

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

TIẾN TRÌNH TRONG CÁC HỆ THỐNG PHÂN TÁN

NỘI DUNG

- Tiến trình và luồng (Thread)
- Ảo hóa
- Máy khách
- Máy chủ
- Di trú mã

KHÁI NIỆM VỀ TIẾN TRÌNH (PROCESS)

- Để thực hiện chương trình, hệ điều hành tạo ra một số bộ xử lý ảo, mỗi bộ xử lý ảo phụ trách một chương trình khác nhau.
- Để lưu vết các bộ xử lý ảo, hệ điều hành có bảng tiến trình, mỗi mục trong bảng lưu trữ giá trị các thanh ghi trong của CPU, bản đồ bộ nhớ, các tập tin đang mở, thông tin tài khoản, quyền...
- Tiến trình thường được định nghĩa là chương trình đang chạy trên một bộ xử lý ảo của hệ điều hành
- Hệ điều hành phải đảm bảo sự độc lập của các tiến trình: đảm bảo tính trong suốt trong việc giải quyết vấn đề tương tranh giữa các tiến trình.

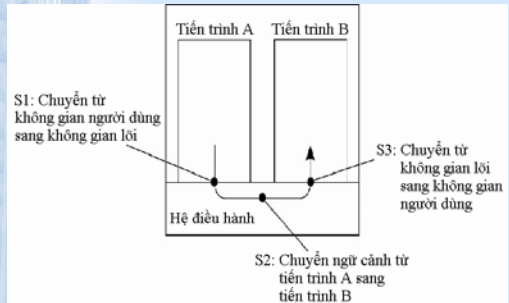
LỜI GỌI HỆ THỐNG (SYSTEM CALL)

- Khái niệm: là tập lệnh mở rộng độ hệ điều hành cung cấp xác định giao diện giữa hệ điều hành và các chương trình người sử dụng.
- Phân loại:
 - Phong tỏa (Blocking System call) là lời gọi hệ thống mà sau khi được gọi bởi tiến trình người sử dụng thì tiến trình này bị dừng lại cho đến khi thực hiện xong lời gọi hệ thống.
 - Không phong tỏa (Non – Blocking System call): sau khi gọi, điều khiển được trả lại cho tiến trình gọi và tiến trình này tiếp tục thực hiện song song với lời gọi hệ thống.

CÁC VẤN ĐỀ NÀY SINH KHI SỬ DỤNG TIẾN TRÌNH

- Thực hiện tính trong suốt tương tranh, hệ điều hành đã phải trả giá khá cao:
 - Tạo không gian địa chỉ hoàn toàn độc lập: Xóa bộ nhớ, sao chép dữ liệu mới của tiến trình vào bộ nhớ đó
 - Chuyển CPU giữa hai tiến trình đòi hỏi chi phí cao: Bên cạnh việc ghi nhớ ngữ cảnh CPU (giá trị các thanh ghi, biên đếm chương trình, con trỏ ngăn xếp...) hệ điều hành còn phải thay đổi các thanh ghi của đơn vị bộ nhớ (MMU) và bộ nhớ đệm trong (TBL)
 - Khi chạy nhiều tiến trình, có thể xảy ra quá trình swap: chuyển dữ liệu giữa ổ đĩa và bộ nhớ (RAM)
- Việc chia nhỏ tiến trình bởi hệ điều hành là chưa đủ, cần thiết phải chia nhỏ tiến trình hơn nữa để dễ dàng xây dựng phần mềm hơn và đồng thời đạt được hiệu năng cao hơn

CHUYỂN NGỮ CẢNH TRONG TRUYỀN THÔNG LIÊN TIẾN TRÌNH



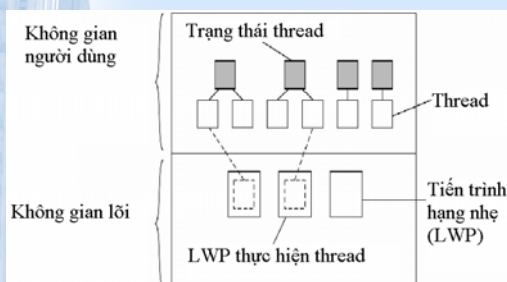
KHÁI NIỆM LUỒNG (THREAD)

- Giống như tiến trình, một luồng thực hiện đoạn mã của mình độc lập với các luồng khác.
- Khác với tiến trình, một luồng không có đạt được mức trong suốt cao, nếu điều đó làm suy giảm hiệu năng.
- Nói chung, mỗi luồng chỉ duy trì thông tin tối thiểu để cho phép các luồng chia sẻ CPU.
- Ngữ cảnh của luồng thường chỉ bao gồm ngữ cảnh của CPU, các thông tin khác bị bỏ qua. Lập trình viên phải tự xây dựng cơ chế bảo vệ dữ liệu cho mỗi luồng.

