

Bài giảng

PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ

(Open Source Software Development)

Tác giả: Ngô Bá Hùng - <http://ngobahung.vn/>



07-2011

MỤC LỤC

Chương 1 - Giới thiệu phần mềm mã nguồn mở	4
1.1 Phần mềm và mã nguồn phần mềm	4
1.2 Chủ sở hữu phần mềm.....	4
1.3 Giấy phép sử dụng phần mềm (License).....	4
1.3.1 Phần mềm thương mại	5
1.3.2 Phần mềm miễn phí (freeware) và phần mềm trả một phần (shareware).....	5
1.3.3 Phần mềm mã nguồn mở	5
1.4 Phòng trào phần mềm tự do	6
1.4.1 Phần mềm tự do (Free Software).....	6
1.4.2 Giấy phép sử dụng phần mềm GPL (General Public License).....	6
1.4.2.1 GNU GPL V2.....	7
1.4.2.2 LGPL.....	7
1.4 Hệ điều hành Linux.....	7
1.5 Phần mềm mã nguồn mở (Open Source Software)	8
1.6 Lợi ích của Phần mềm mã nguồn mở	9
1.7 Một số phần mềm mã nguồn mở thông dụng	10
Chương 2 – Hạt nhân Linux (Linux Kernel).....	11
2.1 Hệ điều hành Unix.....	11
2.2 Lịch sử của hệ điều hành Linux	11
2.3 Hạt nhân Linux (Linux Kernel).....	11
2.4 Hệ điều hành Linux (Linux Operating System)	12
2.5 Các thành phần của một hệ điều hành Linux	12
2.6 Kiến trúc hạt nhân Linux.....	12
2.7 Các nhóm phát triển hạt nhân Linux	13
2.8 Những khác biệt của Hạt nhân Linux so với Unix.....	14
2.9 Phiên bản hạt nhân Linux (Linux Version).....	14
2.10 Mã nguồn của hạt nhân Linux	14
2.12 Những lý do các công ty hỗ trợ cho việc phát triển Linux Kernel.....	15
Chương 3 - Hệ điều hành Linux (Linux Operating System).....	16
3.1 Hệ điều hành Linux	16
3.2 Các thành phần của một hệ điều hành Linux	16
3.3 Lý do để chọn hệ điều hành Linux	16
3.4 Làm việc trên một hệ điều hành Linux	17
3.5 Các loại tập tin.....	17
3.6 Chuẩn phân cấp hệ thống tập tin (FHS-Filesystem Hierarchy Standard).....	17
3.7 Đường dẫn (path).....	17
3.8 Một số thư mục đặc biệt	18
3.9 Một số lệnh cơ bản trên thư mục	18
3.10 Một số lệnh thao tác trên tập tin	18
3.11 Bộ thông dịch lệnh	18
3.12 Lập trình shell	19
3.12.1 Tạo một shell script.....	19
3.12.2 Biến trong shell script.....	20
3.12.3 Lệnh echo	20
3.12.4 Lệnh tính toán biểu thức toán số học.....	21
3.12.5 Các loại dấu nháy	21

3.12.6	Lệnh read	21
3.12.7	Các ký tự đại diện	22
3.12.8	Viết nhiều lệnh trên một dòng	22
3.12.9	Các thành phần của lệnh	22
3.12.10	Lệnh if	23
3.12.11	Cấu trúc lệnh if-else đơn cấp	24
3.12.12	Cấu trúc lệnh if-else đa cấp	25
3.12.13	Vòng lặp for	25
3.12.14	Vòng lặp while	26
3.12.15	Lệnh case	26
Chương 4	- Mô hình phát triển phần mềm mã nguồn mở	28
4.1	Giới thiệu	28
4.2	Mô hình phát triển phần mềm truyền thống	28
4.3	Mô hình phát triển PMMNM	28
4.4	Sự khác biệt giữa mô hình phát triển phần mềm truyền thống và PMMNM	28
4.5	Động cơ của người phát triển PMMNM	29
4.6	Môi trường phát triển PMMNM	29
4.6.1	Các kênh truyền thông	29
4.6.2	Các cơ sở dữ liệu về lỗi	29
4.6.3	Hệ thống quản lý mã nguồn (Version control)	30
4.7	Xưởng phát triển phần mềm mã nguồn mở	30
Chương 5	- Lập trình C trên Linux	31
5.1	Các công cụ cần thiết	31
5.2	Biên dịch chương trình đơn giản	31
5.3	Tập tin tiêu đề (header file)	32
5.4	Tập tin thư viện hàm	32
5.5	Tiện ích make	33
5.5.1	Giới thiệu	33
5.5.2	Tập tin mô tả	34
5.5.3	Cách thức hoạt động của make	34
5.5.4	Xây dựng tập tin mô tả	34
5.5.5	Cú pháp sử dụng lệnh make	35
5.5.6	Sử dụng macro trong tập tin mô tả	35
Chương 6	- Hệ thống quản lý phiên bản Subversion	37
6.1	Hệ thống quản lý phiên bản (Version Control System)	37
6.2	Giới thiệu Subversion	37
6.3	Lịch sử phát triển của Subversion	37
6.4	Kiến trúc của Subversion	37
6.5	Các thành phần của gói phần mềm subversion	38
6.6	Kho chứa (Repository)	38
6.7	Các mô hình quản lý phiên bản	39
6.8	Định vị tập tin thư mục	39
6.9	Phiên bản làm việc (Working copy)	40
6.10	Quản lý sự sửa đổi trên repository	40
6.11	Đồng bộ phiên bản làm việc với repository	41
6.12	Các lệnh cơ bản trên subversion	42
6.12.1	Lệnh trợ giúp - help	42
6.12.2	Đưa dữ liệu vào repository - import	42

6.12.3 Tạo phiên bản làm việc - checkout	42
6.12.4 Sửa đổi phiên bản làm việc	43
6.12.5 Xem lại những sửa đổi status.....	43
6.12.6 Phục hồi lại các sửa chữa -revert.....	43
6.12.7 Xử lý đụng độ khi cập nhật hoặc công bố.....	44
6.12.8 Xác định sự sửa đổi - commit.....	45
6.12.9 Xem lại nhật ký của repository.....	46
6.12.10 Liệt kê nội dung một thư mục trên repository – list.....	47
6.13 Giới thiệu về nhánh (Branch).....	48
6.14 Nhãn	49
Tài liệu tham khảo.....	50

TaiLieu.vn

Chương 1 - Giới thiệu phần mềm mã nguồn mở

1.1 Phần mềm và mã nguồn phần mềm

Cho một phần mềm thực hiện chức năng cộng 2 số nguyên và in kết quả ra màn hình như sau:

Thực thi: cong.exe 1 2
Kết quả: 1+2=3

Giả sử phần mềm (hay còn gọi là chương trình) cong.exe là do lập trình viên Tèo phát triển bằng ngôn ngữ lập trình C.

