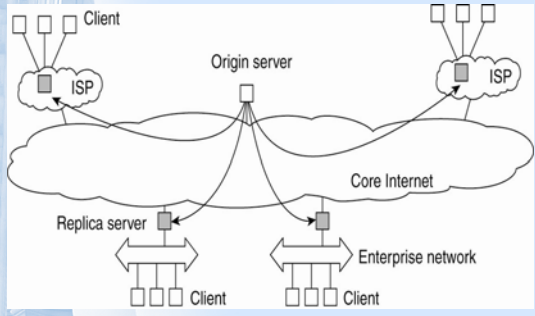


HỆ THỐNG GIÁM SÁT MÁY TÍNH

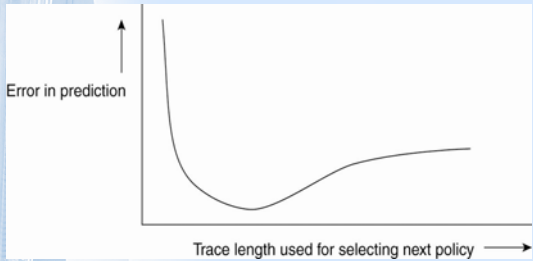
avg_load	avg_mem	avg_procs
0.06	0.55	47



CÁC CHIẾN LƯỢC NHÂN BẢN TRONG Globule



CÁC CHIẾN LƯỢC NHÂN BẢN TRONG Globule



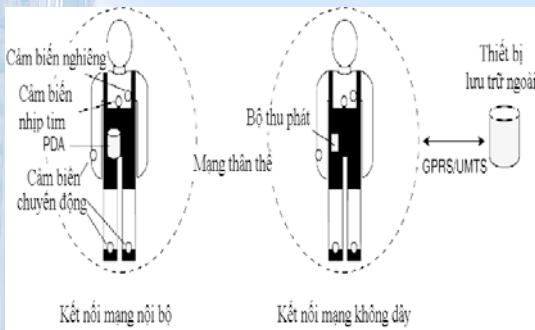
QUẢN LÝ SỬA CHỮA PHÂN HỆ TỰ ĐỘNG Jade

- Dừng việc liên kết phân hệ không lỗi vào nút bị lỗi
- Yêu cầu người quản lý khởi tạo và thêm nút mới
- Cấu hình nút mới giống như nút đã hỏng
- Thiết lập lại liên kết đã bị tạm dừng do lỗi

CÁC YÊU CẦU ĐỐI VỚI HỆ THỐNG CHĂM SÓC SỨC KHỎE

- Ở đâu và bằng cách nào lưu trữ dữ liệu đã được giám sát.
- Làm thế nào để không mất mát dữ liệu quan trọng?
- Dùng hạ tầng nào để phát sinh và phát tán cảnh báo?
- Làm thế nào để các bác sĩ đưa ra các kết luận trực tuyến?
- Hệ thống sẽ triển khai thế nào khi bùng nổ lượng người truy nhập
- Cái gì đáng phải bảo mật? Các chính sách bảo mật riêng được thực hiện như thế nào?

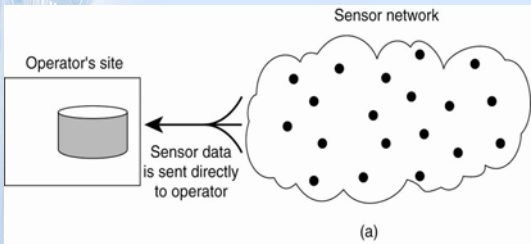
THU THẬP SỐ LIỆU CHĂM SÓC SỨC KHỎE



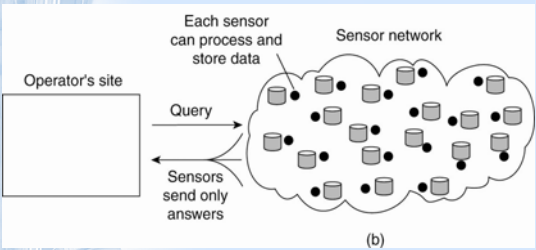
CÁC VẤN ĐỀ LIÊN QUAN ĐẾN MẠNG CẢM BIẾN

- Thiết lập biểu đồ cảm biến như thế nào cho hiệu quả
- Làm thế nào để tập hợp được các kết quả cảm biến? Có thể điều khiển được chúng không?
- Xử lý tình huống đường truyền mạng bị hỏng?

LƯU TRỮ DỮ LIỆU TẬP TRUNG

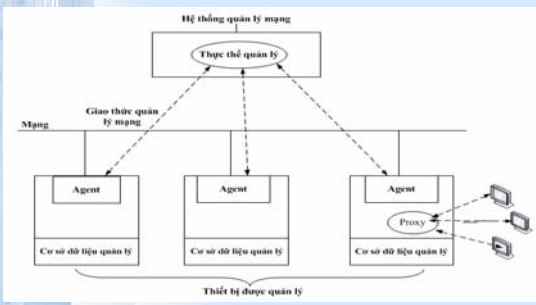


LƯU TRỮ DỮ LIỆU PHÂN TÁN



GIÁM SÁT MẠNG MÁY TÍNH

- Định dạng tên tập tin: Class\_Row\_Column\_FullName



NỘI DUNG THẢO LUẬN

- Thuật toán Chord
- Thuật toán CAN
- Thiết kế hệ thống quản lý mạng máy tính