CÀI ĐẶT LUỒNG

- Luồng thường được cài đặt dưới dạng gói luồng (thread package), các gói đó chứa các thao tác tạo/hủy luồng và thao tác trên các biến đồng bộ như mutex hoặc biến điều kiện
- Hai cách tiếp cận cơ bản:
 - Xây dựng thư viện luồng chạy hoàn toàn trong chế độ người dùng. Chi phí thấp trong việc tạo/hủy luồng, việc chuyển ngữ cảnh luồng chỉ cần thực hiện vài chỉ thị nhưng khi thực hiện lời gọi phong tỏa hệ thống sẽ phong tỏa ngay tiến trình chứa luồng
 - Phân lõi (kernel) biết và lập lịch cho các luồng, chi phí cao khi chuyển ngữ cảnh luồng
- Lai ghép mức lõi và mức người dùng (LWP-tiến trình hạng nhẹ): Là tiến trình hạng nặng (mức lõi) cung cấp gói luồng cho mức người dùng

KẾT HỢP TIẾN TRÌNH NHẸ VÀ CÁC LUỒNG MỨC NGƯỜI DÙNG



LUỒNG TRONG CÁC HỆ PHÂN TÁN

- Luồng cung cấp các phương tiện thuận lợi cho phép phong tỏa lời gọi hệ thống mà không phong tỏa tiến trình của nó
- Dựa trên đặc điểm này, có thể sử dụng luồng để duy trì đồng thời nhiều kênh truyền thông logic.
- Trong hệ thống phân tán cần phải xây dựng đa luồng cả ở hai phía khách và chủ

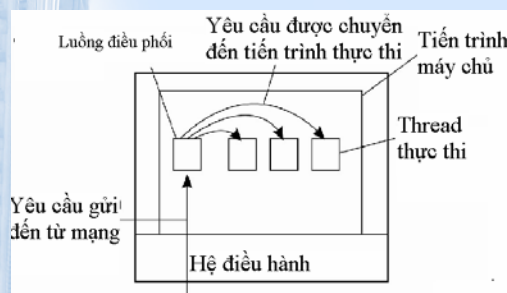
ĐA LUỒNG TRONG MÁY KHÁCH

- Để đạt được độ trong suốt cao, các hệ thống phân tán thường che giấu quá trình truyền thông điệp
- Độ trễ toàn phần trên mạng có thể lên tới hàng trăm ms
- Để giải quyết vấn đề này, cần thiết phải xây dựng hệ thống xử lý song song, điều này đòi hỏi phải thiết kế đa luồng trên cả máy khách lẫn máy chủ

ĐA LUỒNG TRÊN MÁY CHỦ

- Tiến trình trên máy chủ có thể được xây dựng theo kiểu đơn luồng hoặc đa luồng.
- Đơn luồng máy chủ hoạt động theo nguyên tắc khi có yêu cầu gửi đến, máy chủ sẽ tiếp nhận và xử lý yêu cầu, trong thời gian xử lý sẽ không tiếp nhận yêu cầu khác.
- Đa luồng máy chủ hoạt động bằng cách, một luồng điều phối sẽ tiếp nhận và phân phát yêu cầu cho các luồng thực thi.
- Đa luồng trên máy chủ không chỉ đơn giản hóa việc viết mã chương trình mà còn giúp cho khả năng xử lý song song trên các máy chủ

MÔ HÌNH ĐIỀU PHỐI/THỰC THI TRÊN MÁY CHỦ



BẢ MÔ HÌNH XÂY DỰNG MÁY CHỦ

Mô hình	Đặc tính
Đa luồng	Song song, phong tỏa gọi hệ thống
Tiến trình đơn luồng	Không song song, phong tỏa lời gọi hệ thống
Máy trạng thái hữu hạn	Song song, không phong tỏa lời gọi hệ thống