Đầu tiên Tèo viết tập tin cong.c có nội dung như sau:

```
main(int argc, char *argv[]) {  
    int a= atoi(argv[1]); int b= atoi(argv[2]);  
    printf("%d+%d=%d\n",a,b,a+b);  
}
```

Sau đó Tèo sử dụng một trình biên dịch để biên dịch tập tin cong.c thành tập tin cong.exe.

Trong ví dụ này:

- Tập tin cong.exe được gọi là một phần mềm, hay một chương trình phần mềm. Nội dung của cong.exe bao gồm các mã máy, mã thực thi hay mã nhị phân, là các chỉ thị mà máy tính phải thực hiện.
- Tập tin cong.c được gọi là mã nguồn của phần mềm cong.exe
- Anh Tèo được gọi là chủ sở hữu của phần mềm (cả 2 tập tin, cong.exe và cong.c)

1.2 Chủ sở hữu phần mềm

Khi một phần mềm được tạo ra nó thuộc một chủ sở hữu nào đó. Chủ sở hữu có thể là một cá nhân (lập trình viên viết ra phần mềm) hoặc là một công ty phần mềm (người bỏ tiền ra thuê mướn lập trình viên trực tiếp viết phần mềm cho công ty). Chủ sở hữu phần mềm có toàn quyền trên phần mềm mà họ là chủ sở hữu, và sẽ quyết định mức độ sử dụng và khai thác của những người khác trên phần mềm mà họ là chủ sở hữu. Khi muốn sử dụng một phần mềm đó, người sử dụng phải xin phép chủ sở hữu phần mềm thông qua một giấy phép được cấp bởi chủ sở hữu phần mềm.

1.3 Giấy phép sử dụng phần mềm (License)

Được chủ sở hữu phần mềm cấp cho người muốn sử dụng phần mềm. Nó là một bản hợp đồng gồm các điều khoản và điều kiện, mô tả những gì mà chủ sở hữu phần mềm cho phép bạn khai thác phiên bản phần mềm liên quan. Nó qui định về những khả năng mà bạn có thể có được trên phần mềm mà bạn được cấp giấy phép sử dụng.

Xét giấy phép của một số loại phần mềm phổ biến sau

1.3.1 Phần mềm thương mại

Giấy phép sử dụng của phần mềm thương mại chỉ cho phép người sử dụng khai thác phần mềm theo những ràng buộc đã ghi rõ trong giấy phép. Chẳng hạn như không cho phép người sử dụng cài đặt phần mềm trên nhiều máy khác nhau. Bản quyền loại này rất bị hạn chế. Trong trường hợp có những lỗi phần mềm được phát hiện hay một số chức năng hoạt động không tốt thì người sử dụng không còn cách nào khác hơn là phải chờ cho đến khi chủ sở hữu phần mềm sửa đổi chúng. Các nhà sản xuất phần mềm đôi khi không sẵn lòng làm việc đó hoặc thực hiện chúng với thời gian rất lâu hay đôi khi người sử dụng phải trả thêm tiền cho các bản cập nhật. Người sử dụng không có một phương tiện nào để thúc đẩy tiến trình cập nhật và sửa chữa lỗi của các phần mềm thương mại.

1.3.2 Phần mềm miễn phí (freeware) và phần mềm trả một phần (shareware)

Là các phần mềm có chủ sở hữu. Được phân phối một cách tự do. Phần mềm miễn phí không đòi hỏi tiền bản quyền sử dụng phần mềm. Phần mềm trả một phần thì sau một khoản thời gian đã định người sử dụng phải trả tiền nếu như muốn được phép sử dụng tiếp. Cả hai loại phần mềm này đều không cho phép người sử dụng truy cập vào mã nguồn của phần mềm.

1.3.3 Phần mềm mã nguồn mở

Giấy phép phần mềm lại này quy định rằng nó được phân phối đến người sử dụng cùng với mã nguồn của nó mà chúng có thể bị sửa đổi. Nó có thể được phân phối lại mà không bị một ràng buộc nào khác.

Chúng ta có thể phân phối cả những thay đổi mà chúng ta đã thực hiện trên mã nguồn gốc

Các điều khoản, điều kiện mô tả trong các giấy phép sử dụng phần mềm khác nhau là khác nhau. Ở đây ta xem xét các điều khoản liên quan đến 3 khả năng sau đối với người sử dụng :

- Khả năng phân phối lại (Distribution Possibility): Quyền được phép sao chép và phân phối lại phiên bản phần mềm mà bạn đang có trong tay (có giấy phép sử dụng nó) hay không ?
- Khả năng truy cập vào mã nguồn (Accessibility to source code): Chủ sở hữu phần mềm cho phép bạn xem mã nguồn, sử dụng, sửa đổi mã nguồn phần mềm của họ cho mục đích của bạn hay không ?
- Phí sử dụng phần mềm (Free): Khi bạn sử dụng một phần mềm, bạn phải trả tiền hay không cho người chủ sở hữu phần mềm đó ?

