

CHƯƠNG 1: TỔNG QUAN VỀ HỆ ĐIỀU HÀNH.....	4
1.1 Khái niệm hệ điều hành.....	4
1.2 Lịch sử phát triển của hệ điều hành.....	5
1.3. Phân loại hệ điều hành	7
1.3.1 Hệ điều hành xử lý theo lô đơn giản	7
1.3.2 Hệ điều hành xử lý theo lô đa chương	7
1.3.3 Hệ điều hành chia xẻ thời gian	8
1.3.4 Hệ điều hành đa vi xử lý.	8
1.3.5 Hệ điều hành mạng	9
1.3.6 Hệ điều hành xử lý thời gian thực	9
1.4 Các thành phần của hệ điều hành.....	9
1.5 Các cấu trúc của hệ thống	14
1.6 Các tính chất cơ bản của hệ điều hành	19
1.7 Nguyên lý xây dựng chương trình hệ điều hành.....	20
1.8 Các hình thái giao tiếp.....	22
CHƯƠNG 2 QUẢN LÝ TIẾN TRÌNH.....	24
2.1 Tổng quan về tiến trình	24
2.1.1 Tiến trình (Process) và mô hình đa tiến trình (Multiprocess).....	24
2.1.2 Tiểu trình (Thread) và mô hình đa tiểu trình (Multithread).....	25
2.1.3 Phân loại tiến trình	26
2.1.4. Các trạng thái của tiến trình	27
2.1.5. Cấu trúc dữ liệu của khối quản lý tiến trình.....	29
2.1.6. Các thao tác điều khiển tiến trình.....	30
2.1.7 Cấp phát tài nguyên cho tiến trình	32
2.2. Điều phối tiến trình	34
2.2.1. Mục tiêu điều phối	34
2.2.2 Điều phối độc quyền và điều phối không độc quyền (preemptive/nopreemptive)	34
2.2.3. Các danh sách sử dụng trong quá trình điều phối.	36
2.2.4. Các chiến lược điều phối.....	37
2.3. Thông tin liên lạc giữa các tiến trình	43
2.3.1. Nhu cầu liên lạc giữa các tiến trình	43
2.3.2. Các Cơ Chế Thông Tin Liên lạc	44
2.4 Đồng bộ hoá tiến trình.....	50
2.4.1 Nhu cầu đồng bộ hóa (synchronisation)	50
2.4.2. Bài toán đồng bộ hoá	51
2.4.3 Các giải pháp đồng bộ hoá	53
2.5. Tắc nghẽn (Deadlock)	66
2.5.1. Định nghĩa:	66
2.5.2. Điều kiện xuất hiện tắc nghẽn.....	67
2.5.3. Các phương pháp xử lý tắc nghẽn.....	68

2.5.4 Ngăn chặn tắc nghẽn	69
2.5.5. Tránh tắc nghẽn.....	70
2.5.6. Hiệu chỉnh tắc nghẽn.....	75
CHƯƠNG 3 :QUẢN LÝ BỘ NHỚ CHÍNH.....	77
3.1 Tổ chức vùng nhớ.....	77
3.2 Mục tiêu của việc quản lý vùng nhớ	77
3.3 Không gian địa chỉ và không gian vật lý.....	78
3.4. Cấp phát liên tục.....	78
3.4.1 Hệ đơn chương	79
3.4.2 Hệ thống đa chương với phân vùng cố định	79
3.4.3 Hệ thống đa chương với phân vùng động.....	81
3.5. Cấp phát không liên tục.....	84
3.5.1 Kỹ thuật phân trang (Paging)	84
3.5.2. Phân đoạn (Segmentation)	92
3.5.3. Phân đoạn kết hợp phân trang (Paged segmentation)	96
3.6 Kỹ thuật bộ nhớ ảo (Virtual Memory)	97
3.6.1. Bộ nhớ ảo	97
3.6.2. Cài đặt bộ nhớ ảo	99
3.6.3.Các thuật toán thay thế trang.....	101
Chương 4 QUẢN LÝ VÙNG NHỚ PHỤ.....	106
4.1 Một số khái niệm dùng quản lý đĩa.....	106
4.2 Hệ thống bảng FAT.....	110
4.2.1 Quản lý file trên đĩa của MS_DOS	110
4.2.2 Hệ thống NTFS (New Technology File System).....	115
4.3 Các thông số và thuật toán truy nhập đĩa	116
4.3.1Các thông số	116
4.3.2 Các thuật toán đọc đĩa	117
Chương 5 QUẢN LÝ VÀO RA	121
5.1 Khái niệm về hệ thống quản lý vào/ ra	121
5.2 Phần cứng vào/ra.....	121
5.2.1 Các thiết bị vào/ra	121
5.2.2 Tổ chức của chức năng I/O	122
5.2.3 Bộ điều khiển thiết bị	123
5.2.4 Truy nhập bộ nhớ trực tiếp DMA (Direct Memory Access)	125
5.3 Phần mềm vào/ra.....	125
5.3.1 Kiểm soát ngắt.....	125
5.3.2 Điều khiển thiết bị (device drivers).....	126
5.3.3 Phần mềm nhập/xuất độc lập thiết bị	127
5.3.4 Phần mềm vào/ra phạm vi người sử dụng	128
Chương 6: HỆ THỐNG QUẢN LÝ FILE.....	129
6.1 File và các thuộc tính của file	129

6.2 Thư mục: khái niệm, hệ thống thư mục, tổ chức bên trong	131
6.3 Các phương pháp lưu giữ file.....	133
6.4 Hệ thống quản lý tập tin (File management system).	136
6.5 Các thao tác file.....	137
6.6 Tổ chức file, truy nhập file.....	138
6.7 Độ an toàn của hệ thống file	139

TaiLieu.vn

CHƯƠNG 1: TỔNG QUAN VỀ HỆ ĐIỀU HÀNH

1.1 Khái niệm hệ điều hành

Phần mềm máy tính có thể chia thành nhiều loại: chương trình hệ thống quản lý sự hoạt động của chính máy tính. Chương trình ứng dụng, giải quyết các vấn đề liên quan đến việc sử dụng và khai thác máy tính của người sử dụng. Hệ điều hành thuộc nhóm các chương trình hệ thống.

Hệ điều hành là một hệ thống các chương trình hoạt động giữa người sử dụng (user) và phần cứng của máy tính. Mục tiêu của hệ điều hành là cung cấp một môi trường thuận lợi để người sử dụng có thể thi hành các chương trình. Hệ điều hành thực hiện quản lý các tài nguyên máy tính. Hệ điều hành làm cho máy tính dễ sử dụng hơn, thuận lợi hơn và hiệu quả hơn.