VAI TRÒ ẢO HÓA TRONG HỆ PHÂN TÁN

Chương trình	Chương trình
Giao diện A	Giao diện A
Phản ứng/Hệ thống phần mềm A	Cài đặt bất chức A trên B
	Giao diện B
	Phản ứng/Hệ thống phần mềm B
Nguyên lý tổ chức giao diện, chương trình và hệ thống	Ảo hóa nguyên lý tổ chức giao diện, chương trình và hệ thống

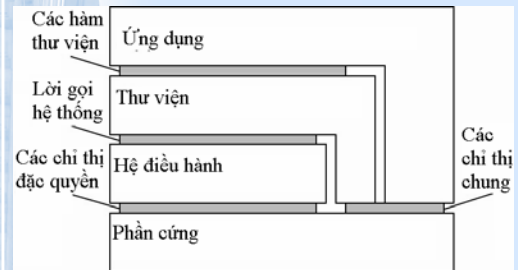
MÁY ẢO

- Tiến trình và luồng là những kỹ thuật dùng để thực hiện đồng thời nhiều thao tác
- Trong các hệ thống chỉ có 01 bộ vi xử lý, việc thực hiện đồng thời nhiều thao tác chỉ là ảo giác, các tiến trình/luồng chia sẻ 01 tài nguyên phần cứng đó.
- Để luân chuyển nhanh chóng giữa các tiến trình/luồng, người ta đã tạo ra một ảo giác xử lý song song.
- Việc tách biệt các tiến trình để bảo đảm tại một thời điểm chỉ có duy nhất một tiến trình được sử dụng tài nguyên phần cứng gọi là ảo hóa.
- Bản chất của ảo hóa là bất chức giao diện giữa ứng dụng và tài nguyên phần cứng.

CÁC MỨC GIAO DIỆN LẬP TRÌNH

- Giao diện giữa phần cứng và phần mềm gồm các chỉ thị máy có thể gọi được bằng bất kỳ chương trình nào
- Giao diện giữa phần cứng và phần mềm gồm các chỉ thị máy chỉ có thể gọi được bằng các chương trình đặc quyền (ví dụ hệ điều hành)
- Giao diện gồm các lời gọi hệ thống do hệ điều hành cung cấp
- Giao diện gồm các lời gọi thư viện (thường hình thành giao diện lập trình API)

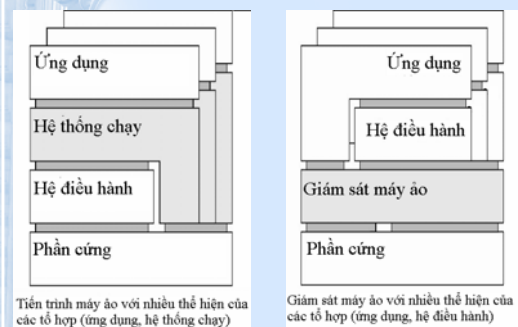
CÁC MỨC GIAO DIỆN LẬP TRÌNH



CÁC PHƯƠNG PHÁP ẢO HÓA

- Tiến trình máy ảo (Process Virtual Machine): cung cấp cho ứng dụng tập các chỉ thị đã được trừu tượng hóa. Về cơ bản, việc ảo hóa chỉ được thực hiện cho 01 tiến trình
- Giám sát máy ảo (VMM- Virtual Machine Monitor): Tạo một lớp bao bọc toàn bộ phần cứng, cho phép nhiều ứng dụng có thể đồng thời chạy trên lớp này.
- VMM ngày càng trở nên quan trọng trong các hệ thống phân tán, nó không những đảm bảo tính bảo mật mà còn cung cấp tính năng khả chuyển cho hệ thống

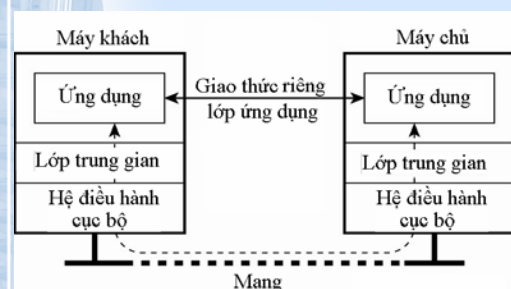
ẢO HÓA MỨC TIỀN TRÌNH VÀ MỨC MÁY



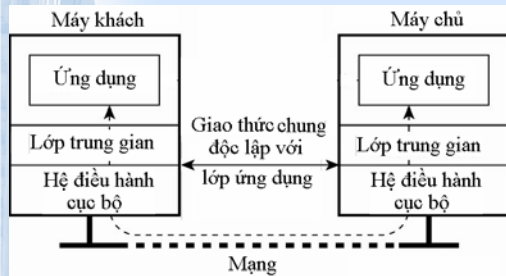
MÁY KHÁCH

- Nhiệm vụ chính của máy khách là cung cấp các phương tiện để người sử dụng truy nhập vào máy chủ
- Hai phương pháp thường sử dụng:
 - Xây dựng giao thức tương ứng với từng dịch vụ
 - Xây dựng lớp trung gian chung cho toàn bộ các dịch vụ