Bảng sau cho thấy các khả năng này trên giấy phép của một số loại phần mềm thông dụng:

	Khả năng phân phối lại	Truy cập vào mã nguồn	Miễn phí
Phần mềm thương mại (Commercial Software)	Không	Không	Không
Phần mềm miễn phí (Freeware)	Đa số	Không	Có
Phần mềm trả một phần (Shareware)	Đôi khi	Không	Không
Phần mềm mã nguồn mở (Open Source Software)	Được phép	Được phép	Đa số

1.4 Phòng trào phần mềm tự do

- Nhằm tạo ra những Phần mềm tự do (free software) là những phần mềm mà người dùng có thể tự do chia sẻ, nghiên cứu và sửa đổi chúng.
- Được khởi xướng bởi Richard M. Stallman vào năm 1983 khi ông bắt đầu dự án GNU, viết tắt của "GNU is Not UNIX".
- Nhằm thay thế hệ điều hành Unix với tính năng tự do
- Thành lập quỹ phần mềm tự do (FSF - Free Software Foundation) năm 1985

1.4.1 Phần mềm tự do (Free Software)

Phần mềm tự do (PMTD) đề cập đến sự do, không đề cập đến vấn đề chi phí/giá cả: "free" as in "free speech," not as in "free beer."

Một phần mềm được gọi là phần mềm tự do nếu giấy phép sử dụng của nó cho phép người sử dụng phần mềm có 4 khả năng tự do sau:

- Tự do thực thi chương trình cho bất kỳ mục đích gì
- Tự do nghiên cứu cách thực thi của chương trình và sửa đổi chúng cho mục đích của bạn. Truy cập vào mã nguồn chương trình là tiền đề
- Tự do phân phối phần mềm cho người khác
- Tự do cải tiến chương trình và phân phối cải tiến của bạn cho cộng đồng. Truy cập vào mã nguồn chương trình là tiền đề

1.4.2 Giấy phép sử dụng phần mềm GPL (General Public License)

Thông thường, các phần mềm đều được copyright nhằm bảo vệ quyền tác giả. Richard M. Stallman đưa ra khái niệm Copyleft là một phương pháp tổng quát nhằm làm cho một chương

trình tự do và yêu cầu tất cả những phiên bản sửa đổi hay mở rộng của chương trình cũng phải tự do. Khái niệm Copyleft được cụ thể hóa trong giấy phép «GNU General Public License», Viết tắt «GNU GPL». Đây là giấy phép cho phần mềm tự do, được phát hành cho phần lớn các sản phẩm của dự án GNU.

Các phiên bản của GPL

Version 1 – General Public License – GPL v1, 1989

Version 2 – General Public License – GPL v2, 1991

Version 2 – Library General Public License – LGPL v2, 1991

Version 2.1 – Lesser General Public License – LGPL v2.1, 1999

Version 3 – GPLv3, 2007

1.4.2.1 GNU GPL V2

Công bố tại địa chỉ <http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>, tạm dịch tiếng việt tại địa chỉ <http://vi.wikisource.org/wiki/GPL>

Một số qui định đáng lưu ý trong GPL Version 2:

- Có thể bán mã thực thi tạo ra từ phiên bản sửa đổi Tuy nhiên mã nguồn phải công bố
- Mã nguồn của sản phẩm và tất cả các sửa đổi sau đó phải tồn tại dưới dạng phần mềm tự do
- Tất cả các chương trình có sử dụng mã nguồn GPL phải phát hành dưới giấy phép GPL
- Liên kết động hay tĩnh đến mã nguồn hoặc thư viện GPL
- Sao chép một số dòng của mã nguồn GPL

1.4.2.2 LGPL

Công bố tại địa chỉ <http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html>, tạm dịch tiếng việt tại địa chỉ <http://vi.wikisource.org/wiki/LGPL>

- Được tạo ra để cho phép liên kết động mã nguồn không phát hành dưới dạng GPL hoặc LGPL vào mã nguồn LGPL
- Dàn xếp việc sử dụng các thư viện tự do vào mục đích thương mại, ví dụ thư viện GNU C
- Hầu hết các điều khoản và điều kiện tương tự GPL
- Nếu bạn thay đổi và phân phối một thư viện LGPL
 - Thư viện và những thay đổi phải được công bố (mã thực thi và mã nguồn cùng với chú thích về những sửa đổi)
 - Bằng sáng chế được gắn với sự phân phối những sửa đổi

1.4 Hệ điều hành Linux

Linux là hạt nhân (kernel) của hệ điều hành được tạo ra bởi Linus Torvalds năm 1991. Linux được phát hành dưới giấy phép GNU/GPL vào năm 1992. Linux kết hợp với các tiện ích/thư viện tạo ra từ dự án GNU tạo thành hệ điều hành GNU/Linux. Những phiên bản đầu tiên là Debian và Slackware được phát hành vào năm 1993

Vào cuối năm 1990, Linus Torvalds, sinh viên đại học Helsinki, Phần Lan cố gắng phát triển các phần mềm giống như hệ thống UNIX để sử dụng cho máy tính cá nhân 386 với bộ nhớ 4Mbytes, đĩa cứng 40 Mbytes của anh ta. Anh ta tích hợp vào hệ thống mới những kết quả mà anh ta đã thực hiện từ năm 1984 trong dự án của tổ chức phần mềm tự do (Free Software Foundation). Linus cảm nhận được chất lượng của các công việc được thực hiện bởi các lập trình viên trên toàn thế giới trong khuôn khổ của dự án GNU, vì thế đã quyết định chuyển sản phẩm của mình dưới bản quyền GPL. Anh hi vọng hệ điều hành của mình cũng được phát triển như thế. Dự án được đẩy mạnh nhanh chóng nhờ sự cộng tác của rất nhiều lập trình viên dưới sự điều phối của Linus. Cho đến thời điểm hiện nay, Linux đã tích hợp hầu hết các tính năng của một hệ điều hành hiện đại và được sự tin tưởng của nhiều người sử dụng

1.5 Phần mềm mã nguồn mở (Open Source Software)

Từ free trong tên gọi của phần mềm tự do (Free Software) làm cho người ta liên tưởng đến một loại phần mềm miễn phí hơn là 4 yếu tố tự do của nó và phần mềm tự do thì không thể làm thương mại được. Để tránh sự hiểu lầm này Eric Raymond and Bruce Perens đề xuất một tên gọi khác là Phần mềm mã nguồn mở (Open Source Software) khi thành lập Sáng kiến mã nguồn mở (Open Source Initiative) vào năm 1998. Sáng kiến này đưa ra 10 tiêu chí để đánh giá xem một giấy phép sử dụng phần mềm có đạt chuẩn là một phần mềm mã nguồn mở hay không.