Để đạt được mục tiêu trên hệ điều hành phải thực hiện 2 chức năng chính sau đây:

- Giả lập một máy tính mở rộng

Hệ điều hành che đậy các chi tiết phần cứng của máy tính bởi một máy tính mở rộng, máy tính mở rộng này có đầy đủ các chức năng của máy tính thực nhưng đơn giản và dễ sử dụng hơn. Theo đó khi cần tác động vào máy tính thực người sử dụng chỉ cần tác động vào máy tính mở rộng, mọi sự chuyển đổi thông tin điều khiển từ máy tính mở rộng sang máy tính thực hoặc ngược lại đều do hệ điều hành thực hiện.

Mục đích của chức năng này là giúp người sử dụng khai thác các chức năng của phần cứng máy tính dễ dàng và hiệu quả hơn.

- Quản lý tài nguyên của hệ thống:

+ Tài nguyên phần cứng: CPU, RAM, I/O device...

Nhu cầu tài nguyên nhiều, do vậy cần quản lý, điều phối tài nguyên một cách có hiệu quả. Hệ điều hành còn phải tổ chức bảo vệ các không gian nhớ đã cấp cho các chương trình, tiến trình để tránh sự truy cập bất hợp lệ và sự tranh chấp bộ nhớ giữa các chương trình, tiến trình, đặc biệt là các tiến trình đồng thời hoạt động trên hệ thống.

+ Tài nguyên phần mềm (data)

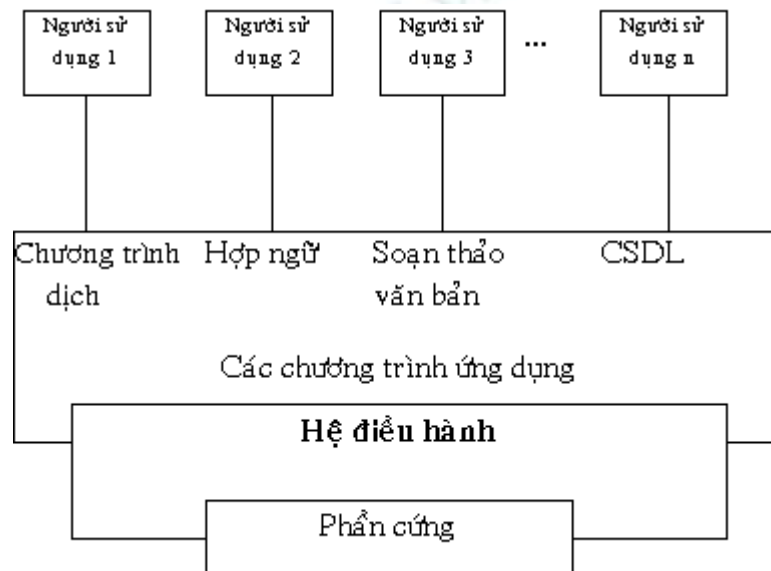
Trong trường hợp nhiều tiến trình đồng thời sử dụng một không gian nhớ hay một tập tin nào đó. Trong trường hợp này hệ điều hành phải tổ chức việc chia sẻ và giám sát việc truy xuất đồng thời trên các tài nguyên nói trên sao cho việc sử dụng tài nguyên có hiệu quả nhưng tránh được sự mất mát dữ liệu và làm hỏng các tập tin.

Hệ điều hành là một phần quan trọng của hầu hết các hệ thống máy tính. Một hệ thống máy tính thường được chia làm bốn phần chính : phần cứng, hệ điều hành, các chương trình ứng dụng và người sử dụng.

Phần cứng bao gồm CPU, bộ nhớ, các thiết bị nhập xuất, đây là những tài nguyên của máy tính.

Chương trình ứng dụng như các chương trình dịch, hệ thống cơ sở dữ liệu, các trò chơi, và các chương trình thương mại. Các chương trình này sử dụng tài nguyên của máy tính để giải quyết các yêu cầu của người sử dụng.

Hệ điều hành điều khiển và phối hợp việc sử dụng phần cứng cho những ứng dụng khác nhau của nhiều người sử dụng khác nhau. Hệ điều hành cung cấp một môi trường mà các chương trình có thể làm việc hữu hiệu trên đó.



Hình 1.1 Mô hình trừu tượng của hệ thống máy tính

1.2 Lịch sử phát triển của hệ điều hành

Thế hệ 1 (1945 – 1955)

Vào khoảng giữa thập niên 1940, Howard Aiken ở Havard và John von Neumann ở Princeton, đã thành công trong việc xây dựng máy tính dùng ống chân không. Những máy này rất lớn với hơn 10000 ống chân không nhưng chậm hơn nhiều so với máy rẻ nhất ngày nay.

Mỗi máy được một nhóm thực hiện tất cả từ thiết kế, xây dựng lập trình, thao tác đến quản lý. Lập trình bằng ngôn ngữ máy tuyệt đối, thường là bằng cách dùng bảng điều khiển để thực hiện các chức năng cơ bản. Ngôn ngữ lập trình chưa được biết đến và hệ điều hành cũng chưa nghe đến.

Vào đầu thập niên 1950, phiếu đục lỗ ra đời và có thể viết chương trình trên phiếu thay cho dùng băng điều khiển.

Thế hệ 2 (1955 – 1965)

Sự ra đời của thiết bị bán dẫn vào giữa thập niên 1950 làm thay đổi bức tranh tổng thể. Máy tính trở nên đủ tin cậy hơn. Nó được sản xuất và cung cấp cho các khách hàng. Lần đầu tiên có sự phân chia rõ ràng giữa người thiết kế, người xây dựng, người vận hành, người lập trình, và người bảo trì.

Để thực hiện một công việc (một chương trình hay một tập hợp các chương trình), lập trình viên trước hết viết chương trình trên giấy (bằng hợp ngữ hay FORTRAN) sau đó đục lỗ trên phiếu và cuối cùng đưa phiếu vào máy. Sau khi thực hiện xong nó sẽ xuất kết quả ra máy in.

Hệ thống xử lý theo lô ra đời, nó lưu các yêu cầu cần thực hiện lên băng từ, và hệ thống sẽ đọc và thi hành lần lượt. Sau đó, nó sẽ ghi kết quả lên băng từ xuất và cuối cùng người sử dụng sẽ đem băng từ xuất đi in.

Hệ thống xử lý theo lô hoạt động dưới sự điều khiển của một chương trình đặc biệt là tiền thân của hệ điều hành sau này. Ngôn ngữ lập trình sử dụng trong giai đoạn này chủ yếu là FORTRAN và hợp ngữ.

Thế hệ 3 (1965 – 1980)

Trong giai đoạn này, máy tính được sử dụng rộng rãi trong khoa học cũng như trong thương mại. Máy IBM 360 là máy tính đầu tiên sử dụng mạch tích hợp (IC). Từ đó kích thước và giá cả của các hệ thống máy giảm đáng kể và máy tính càng phổ biến hơn. Các thiết bị ngoại vi dành cho máy xuất hiện ngày càng nhiều và thao tác điều khiển bắt đầu phức tạp.