ỨNG DỤNG MẠNG VỚI GIAO THỨC RIÊNG

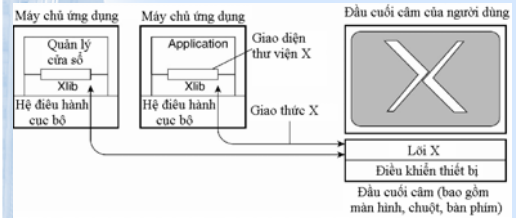


ỨNG DỤNG MẠNG VỚI GIAO THỨC CHUNG



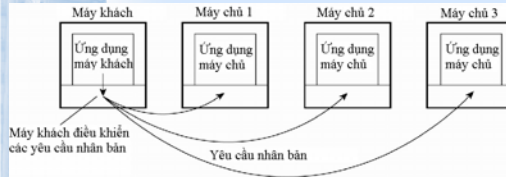
TỔ CHỨC CỦA HỆ THỐNG Xwindow

- Đầu cuối cảm là hệ thống điều khiển ảnh bitmap
- Hạt nhân là lõi X, chứa toàn bộ các trình điều khiển thiết bị



PHẦN MỀM PHÍA MÁY KHÁCH ĐỐI VỚI TRONG SUỐT PHẦN TẢN

- Phần mềm phía máy khách không chỉ giao tiếp với người dùng, trong nhiều trường hợp nó có chức năng xử lý dữ liệu
- Dùng Client Stub để đảm bảo tính trong suốt truy nhập

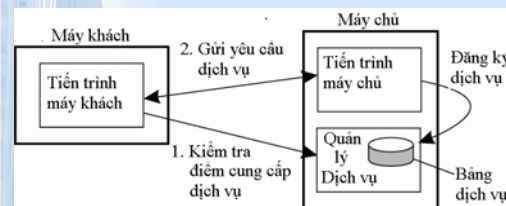


MÁY CHỦ

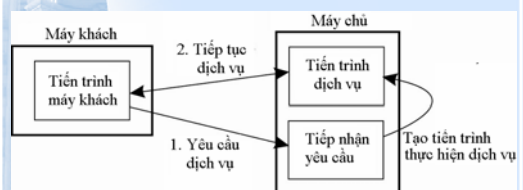
- Máy chủ là tiến trình thực dịch vụ riêng thay cho một nhóm máy trạm
- Nguyên lý chung của máy chủ: Máy chủ chờ yêu cầu từ máy trạm gửi đến, xử lý xong yêu cầu đó lại chờ nhận yêu cầu tiếp theo
- Có thể tổ chức máy chủ theo hai cách:
 - Máy chủ tương tác: Tiếp nhận yêu cầu của máy khách, xử lý và trả về kết quả (nếu yêu cầu) cho máy khách
 - Máy chủ tương tranh: Tiếp nhận yêu cầu của máy khách, chuyển tiếp cho một tiến trình/luồng xử lý và tiếp tục tiếp nhận các yêu cầu khác

MÁY KHÁCH KẾT NỐI QUA CÔNG DỊCH VỤ

- Máy khách truy nhập dịch vụ trên máy chủ qua điểm truy nhập (thường gọi là cổng hay điểm cuối) hoặc sử dụng máy chủ khác để định hướng lại yêu cầu dịch vụ



MÁY KHÁCH KẾT NỐI GIÁN TIẾP QUA THÀNH PHẦN TIẾP NHẬN



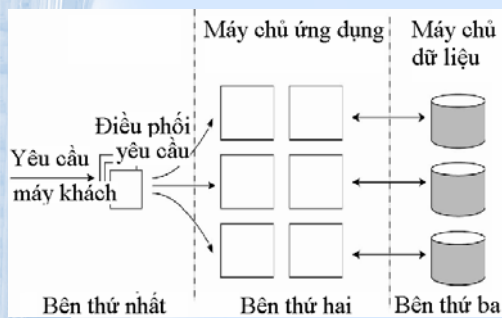
CÁC VẤN ĐỀ KHÁC

- Xử lý khi máy khách hủy bỏ yêu cầu:
 - Hủy bỏ kết nối
 - Cơ chế báo hiệu ngoài băng
- Máy chủ có thể thuộc loại có trạng thái hoặc không trạng thái (lưu hoặc không lưu trạng thái của máy khách)