- Tiêu chí (1): Tự do phân phối lại (Free Redistribution)

Bản quyền sẽ không hạn chế bất cứ ai bán hoặc cho phần mềm; và không đòi hỏi tiền bản quyền hay một chi phí nào cho thương vụ này.

- Tiêu chí (2): Mã nguồn (Source Code)

Chương trình phải được phân phối cùng với mã nguồn được công bố bằng những phương tiện công cộng mà người ta có thể lấy được mã nguồn với một chi phí sao chép hợp lý nhất

- Tiêu chí (3): Sản phẩm kế thừa (Derived Works)

Giấy phép phải công nhận những sửa đổi và những sản phẩm kế thừa; và phải cho phép chúng được phân phối với cùng những điều khoản như giấy phép của phần mềm ban đầu

- Tiêu chí (4) Tính toàn vẹn của mã nguồn của tác giả (Integrity of The Author's Source Code)

Giấy phép có thể ngăn cản việc phân phối mã nguồn dưới dạng bị sửa đổi chỉ khi giấy phép chấp nhận sự phân phối các tập tin vá lỗi (patch file) với mã nguồn vì mục đích sửa đổi chương trình tại thời điểm xây dựng (built time) chương trình. Giấy phép phải cho phép một cách tường minh việc phân phối phần mềm tạo ra từ mã nguồn bị sửa đổi. Giấy phép có thể yêu cầu những sản phẩm kế thừa phải mang một cái tên khác hoặc số phiên bản khác so với phần mềm gốc.

- Tiêu chí (5): Không phân biệt đối xử giữa các cá nhân và các nhóm (No Discrimination Against Persons or Groups)
- Tiêu chí (6): Không phân biệt đối xử với mục đích sử dụng (No Discrimination Against Fields of Endeavor)
- Tiêu chí (7): Phân phối giấy phép (Distribution of License)

Những quyền được kèm với chương trình phải được áp dụng đối với tất cả những người mà sau đó chương trình được phân phối lại mà không cần thiết phải thực thi thêm những giấy phép phụ của những thành phần này

- Tiêu chí (8): Giấy phép không được dành riêng cho một sản phẩm (License Must Not Be Specific to a Product)
Những quyền được kèm theo chương trình thì không bị phụ thuộc vào việc chương trình là thành phần của một bản phân phối phần mềm cụ thể. Nếu phần mềm được rút trích từ bản phân phối đó và được sử dụng hoặc phân phối lại với những điều khoản của giấy phép của chương trình thì tất cả các bên mà chương trình được phân phối đến cũng nên có được các quyền lợi ngang bằng như những quyền lợi được đưa ra theo bản phân phối phần mềm gốc.
- Tiêu chí (9): Giấy phép không được cản trở phần mềm khác (License Must Not Restrict Other Software)
Giấy phép không được đặt những hạn chế lên những phần mềm khác cùng được phân phối với phần mềm của giấy phép này. Ví dụ, giấy phép không được khẳng định rằng tất cả các phần mềm khác được phân phối trên cùng một phương tiện thì phải là phần mềm mã nguồn mở
- Tiêu chí (10): Giấy phép phải trung lập về mặt công nghệ (License Must Be Technology-Neutral)
Không có sự dục trù nào của giấy phép dành cho một công nghệ riêng hay một kiểu giao diện nào đó

OSI duy trì danh sách các giấy phép đạt tiêu chuẩn PMMN: 66 giấy phép (18/07/2009)
Black Duck Software cập nhật thường xuyên 20 giấy phép mã nguồn mở được dùng nhiều nhất

Một định nghĩa khác về Phần mềm mã nguồn mở cũng khá thú vị của Dale Mosby; IBM Linux Technology Center; 15-Feb-07: Phần mềm mã nguồn mở là phần mềm dưới dạng mã nguồn mà nó thường được tạo ra bởi một cộng đồng ảo, cộng tác trên Internet và thường được tải về miễn phí từ Internet hoặc được phân phối dưới dạng các đĩa CD-ROM với một giá không đáng kể. Tác giả giữ bản quyền (copyright) đối với mã nguồn và phân phối mã nguồn dưới một giấy phép định nghĩa những gì bạn được làm (hoặc không được làm) đối với mã nguồn.

1.6 Lợi ích của Phần mềm mã nguồn mở

Phần mềm mã nguồn mở được phát triển bởi một cộng đồng nhiều người nhờ đó có thể tìm ra các lỗi một cách dễ dàng. Đây chính là điểm mạnh nhất của phần mềm mã nguồn mở. Mỗi người, với khả năng có hạn của mình có thể xem xét và cải tiến các công việc được thực hiện bởi những người bạn khác. Mỗi thành viên chỉ tập trung vào phần thuộc lĩnh vực chuyên sâu của mình. Năm trăm lập trình viên làm việc với thời gian khác nhau, mỗi người tập trung vào lĩnh vực chuyên sâu của mình thì sẽ tốt hơn năm mươi lập trình viên làm việc toàn thời gian. Cách phân phối của phần mềm mã nguồn mở giúp nhiều người có điều kiện tiếp cận với chúng hơn. Nhất là đối với các nước đang phát triển, nơi mà giá phần mềm dành cho phần bảo trì, bảo hành luôn là gánh nặng.

1.7 Một số phần mềm mã nguồn mở thông dụng

Tham khảo các tài liệu sau:

- Các phần mềm tương ứng trong Windows và Ubuntu, Dịch từ trang tài liệu công đồng Pháp ngữ Ubuntu-fr, Người dịch : Vũ Đỗ Quỳnh, Hà Nội – tháng 02/2008
- OPEN SOURCE GOD: 480+ Open Source Applications
<http://mashable.com/2007/09/23/open-source/>

TaiLieu.vn

Chương 2 – Hạt nhân Linux (Linux Kernel)

2.1 Hệ điều hành Unix

Được phát triển bởi Dennis Ritchie and Ken Thompson, các lập trình viên của Bell Lab vào năm 1969 từ hệ điều hành đa người dùng Multics. Năm 1973 được viết lại hoàn toàn bằng ngôn ngữ C

Version 6 được sử dụng rộng rãi ngoài Bell Lab.