Hệ điều hành ra đời nhằm điều phối, kiểm soát hoạt động và giải quyết các yêu cầu tranh chấp thiết bị. Chương trình hệ điều hành dài cả triệu dòng hợp ngữ và do hàng ngàn lập trình viên thực hiện.

Sau đó, hệ điều hành ra đời khái niệm **đa chương**. CPU không phải chờ thực hiện các thao tác nhập xuất. Bộ nhớ được chia làm nhiều phần, mỗi phần có một công việc (job) khác nhau, khi một công việc chờ thực hiện nhập xuất CPU sẽ xử lý các công việc còn lại. Tuy nhiên khi có nhiều công việc cùng xuất hiện trong bộ nhớ, vấn đề là phải có một cơ chế bảo vệ tránh các công việc ảnh hưởng đến nhau. Hệ điều hành cũng cài đặt thuộc tính spool.

Giai đoạn này cũng đánh dấu sự ra đời của **hệ điều hành chia sẻ thời gian** như CTSS của MIT. Đồng thời các hệ điều hành lớn ra đời như MULTICS, UNIX và hệ thống các máy mini cũng xuất hiện như DEC PDP-1.

Thế hệ 4 (1980 - nay)

Máy tính dùng mạch tích hợp cỡ lớn.

Giai đoạn này đánh dấu sự ra đời của máy tính cá nhân, đặc biệt là hệ thống IBM PC với hệ điều hành MS-DOS và Windows sau này. Bên cạnh đó là sự phát triển mạnh của các hệ điều hành tựa Unix trên nhiều hệ máy khác nhau như Linux. Ngoài ra, từ đầu thập niên 90 cũng đánh dấu sự phát triển mạnh mẽ của **hệ điều hành mạng** và **hệ điều hành phân tán**.

Các hệ điều hành ra đời ở giai đoạn này như: Ubuntu, Windows 7, Windows 8,...

1.3. Phân loại hệ điều hành

1.3.1 Hệ điều hành xử lý theo lô đơn giản

Hệ điều hành thực hiện các công việc lần lượt theo những chỉ thị đã được xác định trước. Khi một công việc chấm dứt thì hệ thống sẽ tự động thực hiện công việc tiếp theo mà không cần sự can thiệp từ bên ngoài, do đó hệ thống đạt tốc độ thực hiện cao. Để thực hiện được điều này hệ điều hành phải có bộ giám sát thường trực để giám sát việc thực hiện của các công việc trong hệ thống, bộ phận này thường trú trong bộ nhớ chính. Trong hệ điều hành này khi hệ thống cần thực hiện một công việc thì nó phải lưu chương trình và dữ liệu của công việc vào hàng đợi các công việc, sau đó sẽ thực hiện lần lượt từng bộ chương trình và dữ liệu của công việc tương ứng trong hàng đợi và cho ra lần lượt các kết quả.

Với cách tổ chức công việc, thì hệ thống không thể thay đổi chương trình và dữ liệu của các công việc ngay cả khi chúng còn nằm trong hàng đợi, đây là một hạn chế. Mặt khác trong quá trình thực hiện công việc, nếu công việc chuyển sang truy xuất trên thiết bị vào/ra thì processor rơi vào trạng thái chờ, điều này gây lãng phí thời gian xử lý processor.

1.3.2 Hệ điều hành xử lý theo lô đa chương

Hệ điều hành có khả năng thực hiện nhiều công việc, nhiều chương trình đồng thời. Khi cần thực hiện nhu công việc đồng thời hệ điều hành sẽ nạp một phần code và data của các công việc vào bộ nhớ (các phần còn lại sẽ được nạp sau tại thời điểm thích hợp) và tất cả đều ở trạng thái sẵn sàng thực hiện, sau đó hệ điều hành bắt đầu

thực hiện một công việc nào đó, nhưng khi công việc đang thực hiện cần truy xuất thiết bị vào/ra thì processor sẽ được chuyển sang thực hiện các công việc khác, và cứ như thế hệ điều hành chuyển hướng processor để thực hiện hết các phần công việc trong bộ nhớ cũng như các công việc mà hệ thống yêu cầu.

Hệ điều hành này có 2 ưu điểm đó là tiết kiệm được bộ nhớ, vì không nạp hết code và data của các công việc vào bộ nhớ, và hạn chế thời gian rỗi của processor. Tuy nhiên nó phải chịu chi phí cao cho việc lập lịch processor. Ngoài ra hệ điều hành còn phải giải quyết việc chia sẻ bộ nhớ chính cho các công việc khác nhau. Ví dụ hệ điều hành MS_DOS là hệ điều hành đơn nhiệm, đa chương.

1.3.3 Hệ điều hành chia sẻ thời gian

Khái niệm chia sẻ thời gian ra đời đã đánh dấu một bước phát triển mới của hệ điều hành trong việc điều khiển các hệ thống đa người dùng. Chia sẻ thời gian ở đây chính là chia sẻ thời gian xử lý của processor cho các công việc, các tiến trình đang ở trong trạng thái sẵn sàng thực hiện.

Nguyên tắc của hệ điều hành chia sẻ thời gian tương tự như trong hệ điều hành xử lý theo lô đa chương nhưng việc chuyển processor từ công việc, tiến trình này sang công việc, tiến trình khác không phụ thuộc vào việc công việc, tiến trình hiện tại có truy xuất đến thiết bị vào/ra hay không mà chỉ phụ thuộc vào sự điều phối processor của hệ điều hành. Công việc điều phối processor của hệ điều hành rất phức tạp phụ thuộc vào nhiều yếu tố khác nhau.

Trong hệ điều hành này thời gian chuyển đổi processor giữa các công việc là rất nhỏ nên ta có cảm giác các công việc thực hiện song song với nhau. Với hệ điều hành này người sử dụng có thể yêu cầu hệ điều hành thực hiện nhiều chương trình, công việc đồng thời với nhau.

Hệ điều hành chia sẻ thời gian là mở rộng logic của hệ điều hành đa chương và nó thường được gọi là hệ điều hành đa nhiệm (Multitasking). Hệ điều hành Windows 9x/NT là các hệ điều hành đa nhiệm.

1.3.4 Hệ điều hành đa vi xử lý.

Là các hệ điều hành dùng điều khiển sự hoạt động của các hệ thống máy tính có nhiều bộ vi xử lý. Các hệ điều hành đa vi xử lý (multiprocessor) gồm có 2 loại:

- Đa xử lý đối xứng (SMP: symmetric): Trong hệ thống này vi xử lý nào cũng có thể chạy một loại tiểu trình bất kỳ, các vi xử lý giao tiếp với nhau thông qua một bộ nhớ dùng chung. Hệ SMP cung cấp một cơ chế chịu lỗi và khả năng cân bằng tải tối

ưu hơn, vì các tiểu trình của hệ điều hành có thể chạy trên bất kỳ vi xử lý nào nên nguy cơ xảy ra tình trạng tắc nghẽn ở CPU giảm đi đáng kể. Vấn đề đồng bộ giữa các vi xử lý được đặt lên hàng đầu khi thiết kế hệ điều hành cho hệ thống cho hệ thống SMP. Hệ điều hành Windows NT, Windows 2000 là các hệ điều hành đa xử lý đối xứng.