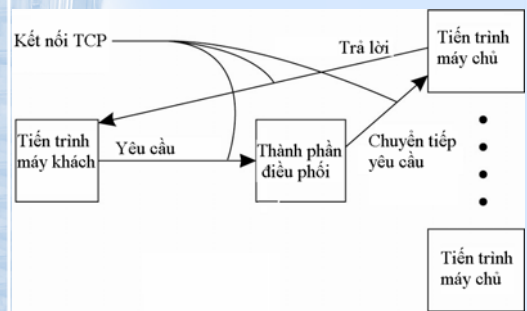
CỤM MÁY CHỦ

- Cụm máy chủ bao gồm các máy tính được kết nối với nhau qua mạng (thường là mạng nội bộ để đảm bảo băng thông cao và độ trễ nhỏ)
- Trên mỗi máy tính có thể chạy nhiều tiến trình máy chủ.
- Thường các máy chủ được tổ chức logic dưới dạng nhiều bên (Multitier)

CỤM MÁY CHỦ



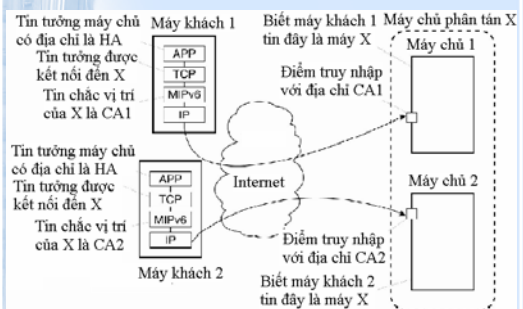
NGUYÊN LÝ CHUYỂN YÊU CẦU XỬ LÝ



MÁY CHỦ PHÂN TÁN

- Cụm máy chủ nói chung được cấu hình tương đối tĩnh.
- Thường có một máy riêng biệt để theo dõi các máy chủ khác, tiếp nhận yêu cầu và chuyển đến máy chủ tương ứng. Điều này dẫn đến hệ thống chỉ có một điểm truy nhập duy nhất.
- Để khắc phục nhược điểm truy nhập duy nhất tại một điểm, có thể sử dụng dịch vụ DNS (tra về tập địa chỉ), máy khách có thể thử kết nối từng địa chỉ cho đến khi thành công, điều đó vẫn chưa giải quyết triệt để vấn đề truy nhập qua một điểm.
- Mong muốn có sự ổn định kết nối giữa máy khách và máy chủ là điều tất yếu, nhưng bên cạnh đó nhu cầu linh hoạt của hệ thống cũng quan trọng không kém. Điều này dẫn tới ý tưởng thiết kế hệ thống có điểm truy nhập thay đổi nhưng đối với thế giới bên ngoài vẫn chỉ có một điểm truy nhập.

TỐI ƯU HÓA ĐỊNH TUYẾN TRONG HỆ THỐNG MÁY CHỦ PHÂN TÁN

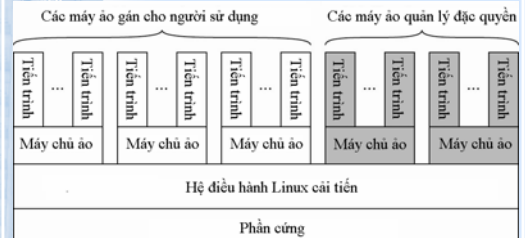


QUẢN LÝ CỤM MÁY CHỦ

- Đối với bên ngoài, cụm máy chủ chỉ là một máy tính duy nhất
- Giải pháp chung cho việc quản lý cụm máy chủ là mở rộng các chức năng quản lý một máy chủ cho cụm máy chủ
- Với mức kinh điển, người quản trị truy nhập vào từng máy bằng các công cụ truy nhập từ xa và thực hiện các thao tác quản trị thông thường
- Với mức cao hơn, người quản trị sử dụng phần mềm quản trị mạng máy chủ