Có nhiều dòng Unix

- Bell Labs: Unix System III năm 1977, hỗ trợ nhiều chủng loại máy tính
- AT&T: System V năm 1983
- University of California at Berkeley:
- 3BSD năm 1979, 4.3 BSD thêm vào Bộ nhớ ảo, quản lý phân trang, TCP/IP
- 4.4BSD năm 1993, thương mại hóa
- Darwin, Dragonfly BSD, FreeBSD, NetBSD, và OpenBSD
- AT&T và BSD dùng nhiều trong thương mại

Điểm mạnh của hệ điều hành Unix là:

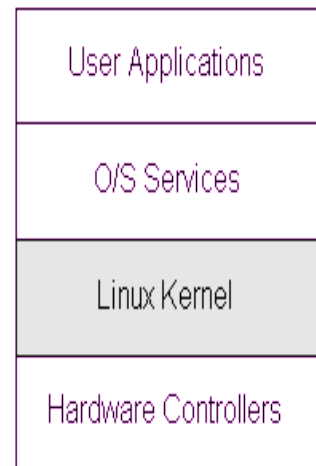
- Thiết kế đơn giản, trong sáng chỉ vài trăm lời gọi hệ thống
- Tất cả đều là tập tin, giúp đơn giản hóa thao tác xử lý dữ liệu và xuất nhập
- Hỗ trợ việc tạo tiến trình nhanh
- Cung cấp cơ chế giao tiếp liên quá trình hiệu quả
- Dễ dàng tạo ra các công cụ nhỏ, đơn giản «Do one thing and do it well»
- Dễ dàng tích hợp nhiều công cụ nhỏ để hoàn thành các tác vụ phức tạp

2.2 Lịch sử của hệ điều hành Linux

Linus Torvalds, sinh viên đại học Helsinki - Phần lan cần một Hệ điều hành có các tính năng như Unix, miễn phí, dễ dàng sửa đổi và phân phối lại mã nguồn để sử dụng cho máy tính cá nhân 386 với bộ nhớ 4Mbytes, đĩa cứng 40 Mbytes. Vì thế anh ta đã tiến hành viết một Terminal emulator nối kết vào hệ thống Unix; Sau đó anh đã tích hợp vào hệ thống mới những kết quả mà anh ta đã thực hiện từ năm 1984 trong dự án của tổ chức phần mềm tự do để tạo thành một hệ điều hành hoành chính và công bố lên Internet năm 1991.

2.3 Hạt nhân Linux (Linux Kernel)

Là phần cốt lõi nhất của một hệ điều hành, được tạo ra bởi Linus Torvald, 1991, phát hành dưới license GPL. Linux chỉ là một thành phần của hệ điều hành, thành phần hạt nhân (Kernel), cốt lõi nhất của một hệ điều hành có nhiều vụ: Trừu tượng hóa các thiết bị phần cứng, giới thiệu một máy ảo cho các chương trình người dùng; Hỗ trợ đa nhiệm (multi tasking) và hỗ trợ giao tiếp liên quá trình.

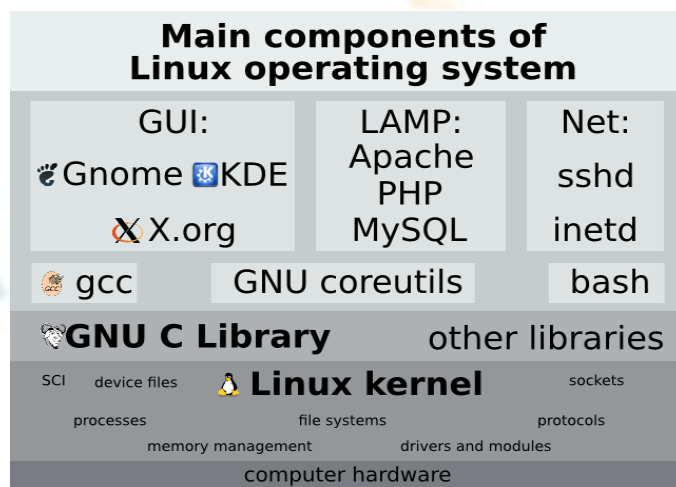


2.4 Hệ điều hành Linux (Linux Operating System)

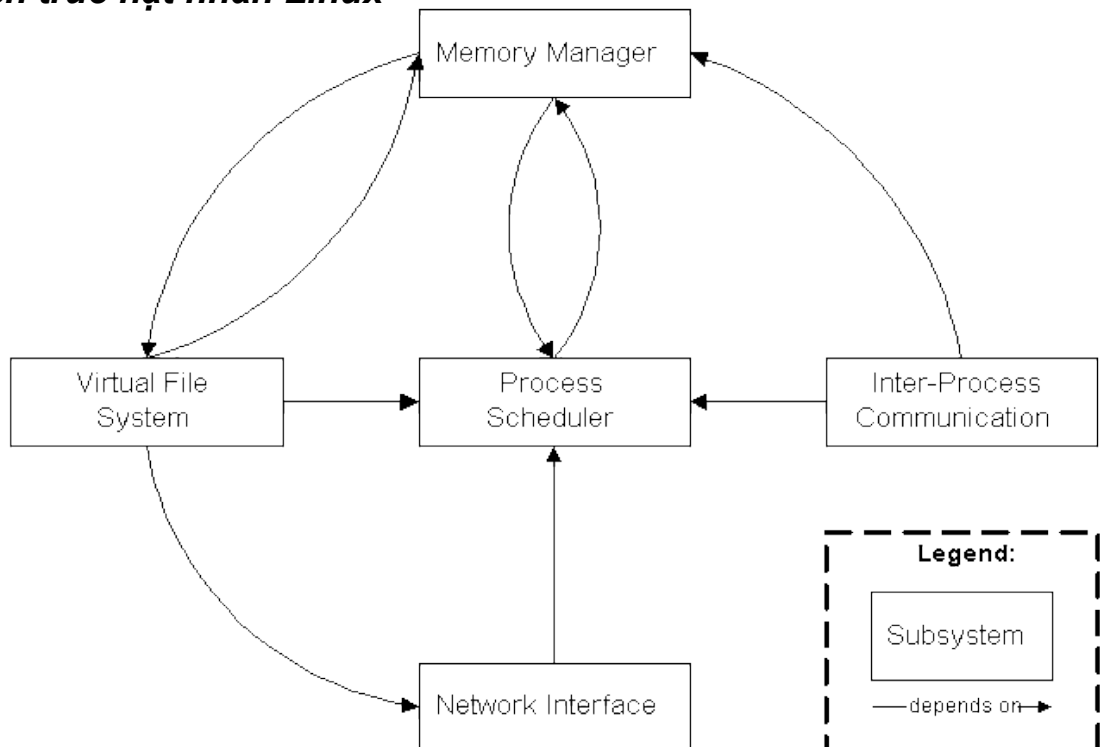
- Là các hệ điều hành sử dụng hạt nhân Linux
 - Được gọi với tên Bản phân phối Linux (Linux Distribution), gọi tắt là Linux Distro
 - Được phát hành bởi các nhà phân phối hệ điều hành (Linux Distributor)
- Có hơn 500 bản phân phối Linux. 10 bản phân phối phổ biến nhất năm 2010 gồm Ubuntu, Fedora, OpenSuSe, Debian, Mandriva, LinuxMint, PCLinuxOS, Slackware, Gentoo Linux, CentOS.