- Đa xử lý bất đối xứng (ASMP: asymmetric): Hệ điều hành dành ra một hoặc hai vi xử lý để sử dụng riêng, các vi xử lý còn lại dùng để điều khiển các chương trình của người sử dụng. Hệ AMSP đơn giản hơn nhiều so với hệ SMP, nhưng trong hệ này nếu có một vi xử lý trong các vi xử lý dành riêng cho hệ điều hành bị hỏng thì hệ thống có thể ngừng hoạt động.

1.3.5 Hệ điều hành mạng

Là các hệ điều hành dùng để điều khiển sự hoạt động của mạng máy tính. Ngoài các chức năng cơ bản của một hệ điều hành, các hệ điều hành mạng còn phải thực hiện việc chia sẻ và bảo vệ tài nguyên của mạng. Hệ điều hành Windows9x/NT, Windows 2000, Linux.

1.3.6 Hệ điều hành xử lý thời gian thực

Hệ điều hành này có khả năng cho kết quả tức thời, chính xác sau mỗi công việc.

Một hệ điều hành xử lý thời gian thực phải được định nghĩa tốt, thời gian xử lý nhanh. Hệ thống phải cho kết quả chính xác trong khoảng thời gian bị thúc ép nhanh nhất. Có hai hệ thống xử lý thời gian thực là hệ thống thời gian thực cứng và hệ thống thời gian thực mềm.

Hệ thống thời gian thực cứng là công việc được hoàn tất đúng lúc đó dữ liệu thường được lưu trong bộ nhớ ngắn hạn hay trong ROM. Việc xử lý theo thời gian thực sẽ xung đột với tất cả hệ thống liệt kê ở trên.

Dạng thứ hai là hệ thống thời gian thực mềm, mỗi công việc có một độ ưu tiên riêng và sẽ được thi hành theo độ ưu tiên đó. Có một số lĩnh vực áp dụng hữu hiệu phương pháp này là multimedia hay thực tại ảo.

1.4 Các thành phần của hệ điều hành

a)Thành phần quản lý tiến trình

Một **tiến trình** là một chương trình đang được thi hành. Một tiến trình phải sử dụng tài nguyên như thời gian sử dụng CPU, bộ nhớ, tập tin, các thiết bị nhập xuất để hoàn tất công việc của nó. Các tài nguyên này được cung cấp khi tiến trình được tạo hay trong quá trình thi hành.

Một tiến trình là hoạt động (active) hoàn toàn-ngược lại với một tập tin trên đĩa là thụ động (passive)-với một bộ đếm chương trình cho biết lệnh kế tiếp được thi hành. Việc thi hành được thực hiện theo cơ chế tuần tự, CPU sẽ thi hành từ lệnh đầu đến lệnh cuối.

Một tiến trình được coi là một đơn vị làm việc của hệ thống. Một hệ thống có thể có nhiều tiến trình cùng lúc, trong đó một số tiến trình là của hệ điều hành, một số tiến trình là của người sử dụng. Các tiến trình này có thể diễn ra đồng thời.

Vai trò của hệ điều hành trong việc quản lý tiến trình là :

- Tạo và hủy các tiến trình của người sử dụng và của hệ thống.
- Tạm dừng và thực hiện tiếp một tiến trình.
- Cung cấp các cơ chế đồng bộ tiến trình.
- Cấp phát tài nguyên cho tiến trình.
- Cung cấp các cơ chế giao tiếp giữa các tiến trình.
- Cung cấp cơ chế kiểm soát deadlock

b) Thành phần quản lý bộ nhớ chính :

Bộ nhớ là thiết bị lưu trữ duy nhất mà CPU có thể truy xuất trực tiếp. Bộ nhớ chính có thể xem như một mảng kiểu byte hay kiểu word. Mỗi phần tử đều có địa chỉ. Đó là nơi lưu dữ liệu được CPU truy xuất một cách nhanh chóng so với các thiết bị nhập/xuất. CPU đọc những chỉ thị từ bộ nhớ chính. Các thiết bị nhập/xuất cài đặt cơ chế DMA cũng đọc và ghi dữ liệu trong bộ nhớ chính. Thông thường bộ nhớ chính chứa các thiết bị mà CPU có thể định vị trực tiếp. Ví dụ CPU truy xuất dữ liệu từ đĩa, những dữ liệu này được chuyển vào bộ nhớ qua lời gọi hệ thống nhập/xuất.

Một chương trình muốn thi hành trước hết phải được ánh xạ thành địa chỉ tuyệt đối và nạp vào bộ nhớ chính. Khi chương trình thi hành, hệ thống truy xuất các chỉ thị và dữ liệu của chương trình trong bộ nhớ chính. Ngay cả khi tiến trình kết thúc, dữ liệu vẫn còn trong bộ nhớ cho đến khi một tiến trình khác được ghi chồng lên.

Hệ điều hành có những vai trò như sau trong việc quản lý bộ nhớ chính :

- Lưu giữ thông tin về các vị trí trong bộ nhớ đã được sử dụng và tiến trình nào đang sử dụng.
- Quyết định tiến trình nào được nạp vào bộ nhớ chính, khi bộ nhớ đã có thể dùng được.
- Cấp phát và thu hồi bộ nhớ khi cần thiết.

- Bảo vệ bộ nhớ

c) Thành phần quản lý bộ nhớ phụ

Bộ nhớ chính quá nhỏ để có thể lưu giữ mọi dữ liệu và chương trình, ngoài ra dữ liệu sẽ mất khi không còn được cung cấp năng lượng. Hệ thống máy tính ngày nay cung cấp **hệ thống lưu trữ phụ**. Đa số các máy tính đều dùng đĩa để lưu trữ cả chương trình và dữ liệu. Hầu như tất cả chương trình : chương trình dịch, hợp ngữ, thủ tục, trình soạn thảo, định dạng... đều được lưu trữ trên đĩa cho tới khi nó được thực hiện, nạp vào trong bộ nhớ chính và cũng sử dụng đĩa để chứa dữ liệu và kết quả xử lý. Vai trò của hệ điều hành trong việc quản lý đĩa :

- Quản lý vùng trống trên đĩa.
- Định vị lưu trữ thông tin trên đĩa.
- Lập lịch cho vấn đề ghi/đọc thông tin trên đĩa của đầu từ.

d) Quản lý hệ thống vào/ ra :

Một trong những mục tiêu của hệ điều hành là *che giấu* những đặc thù của các thiết bị phần cứng đối với người sử dụng thay vào đó là một lớp thân thiện hơn, người sử dụng dễ thao tác hơn. Một hệ thống vào/ra bao gồm :

- Thành phần quản lý bộ nhớ chứa vùng đệm (buffering), lưu trữ (caching) và spooling (vùng chứa).
- Giao tiếp điều khiển thiết bị (device drivers) tổng quát.
- Bộ điều khiển cho các thiết bị xác định.