HỆ THỐNG PLANETLAB

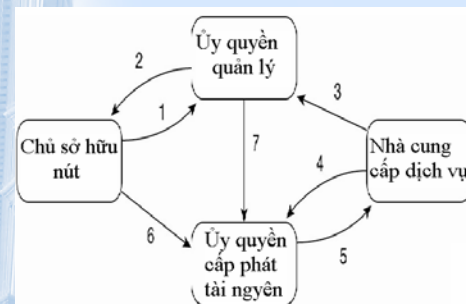
PlanetLab là hệ thống phục vụ nghiên cứu các mạng và các hệ thống phân tán, trụ sở đặt tại Đại học Princeton, California.



CÁC VẤN ĐỀ QUẢN LÝ PLANETLAB

- Các nút thuộc nhiều đơn vị khác nhau
 - Phải cho phép người của mỗi đơn vị chạy các ứng dụng trên nút của mình
 - Hạn chế sử dụng tài nguyên thích hợp
- Các công cụ giám sát có sẵn đảm đương tổ hợp riêng phần cứng và phần mềm đáp ứng yêu cầu sử dụng trong mỗi đơn vị
- Các chương trình của đơn vị khác nhau chạy trên cùng một nút không ảnh hưởng lẫn nhau

CÁC QUAN HỆ QUẢN LÝ GIỮA CÁC THỰC THỂ KHÁC NHAU CỦA PlanetLab



QUAN HỆ GIỮA CÁC THỰC THỂ PlanetLab

- Chủ sở hữu giao máy chủ cho người quản trị, có giới hạn sử dụng thích hợp
- Người quản trị cung cấp phần mềm cần thiết để thêm nút vào hệ thống PlanetLab.
- Nhà cung cấp dịch vụ tự đăng ký với người quản trị để được phép truy nhập vào máy chủ.
- Người quản trị tài nguyên (slice) cần xác thực người cung cấp dịch vụ
- Chủ sở hữu cung cấp dịch vụ tạo quyền sử dụng tài nguyên cho người quản trị tài nguyên.
- Người quản trị giao cho người quản trị tài nguyên tạo tài nguyên cho người sử dụng

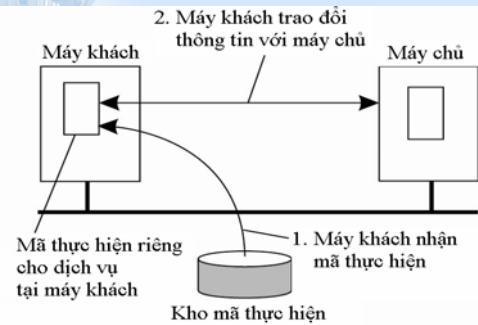
DI TRÚ MÃ

- Truyền thông trong hệ thống phân tán cho đến nay đang giới hạn ở việc truyền dữ liệu.
- Trong một số tình huống, việc truyền chương trình (thậm chí đang chạy) sẽ đơn giản hóa việc thiết kế và đồng thời tăng hiệu năng và độ linh hoạt của hệ thống.
- Thông thường, di trú mã trong hệ thống phân tán được thực hiện dưới dạng di trú tiến trình, toàn bộ tiến trình được chuyển sang máy khác. Việc di chuyển tiến trình đang chạy là công việc khá phức tạp.
- Tư tưởng cơ bản của việc di trú mã dựa trên hiệu năng tổng thể, chuyển tiến trình từ máy tính đang có tải cao sang máy tính đang chịu tải thấp hơn.

DI TRÚ MÃ

- Tải của hệ thống dựa trên hàng đợi CPU, bộ nhớ RAM...
- Thuật toán phân bổ tài xem xét tới việc định vị và phân bổ lại các nhiệm vụ nhưng phải xem xét tới việc sử dụng tài nguyên của mỗi máy tính (số lượng bộ xử lý, RAM...).
- Trong các hệ thống phân tán, việc tối ưu hóa khả năng tính toán ít được coi trọng hơn việc giảm thiểu truyền thông.
- Việc di trú mã có thể thực hiện bằng giải pháp chia sẻ xử lý giữa máy khách và máy chủ, máy khách có thể áp dụng các giải pháp xử lý song song. Điều này còn góp phần nâng cao tính linh hoạt của hệ thống phân tán.