2.5 Các thành phần của một hệ điều hành Linux

Một hệ điều hành Linux thường bao gồm các thành phần sau: Hạt nhân Linux, trình điều khiển thiết bị, bộ khởi động, cửa sổ lệnh hoặc giao diện người dùng đồ họa, các tiện ích về tập tin và hệ thống...



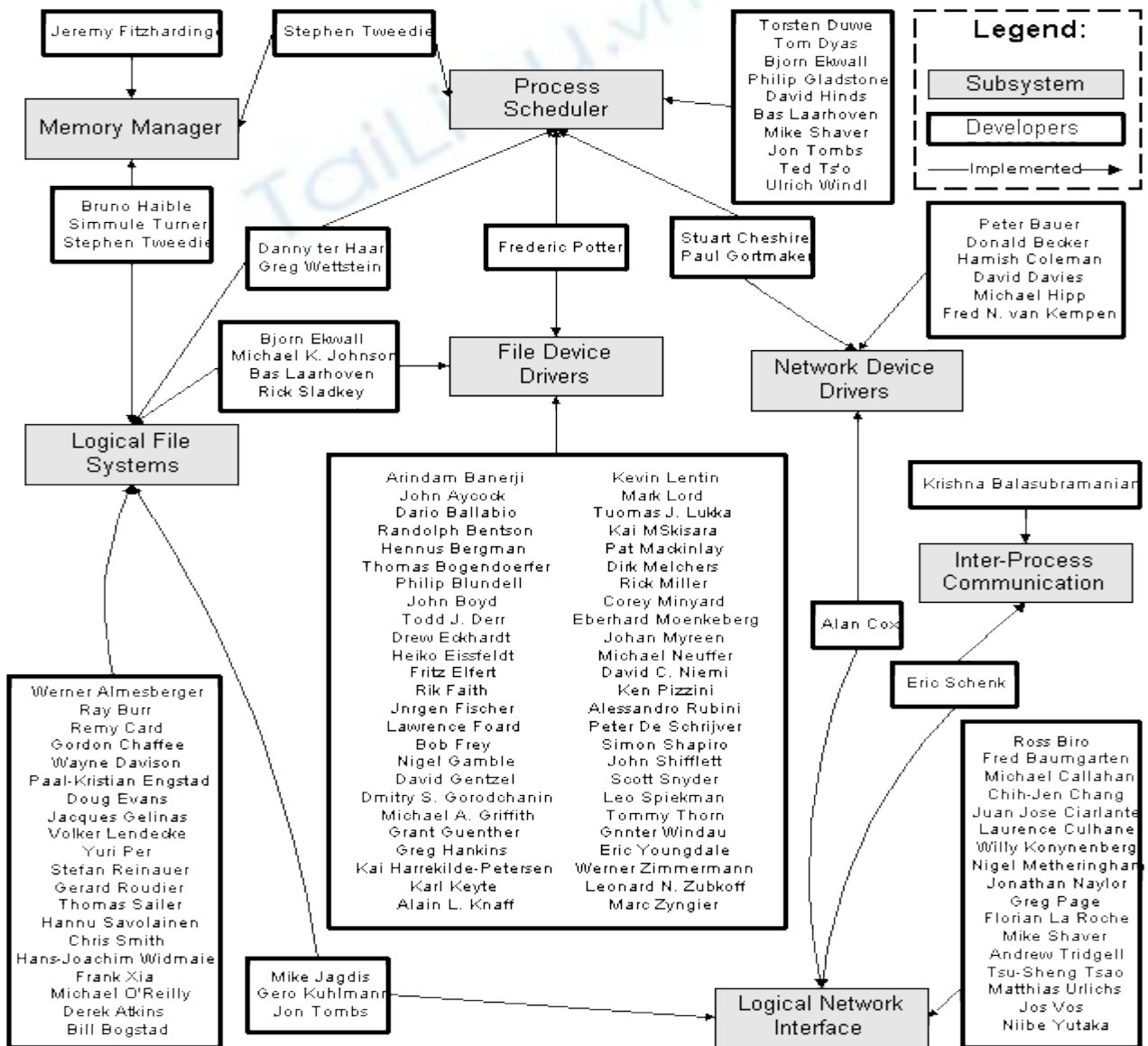
2.6 Kiến trúc hạt nhân Linux



Hạt nhân Linux gồm 5 thành phần cơ bản sau:

- Bộ định thời : Điều khiển việc truy cập đến CPU
- Bộ quản lý bộ nhớ: Đảm bảo nhiều tiến trình cùng sử dụng bộ nhớ máy tính một cách an toàn; cung cấp cơ chế bộ nhớ ảo
- Hệ thống tập tin trừu tượng: Trừu tượng hóa những chi tiết khác biệt của các loại thiết bị bằng cách giới thiệu một giao diện tập tin chung cho tất cả các thiết bị
- Giao diện mạng: Cung cấp truy cập đến nhiều chuẩn mạng và những loại thiết bị mạng khác nhau.
- Giao tiếp liên quá trình: Hỗ trợ cơ chế giao tiếp giữa các tiến trình trên cùng một máy tính

2.7 Các nhóm phát triển hạt nhân Linux



2.8 Những khác biệt của Hạt nhân Linux so với Unix

- Hỗ trợ nạp động các modul của kernel
- Hỗ trợ đa bộ xử lý đồng bộ (Symmetrical MultiProcessor)
- Là kernel theo kiểu trung dụng (Preemptive)
- Hỗ trợ đa luồng
- Hỗ trợ mô hình thiết bị hướng đối tượng, gắn nóng, hệ thống tập tin trên không gian người dùng
- Linux là tự do (Free)

2.9 Phiên bản hạt nhân Linux (Linux Version)

Có hai loại phiên bản Linux kernel: Phiên bản ổn định (Stable) và phiên bản phát triển (Development). Stable (ổn định) là phiên bản ở mức sản phẩm phù hợp cho việc triển khai rộng rãi. Development (phát triển) là phiên bản thử nghiệm với nhiều cải tiến được đưa vào.