Chỉ có bộ điều khiển cho các thiết bị xác định mới hiểu đến cấu trúc đặc thù của thiết bị mà nó mô tả.

e) Thành phần quản lý tập tin :

Máy tính có thể lưu trữ thông tin trong nhiều dạng thiết bị vật lý khác nhau : băng từ, đĩa từ, đĩa quang, ... Mỗi dạng có những đặc thù riêng về mặt tổ chức vật lý. Mỗi thiết bị có một bộ kiểm soát như bộ điều khiển đĩa (disk driver) và có những tính chất riêng. Những tính chất này là tốc độ, khả năng lưu trữ, tốc độ truyền dữ liệu và cách truy xuất.

Để cho việc sử dụng hệ thống máy tính thuận tiện, hệ điều hành cung cấp một cái nhìn logic đồng nhất về hệ thống lưu trữ thông tin. Hệ điều hành định nghĩa một đơn vị lưu trữ logic là tập tin. Hệ điều hành tạo một ánh xạ từ tập tin đến vùng thông tin trên đĩa và truy xuất những tập tin này thông qua thiết bị lưu trữ.

Một tập tin là một tập hợp những thông tin do người tạo ra nó xác định. Thông thường một tập tin đại diện cho một chương trình và dữ liệu. Dữ liệu của tập tin có thể là số, là ký tự, hay ký số.

Vai trò của hệ điều hành trong việc quản lý tập tin :

- Tạo và xoá một tập tin.
- Tạo và xoá một thư mục.
- Hỗ trợ các thao tác trên tập tin và thư mục.
- Tạo mối quan hệ giữa tập tin và bộ nhớ phụ chứa tập tin.
- Sao lưu dự phòng các tập tin trên các thiết bị lưu trữ.
- Bảo vệ tập tin khi có hiện tượng truy xuất đồng thời
- Tạo cơ chế truy xuất tập tin thông qua tên tập tin

f) Hệ thống bảo vệ :

Trong một hệ thống nhiều người sử dụng và cho phép nhiều tiến trình diễn ra đồng thời, các tiến trình phải được bảo vệ đối với những hoạt động khác. Do đó, hệ thống cung cấp cơ chế để đảm bảo rằng tập tin, bộ nhớ, CPU, và những tài nguyên khác chỉ được truy xuất bởi những tiến trình có quyền. Ví dụ, bộ nhớ đảm bảo rằng tiến trình chỉ được thi hành trong phạm vi địa chỉ của nó. Bộ thời gian đảm bảo rằng không có tiến trình nào độc chiếm CPU. Cuối cùng các thiết bị ngoại vi cũng được bảo vệ.

Hệ thống bảo vệ là một cơ chế kiểm soát quá trình truy xuất của chương trình, tiến trình, hoặc người sử dụng với tài nguyên của hệ thống. Cơ chế này cũng cung cấp cách thức để mô tả lại mức độ kiểm soát.

Hệ thống bảo vệ cũng làm tăng độ an toàn khi kiểm tra lỗi trong giao tiếp giữa những hệ thống nhỏ bên trong.

g) Thành phần thông dịch lệnh :

Một trong những phần quan trọng của hệ điều hành là hệ thống thông dịch lệnh, đó là giao tiếp giữa người sử dụng và hệ điều hành. Thành phần này chính là Shell.

Một số hệ điều hành chứa Shell trong nhân của nó, một số hệ điều hành khác thì Shell được thiết kế dưới dạng một chương trình đặc biệt.

Shell là một bộ phận hay một tiến trình đặc biệt của hệ điều hành, nó có nhiệm vụ nhận lệnh của người sử dụng, phân tích lệnh và phát sinh tiến trình mới để thực hiện yêu cầu của lệnh, tiến trình mới này được gọi là tiến trình đáp ứng yêu cầu.

Shell nhận lệnh thông qua cơ chế dòng lệnh, đó chính là nơi giao tiếp giữa người sử dụng và hệ điều hành, mỗi hệ điều hành khác nhau có cơ chế dòng lệnh khác nhau, với MS_DOS đó là con trỏ lệnh và dấu nhắc của hệ điều hành (C:\>), với Windows 9x đó là nút Start/Run. Tập tin Command.com chính là Shell của MS_DOS.

Trong môi trường đơn nhiệm, ví dụ như MS_DOS, khi tiến trình đáp ứng yêu cầu hoạt động thì Shell sẽ chuyển sang trạng thái chờ, để chờ cho đến khi tiến trình đáp ứng yêu cầu kết thúc thì Shell trở lại trạng thái sẵn sàng nhận lệnh mới.

Trong môi trường hệ điều hành đa nhiệm, ví dụ như Windows 9x sau khi phát sinh tiến trình đáp ứng yêu cầu và đưa nó vào trạng thái hoạt động thì Shell sẽ chuyển sang trạng thái sẵn sàng nhận lệnh mới, nhờ vậy Shell có khả năng khởi tạo nhiều tiến trình đáp ứng yêu cầu để nó hoạt động song song với nhau, hay chính xác hơn trong môi trường hệ điều hành đa nhiệm, người sử dụng có thể khởi tạo nhiều chương trình để nó hoạt động đồng thời với nhau.

Chú ý: hầu hết các ngôn ngữ lập trình đều hỗ trợ các công cụ để người sử dụng hay người lập trình có thể gọi Shell ngay trong các ứng dụng của họ. Khi một ứng dụng cần gọi thực hiện một chương trình nào đó thì:

Trong Assembly, các ứng dụng gọi hàm 4Bh/21h của MS_DOS.

Trong Pascal, các ứng dụng gọi thủ tục Exec

Trong VisualBasic, các ứng dụng gọi hàm/ thủ tục shell. Ví dụ dòng lệnh sau: Shell"C:\Windows\notepad.exe" có thể gọi thực hiện chương trình Notepad của Windows.

Trong Windows 9x/WindowsNT, các ứng dụng gọi hàm ShellExecute.

Chú ý: Cần phải phân biệt sự khác nhau giữa Shell và System Call. Shell tạo môi trường giao tiếp giữa người sử dụng và hệ điều hành, System call tạo môi trường giao tiếp giữa chương trình người sử dụng và hệ điều hành.

h) Thành phần quản lý mạng

Xem xét đến các vấn đề liên lạc giữa các tiến trình, chia sẻ tài nguyên chung, vấn đề bảo mật trên các tiến trình trong các hệ thống khác nhau.

1.5 Các cấu trúc của hệ thống

a) Hệ thống đơn khối (monolithic systems)

Trong hệ thống này hệ điều hành là một tập hợp các thủ tục, mỗi thủ tục có thể gọi thực hiện một thủ tục khác bất kỳ lúc nào khi cần thiết.

Thông thường hệ điều hành bắt đầu là một hệ thống nhỏ, đơn giản và có giới hạn.