NGUYÊN LÝ CẤU HÌNH ĐỘNG CHO MÁY TRẠM



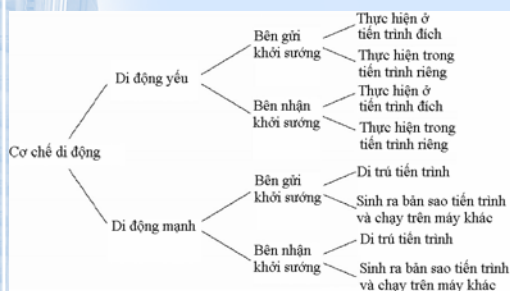
CÁC THÀNH PHẦN CỦA TIẾN TRÌNH

- Phần mã** (Code Segment): chứa tập các lệnh của tiến trình đang thực hiện.
- Phần tài nguyên** (Resource Segment): chứa các tham chiếu đến tất cả các tài nguyên bên ngoài mà tiến trình đang cần.
- Phần thực thi** (Execution segment): chứa các trạng thái thực thi hiện hành của tiến trình.

CÁC GIẢI PHÁP DI TRÚ MÃ

- Di động yếu:** chỉ truyền phần mã và một số dữ liệu khởi động của tiến trình. Chương trình được truyền đi luôn được bắt đầu từ trạng thái khởi động, chỉ yêu cầu máy đích có thể thực thi yêu cầu đó.
- Di động mạnh:** truyền cả phần mã và phần thực thi. Một tiến trình đang chạy có thể được dừng lại rồi chuyển đến một máy khác và tiếp tục thực hiện tiếp tiến trình đó, giải pháp này khó thực thi hơn.
- Di trú do bên gửi khởi sướng:** Di trú được khởi động từ máy mà phần mã của tiến trình được lưu trữ hoặc đang thực hiện trên máy gửi. Di trú này hoàn thành khi tài xong chương trình.
- Di trú do bên nhận khởi sướng:** Di trú mã ban đầu của máy gửi. Thực hiện đơn giản hơn so với di trú được khởi sướng từ bên gửi.

CÁC PHƯƠNG ÁN DI TRÚ MÃ



DI TRÚ VÀ TÀI NGUYÊN CỤC BỘ

- Khác với di trú phần mã và phần thực thi, việc di trú phần tài nguyên đòi hỏi những thay đổi nhất định.
- Phân biệt ba loại những tiến trình với tài nguyên
 - Mạnh: Tiến trình tham chiếu đến tài nguyên bằng định danh (ví dụ URL)
 - Yếu: Những bằng giá trị, tiến trình không bị ảnh hưởng nếu được cung cấp nguồn tài nguyên khác cùng giá trị (ví dụ thư viện lập trình)
 - Yếu nhất: Tiến trình chỉ cần kiểu tài nguyên (ví dụ thiết bị ngoại vi...)
- Phân biệt ba loại những tài nguyên với máy tính
 - Tài nguyên không dính kèm: Dễ dàng di chuyển giữa các máy (ví dụ các tập tin)
 - Tài nguyên gắn kèm: Khó di chuyển giữa các máy (ví dụ CSDL cục bộ, trang web...)
 - Tài nguyên cố định: Những tài nguyên gắn chặt với từng máy (ví dụ thiết bị, công...)

DI TRÚ VÀ TÀI NGUYÊN CỤC BỘ

GR: Thiết lập tham chiếu trên toàn bộ hệ thống

MV: Di chuyển tài nguyên

CP: Sao chép giá trị của tài nguyên

RB: Nhúng lại tiến trình đến tài nguyên cục bộ có sẵn

	Tài nguyên -Máy tính	Không gắn kết	Gắn kết	Cố định
Tiến trình -Tài nguyên				
Mạnh		MV hoặc GR	GR hoặc MV	GR
Yếu		CP hoặc MV,GR	GR hoặc CP	GR
Rất yếu		RB hoặc MV,CP	RB hoặc GR, CP	RB hoặc GR

DI TRÚ TRONG CÁC HỆ THỐNG KHÔNG ĐỒNG NHẤT

- Đẩy các trang nhớ sang máy mới và gửi lại các trang đã thay đổi gần nhất trong quá trình di trú.
- Dùng máy ảo hiện hành, di trú bộ nhớ và khởi tạo máy ảo mới
- Cho phép máy ảo mới kéo các trang cần thiết (cho phép các tiến trình bắt đầu trên máy ảo mới ngay lập tức sao chép các trang bộ nhớ theo nhu cầu)