Tiến trình phát triển các phiên bản diễn ra như sau:

- Đầu tiên các tính năng mới được tạo ra và thêm vào phiên bản Development của Linux kernel.
- Qua thời gian phiên bản development này được trưởng thành, và đến thời điểm tuyên bố đóng băng các tính năng: không cho thêm mới tính năng, chỉ cho chỉnh sửa tính năng đã có.
- Khi phiên bản development được xem là ổn định mã nguồn sẽ được đóng băng: chỉ chấp nhận các hiệu chỉnh lỗi.
- Phiên bản phát triển sẽ được phát hành như phiên bản stable đầu tiên của chuỗi phiên bản stable mới.

2.10 Mã nguồn của hạt nhân Linux

Mã nguồn của Linux Kernel có thể download từ địa chỉ <http://www.kernel.org>; Sau đó giải nén bằng lệnh `tar xvjf linux-x.y.z.tar.bz2` hoặc `tar xvzf linux-x.y.z.tar.gz`.

Patch là đơn vị mã nguồn dùng để trao đổi trong cộng đồng phát triển, phân phối những thay đổi trên mã nguồn hay nâng cấp version mà không download toàn bộ mã nguồn version mới

2.11 Tình hình phát triển hạt nhân Linux

Để trả lời cho câu hỏi về tình hình phát triển hạt nhân Linux như Nó được phát triển nhanh như thế nào ? Ai đang phát triển nó ? Họ phát triển những gì ? và ai tài trợ cho việc phát triển Hạt nhân Linux? các tác giả Greg Kroah-Hartman, SuSE Labs / Novell Inc.; Jonathan Corbet, LWN.net và Amanda McPherson, The Linux Foundation đã thực hiện một báo vào tháng 8 năm 2009. Một số thông tin đáng chú ý từ báo cáo này như sau:

- Về bức tranh tổng thể: Từ năm 2005, hơn 5000 các nhà phát triển của gần 500 công ty tham gia vào việc xây dựng Linux kernel. Từ năm 2008 đến 2009, số người tham gia phát triển tăng 10% cho mỗi phiên bản, số lượng mã nguồn thêm vào kernel mỗi ngày tăng gần 3 lần. Quan đây cho thấy một cộng đồng phát triển vững mạnh về cả số lượng và năng suất.

- Về mô hình phát triển: Dựa trên « loose, time-based release model», Nhanh chóng đưa các tính năng mới đến cho người dùng, giảm sự cách biệt giữa các phiên bản. Tần suất phát hành: 81 ngày/phiên bản. Tỷ lệ sửa đổi : 3.83 patch/giờ (từ 2.6.11 đến 2.6.30). Kích thước mã nguồn: thêm 10,923, xóa 5,547 và thay đổi 2,243 dòng / ngày (tính từ version 2.6.24). Version 2.6.30 có 27,911 file gồm 11,560,971 dòng lệnh
- Về mức độ đóng góp của các cá nhân vào việc phát triển mã nguồn cho Linux kernel: Đưa ra một danh sách 30 người tham gia sửa đổi Linux Kernel nhiều nhất. Bảng dưới đây chỉ trích 10 người đầu tiên trong danh sách:

Name	Number of Changes	Percent of Total
David S. Miller	2,239	1.5%
Ingo Molnar	2,125	1.5%
Al Viro	1,981	1.4%
Adrian Bunk	1,883	1.3%
Takashi Iwai	1,801	1.2%
Bartlomiej Zolnierkiewicz	1,651	1.1%
Ralf Baechle	1,471	1.0%
Tejun Heo	1,457	1.0%
Stephen Hemminger	1,408	1.0%
Andrew Morton	1,370	0.9%

- Về mức độ đóng góp của các công ty vào việc phát triển mã nguồn cho Linux kernel: Đưa ra một danh sách 30 công ty đóng góp vào việc phát triển Linux Kernel nhiều nhất. Bảng dưới đây chỉ trích 10 người đầu tiên trong danh sách:

Company	Number of Changes	Percent of Total
None	26,644	18.2%
Red Hat	17,981	12.3%
Unknown	11,164	7.6%
IBM	11,151	7.6%
Novell	11,046	7.6%
Intel	7,782	5.3%
Consultant	3,657	2.5%
Oracle	3,513	2.4%
Linux Foundation	2,345	1.6%
SGI	2,317	1.6%

2.12 Những lý do các công ty hỗ trợ cho việc phát triển Linux Kernel

- Để Linux có thể chạy trên phần cứng của họ và thu hút người dùng Linux: IBM, Intel, SGI, MIPS, Freescale, HP, Fujitsu, etc...
- Để chứng tỏ khả năng của họ để thu hút khách hàng sử dụng bản phân phối của họ: Red Hat, Novell, và MontaVista,...
- Vì Linux là một thành phần trong các sản phẩm (video, tele set, mobilphone) của họ: Sony, Nokia, and Samsung
- Để xây dựng ứng dụng trên nền Linux và họ muốn phiên bản mới tiếp tục hỗ trợ ứng dụng của họ