Cấu trúc tối thiểu phân chia các thủ tục trong hệ thống thành 3 cấp độ:

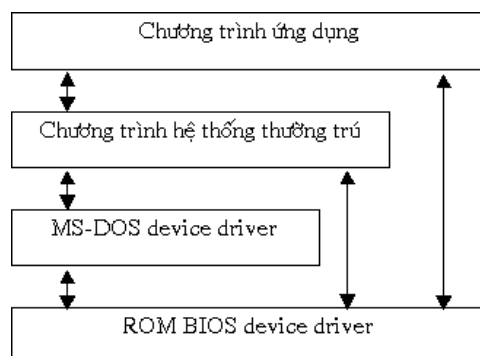
- Một chương trình chính(chương trình của người sử dụng) gọi đến một thủ tục dịch vụ của HĐH. Lời gọi này được gọi là lời gọi hệ thống (system call)
- Một tập các thủ tục dịch vụ để đáp ứng những lời gọi hệ thống từ các chương trình của người sử dụng
- Một tập các thủ tục tiện ích hỗ trợ các thủ tục dịch vụ trong việc thực hiện các lời gọi hệ thống

Nhược điểm:

- Không có sự che dấu dữ liệu, mỗi thủ tục có thể gọi đến tất cả các thủ tục khác. Chương trình ứng dụng có thể truy xuất các thủ tục cấp thấp tác động đến cả phần cứng do vậy HĐH khó kiểm soát và bảo vệ hệ thống.
- Các thủ tục dịch vụ mang tính chất tĩnh, nó chỉ hoạt động khi được gọi bởi chương trình của người sử dụng, điều này làm cho hệ điều hành thiếu chủ động trong việc quản lý môi trường

MS-DOS là một hệ điều hành có cấu trúc đơn giản, nó cung cấp những chức năng cần thiết nhất trong một không gian nhỏ nhất do sự giới hạn của phần cứng mà nó chạy trên đó và không chia thành những đơn thể rõ rệt.

Hình 1.2 Cấu trúc của MS-DOS



Mặc dù MS-DOS có cấu trúc nhưng giữa giao diện và chức năng

không có sự phân chia rõ rệt. Các chương trình ứng dụng có thể truy xuất trực tiếp các thủ tục nhập xuất cơ bản và ghi trực tiếp lên màn hình hay bộ điều khiển đĩa.

b) Các hệ thống phân lớp

Hệ thống được chia thành một số lớp, mỗi lớp được xây dựng dựa vào các lớp bên trong. Lớp trong cùng thường là phần cứng, lớp ngoài cùng là giao tiếp với người sử dụng.

Mỗi lớp là một đối tượng trừu tượng chứa đựng bên trong nó các dữ liệu và các thao tác xử lý dữ liệu đó. Lớp n chứa đựng một cấu trúc dữ liệu và các thủ tục có thể được gọi bởi lớp $n+1$ hoặc ngược lại có thể gọi các thủ tục ở lớp $n-1$.

Nhận xét:

-Hệ thống này mang tính đơn thể, nên dễ cài đặt, tìm lỗi và kiểm chứng hệ thống.

Ưu điểm là tính module. Các lớp được chọn dựa trên cơ sở lớp trên sử dụng chức năng và các dịch vụ chỉ của lớp dưới nó. Tiếp cận này đơn giản hóa việc gỡ rối và kiểm tra hệ thống. Lớp đầu tiên có thể được gỡ rối mà không có bất cứ sự quan tâm nào cho lớp còn lại của hệ thống. Bởi vì theo định nghĩa, nó chỉ sử dụng phần cứng cơ bản để cài đặt các chức năng của nó. Một khi lớp đầu tiên được gỡ rối, chức năng sửa lỗi của nó có thể được đảm đương trong khi lớp thứ 2 được gỡ rối, ...Nếu một lỗi được tìm thấy trong khi gỡ rối cho một lớp xác định, lỗi phải được nằm trên lớp đó vì các lớp bên dưới đã được gỡ rối rồi. Do đó, thiết kế và cài đặt hệ thống được đơn giản hóa khi hệ thống được phân chia thành nhiều lớp.

Mỗi lớp được cài đặt chỉ với các thao tác được cung cấp bởi các lớp bên dưới. Một lớp không cần biết các thao tác được cài đặt như thế nào; nó chỉ cần biết các thao tác đó làm gì. Do đó, mỗi lớp che giấu sự tồn tại của cấu trúc dữ liệu, thao tác và phần cứng từ các lớp cấp cao hơn.

-Các nhà thiết kế gặp khó khăn trong việc xác định số lượng lớp, thứ tự và chức năng của mỗi lớp.

Khó khăn chính của tiếp cận phân lớp liên quan tới việc định nghĩa cẩn thận các lớp vì một lớp chỉ có thể sử dụng các lớp bên dưới nó. Thí dụ, trình điều khiển thiết bị cho không gian đĩa được dùng bởi các giải thuật bộ nhớ ảo phải nằm ở cấp thấp hơn trình điều khiển thiết bị của các thủ tục quản lý bộ nhớ vì quản lý bộ nhớ yêu cầu khả năng sử dụng không gian đĩa.

Các yêu cầu có thể không thật sự rõ ràng. Thường thì các trình điều khiển lưu trữ dữ phòng nằm trên bộ định thời CPU vì trình điều khiển cần phải chờ nhập/xuất và CPU có thể được định thời lại trong thời gian này. Tuy nhiên, trên hệ thống lớn, bộ định thời có thể có nhiều thông tin hơn về tất cả quá trình đang hoạt động hơn là có thể đặt vừa trong bộ nhớ. Do đó, thông tin này có thể cần được hoán vị vào và ra bộ nhớ, yêu cầu thủ tục trình điều khiển lưu trữ dữ phòng nằm bên dưới bộ định thời CPU.

-Trong một số trường hợp lời gọi thủ tục có thể lan truyền đến các thủ tục khác ở các lớp bên trên nên chi phí cho vấn đề truyền tham số và chuyển đổi ngữ cảnh tăng lên, dẫn đến lời gọi hệ thống trong cấu trúc này thực hiện chậm hơn so với các cấu trúc khác

Vấn đề cuối cùng với các cài đặt phân lớp là chúng có khuynh hướng ít hiệu quả hơn các loại khác. Thí dụ, khi chương trình người dùng thực thi thao tác nhập/xuất, nó thực thi một lời gọi hệ thống. Lời gọi hệ thống này được bẫy (trapped) tới lớp nhập/xuất, nó yêu cầu tăng quản lý bộ nhớ, sau đó gọi tăng định thời CPU, sau đó được truyền tới phần cứng. Tại mỗi lớp, các tham số có thể được hiệu chỉnh, dữ liệu có thể được truyền,...Mỗi tầng thêm chi phí cho lời gọi hệ thống; kết quả thực sự là lời gọi hệ thống mất thời gian lâu hơn khi chúng thực hiện trên hệ thống không phân tầng.

Cấu trúc lớp này lần đầu tiên được thiết kế và áp dụng cho hệ điều hành THE (Technische Hogeschool Eindhoven). Hệ thống này được chia thành sáu lớp như hình sau:

Lớp 5	Chương trình của người sử dụng
Lớp 4	Tạo buffer cho thiết bị nhập xuất
Lớp 3	Device driver thao tác màn hình
Lớp 2	Quản lý bộ nhớ
Lớp 1	Lập lịch CPU
Lớp 0	Phần cứng

Hình 1.3 Cấu trúc của hệ điều hành THE

Các ví dụ khác như cấu trúc lớp của hệ điều hành VENUS và OS/2

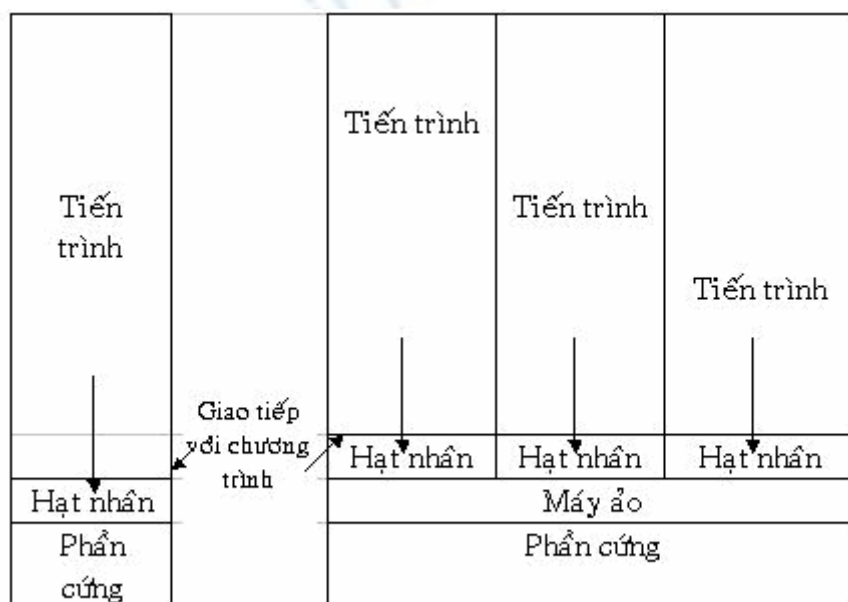
c) Máy ảo (Virtual Machine)

Các máy ảo là những bản sao ảo chính xác các đặc tính phần cứng của máy tính thực sự và cho phép hệ điều hành hoạt động trên đó như trên phần cứng thực sự. Phần nhân hệ thống thực hiện giám sát máy ảo chịu trách nhiệm giao tiếp với phần cứng, chia sẻ tài nguyên hệ thống để tạo ra nhiều máy ảo, hoạt động độc lập với nhau, để cung cấp cho lớp bên trên.

Với cấu trúc này mỗi tiến trình hoạt động trên một máy ảo độc lập và nó có cảm giác như đang sở hữu một máy tính thực sự.

Mục đích của việc sử dụng máy ảo là xây dựng các hệ thống đa chương với nhiều tiến trình thực hiện đồng thời, mỗi tiến trình được cung cấp một máy ảo với đầy đủ tài nguyên, tất nhiên là tài nguyên ảo, để nó thực hiện được.

Vấn đề phức tạp nhất của máy ảo là hệ thống đĩa. Giả sử hệ thống chỉ có ba bộ điều khiển đĩa nhưng có tới bảy máy ảo. Như vậy không thể gán cho mỗi máy ảo một bộ điều khiển đĩa và giải pháp là xây dựng hệ thống đĩa ảo.



Hình 1.4 So sánh giữa máy thực và máy ảo

Nhận xét:

- Việc cài đặt các phần mềm giả lập phần cứng để tạo ra máy ảo thường rất khó khăn và phức tạp.
- Trong hệ thống này vấn đề bảo vệ tài nguyên hệ thống và tài nguyên đã cấp phát cho tiến trình, sẽ trở nên đơn giản hơn vì mỗi tiến trình thực hiện trên một máy tính (ảo) độc lập với nhau nên việc tranh chấp tài nguyên là không thể xảy ra.

- Nhờ hệ thống máy ảo mà một ứng dụng được xây dựng trên hệ điều hành có thể hoạt động được trên hệ điều hành khác.

Trong môi trường hệ điều hành Windows 9x người sử dụng có thể thực hiện được các ứng dụng được thiết kế để thực hiện trên môi trường MS_DOS, sở dĩ như vậy vì Windows đã cung cấp cho các ứng dụng này một máy ảo DOS (VMD: Virtual Machine DOS) để nó hoạt động như đang hoạt động trong hệ điều hành DOS. Tương tự như trong môi trường hệ điều hành Windows NT người sử dụng có thể thực hiện được các ứng dụng được thiết kế trên một số hệ điều hành khác, có được điều này là nhờ cấu trúc của Windows NT có chứa các hệ thống con (subsystem) môi trường tương thích với các môi trường hệ điều hành khác như : Win32, OS/2,...các ứng dụng khi cần thiết thực hiện trên Windows NT sẽ thực hiện trong các hệ thống con môi trường tương ứng, đúng với môi trường mà ứng dụng đó được tạo ra.

d) Mô hình Client/Server

Các hệ điều hành hiện đại thường chuyển dần các nhiệm vụ của hệ điều hành ra các lớp bên ngoài nhằm thu nhỏ phần cốt lõi của hệ điều hành thành hạt nhân cực tiểu (kernel) sao cho chỉ phần hạt nhân này chỉ phụ thuộc vào phần cứng. Để thực hiện được điều này hệ điều hành xây dựng theo mô hình Client/Server, theo mô hình này hệ điều hành bao gồm nhiều tiến trình đóng vai trò server có các chức năng chuyên biệt như quản lý tiến trình, quản lý bộ nhớ,...phần hạt nhân của hệ điều hành chỉ thực hiện nhiệm vụ tạo cơ chế thông tin liên lạc giữa các tiến trình client và các tiến trình server.

Như vậy các tiến trình trong hệ thống được chia thành 2 loại:

- Tiến trình bên ngoài hay tiến trình của chương trình người sử dụng được gọi là các tiến trình client.
- Tiến trình của hệ điều hành được gọi là các tiến trình server.

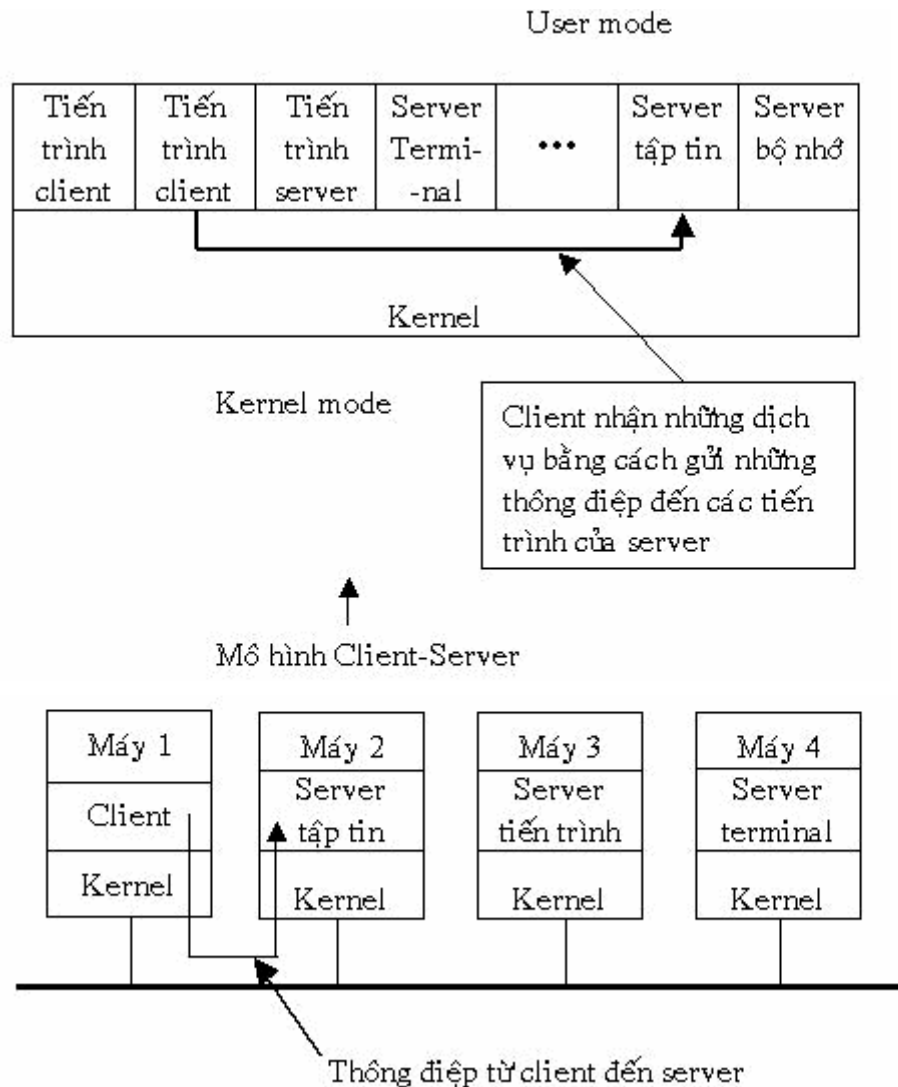
Khi cần thực hiện một chức năng hệ thống các tiến trình Client sẽ gửi yêu cầu tới tiến trình server tương ứng, tiến trình server sẽ xử lý và trả lời kết quả cho tiến trình client.

Nhận xét:

-Hệ thống này dễ thay đổi và dễ mở rộng hệ điều hành. Để thay đổi các chức năng của hệ điều hành chỉ cần thay đổi server tương ứng, để mở rộng hệ điều hành chỉ cần thêm các server mới vào hệ thống.

-Các tiến trình server của hệ điều hành hoạt động trong chế độ không đặc quyền nên không thể truy cập trực tiếp tới phần cứng, điều này giúp cho hệ thống được bảo vệ tốt hơn.

Ví dụ: WindowsNT



Hình 1.5 Mô hình Client-Server trong hệ thống phân tán

1.6 Các tính chất cơ bản của hệ điều hành

a) Tin cậy

Mọi hoạt động, mọi thông báo của HĐH đều phải chuẩn xác, tuyệt đối. chỉ khi nào biết chắc chắn là đúng thì HĐH mới cung cấp thông tin cho người sử dụng. Để đảm bảo được yêu cầu này, phần thiết bị kỹ thuật phải có những phương tiện hỗ trợ kiểm tra tính đúng đắn của dữ liệu trong các phép lưu trữ và xử lý. Trong các trường hợp còn lại HĐH thông báo lỗi và ngừng xử lý trao quyền quyết định cho người vận hành hoặc người sử dụng.

b) An toàn

Hệ thống phải tổ chức sao cho chương trình và dữ liệu không bị xoá hoặc bị thay đổi ngoài ý muốn trong mọi trường hợp và mọi chế độ hoạt động. Điều này đặc biệt quan trọng khi hệ thống là đa nhiệm. Các tài nguyên khác nhau đòi hỏi những yêu cầu khác nhau trong việc đảm bảo an toàn.

c) Hiệu quả

Các tài nguyên của hệ thống phải được khai thác triệt để sao cho ngay cả điều kiện tài nguyên hạn chế vẫn có thể giải quyết những yêu cầu phức tạp. Một khía cạnh quan trọng của đảm bảo hiệu quả là duy trì đồng bộ trong toàn bộ hệ thống, không để các thiết bị tốc độ chậm trì hoãn hoạt động của toàn bộ hệ thống.

d) Tổng quát theo thời gian

HĐH phải có tính kế thừa, đồng thời có khả năng thích nghi với những thay đổi có thể có trong tương lai. Tính thừa kế là rất quan trọng ngay cả với các hệ điều hành thế hệ mới. Đối với việc nâng cấp, tính kế thừa là bắt buộc. Các thao tác, thông báo là không được thay đổi, hoặc nếu có thì không đáng kể và phải được hướng dẫn cụ thể khi chuyển từ phiên bản này sang phiên bản khác, bằng các phương tiện nhận biết của hệ thống. Đảm bảo tính kế thừa sẽ duy trì và phát triển đội ngũ người sử dụng-một nhân tố quan trọng để HĐH có thể tồn tại. Ngoài ra người sử dụng cũng rất quan tâm, liệu những kinh nghiệm và kiến thức của mình về HĐH hiện tại còn được sử dụng bao lâu nữa. Khả năng thích nghi với những thay đổi đòi hỏi HĐH phải được thiết kế theo một số nguyên tắc nhất định.

e) Thuận tiện

Hệ thống phải dễ dàng sử dụng, có nhiều mức hiệu quả khác nhau tùy theo kiến thức và kinh nghiệm người dùng. Hệ thống trợ giúp phong phú để người sử dụng có thể tự đào tạo ngay trong quá trình khai thác.

Trong một chừng mực nào đó, các tính chất trên mâu thuẫn lẫn nhau. Mỗi HĐH có một giải pháp trung hoà, ưu tiên hợp lý ở tính chất này hay tính chất khác.

1.7 Nguyên lý xây dựng chương trình hệ điều hành

a) Module

- HĐH phải được xây dựng từ các module độc lập nhưng có khả năng liên kết thành một hệ thống có thể thu gọn hoặc mở rộng tùy ý.
- Các module đồng cấp quan hệ với nhau thông qua dữ liệu vào và ra.