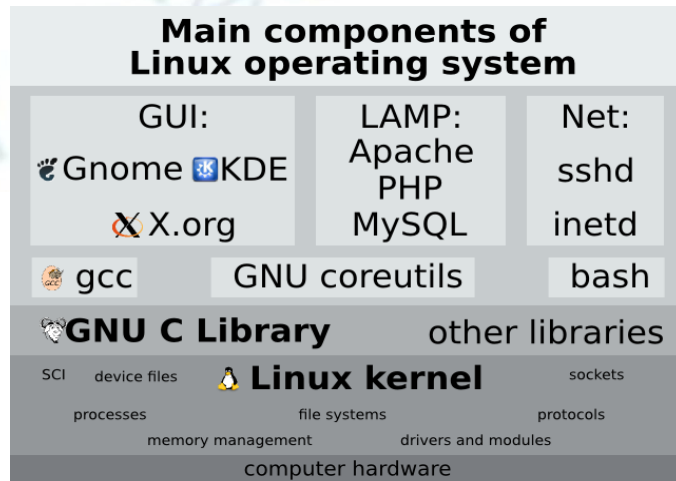
Chương 3 - Hệ điều hành Linux (Linux Operating System)

3.1 Hệ điều hành Linux

- Là các hệ điều hành sử dụng hạt nhân Linux.
- Được gọi với tên Bản phân phối Linux (Linux Distribution), gọi tắt là Linux Distro
- Được phát hành bởi các nhà phân phối hệ điều hành (Linux Distributor)
 Có hơn 500 bản phân phối Linux. 10 bản phân phối phổ biến nhất năm 2010 gồm Ubuntu, Fedora, OpenSuSe, Debian, Mandriva, LinuxMint, PCLinuxOS, Slackware, Gentoo Linux, CentOS.

3.2 Các thành phần của một hệ điều hành Linux

Một hệ điều hành Linux thường bao gồm các thành phần sau: Hạt nhân Linux, trình điều khiển thiết bị, bộ khởi động, cửa sổ lệnh hoặc giao diện người dùng đồ họa, các tiện ích về tập tin và hệ thống...



3.3 Lý do để chọn hệ điều hành Linux

- Ứng dụng: Nhiều ứng dụng sẵn dùng trên Linux (miễn phí lẫn thương mại): văn bản, đồ họa, đa phương tiện, Internet, bảo mật, quản trị, máy chủ ...
- Ngoại vi: Hỗ trợ nhiều chủng loại thiết bị ngoại vi, hỗ trợ nhanh chóng các thiết bị ngoại vi mới
- Phần mềm: Tồn tại một lượng lớn các phần mềm dưới dạng mã nguồn hoặc mã thực thi
- Nền: Hỗ trợ nhiều kiến trúc máy tính: Intel, Alpha, MIPS, Motorola, 64bits system, IBM S/390, SMPs
- Bộ giả lập: Cho phép chạy các ứng dụng của các hệ điều hành khác như MS-DOS, Windows, Macintosh
- Máy ảo: Bộ quản lý máy ảo cho phép chạy nhiều máy ảo với những hệ điều hành khác nhau trên cùng một máy tính thật (máy chủ)
- Hệ điều hành chuẩn: Dùng như hệ điều hành cho những nhà sản xuất phần cứng khác nhau.
- Đa người dùng & Đa tác vụ

- Tương thích: Hơn 95% mã nguồn được viết bằng C, độc lập thiết bị, nên có thể dịch dễ dàng cho nhiều loại máy khác nhau: Máy chủ, máy để bàn, di động,
- POSIX (Portable Operating System Interface for Computer Environments): Cho phép ứng dụng phát triển trên Linux có thể dùng trên các hệ thống khác như UNIX
- Miễn phí, mã nguồn mở & tự do: Tiết kiệm chi phí, không phụ thuộc nhà phát triển ứng dụng

3.4 Làm việc trên một hệ điều hành Linux

- Cần được nhà quản trị máy tính Linux cung cấp một tài khoản biểu hiện bằng một tên đăng nhập (login name/username) và một mật khẩu (password)
- Thực hiện thao tác đăng nhập (login/logon) vào máy tính Linux bằng giao diện đồ họa hoặc dòng lệnh. Người dùng phải khai báo username và password đã cấp trên máy này.

3.5 Các loại tập tin

Tập tin là một khái niệm trừu tượng để chỉ các thiết bị có thể ghi hoặc đọc dữ liệu vào/ra như đĩa cứng, màn hình, con chuột, ...

Có 3 loại tập tin dưới Linux:

- Tập tin bình thường: là các tập tin chương trình hoặc tập tin chứa dữ liệu, văn bản
- Thư mục
- Các tập tin là các thiết bị ngoại vi

3.6 Chuẩn phân cấp hệ thống tập tin (FHS-Filesystem Hierarchy Standard)

Chuẩn phân cấp hệ thống tập tin là một tài liệu mô tả cách sắp xếp các thư mục trên hệ thống Linux. FHS được phát triển để cấp một khuôn mẫu chung nhằm giúp cho việc phát triển các ứng dụng mà không phụ thuộc vào bản phân phối Linux. FHS mô tả các thư mục sau:

- * / : Thư mục gốc
- * /boot: Các tập tin tĩnh cần thiết cho tiến trình khởi động
- * /dev : Các tập tin thiết bị
- * /etc : Các tập tin cấu hình hệ thống và các ứng dụng
- * /lib : Các thư viện chia sẻ và các module của hạt nhân
- * /mnt : Điểm gắn nối các hệ thống tập tin một cách tạm thời
- * /opt : Nơi tích hợp các gói chương trình ứng dụng
- * /sbin: Các tập tin thực thi cần thiết cho hệ thống
- * /tmp : Nơi chứa các tập tin tạm
- * /usr : Hệ phân cấp thứ cấp
- * /var : Dữ liệu biến đổi

3.7 Đường dẫn (path)

Đường dẫn là một chuỗi các tên thư mục ngăn cách nhau bởi ký tự '/', kết thúc đường dẫn có thể là tên một tập tin.

Đường dẫn tuyệt đối: là đường dẫn bắt đầu bằng thư mục gốc '/';

Ví dụ: /home/nbhung/Desktop

Thư mục hiện hành: là một vị trí trên cây thư mục

Ví dụ: /home/nbhung

Đường dẫn tương đối: là đường dẫn được tính bắt đầu từ thư mục hiện hành

Ví dụ: Desktop ; Với thư mục hiện hành là /home/nbhung

3.8 Một số thư mục đặc biệt

- Thư mục gốc ký hiệu /
- Thư mục hiện hành ký hiệu là . (một chấm)
- Thư mục cha ký hiệu .. (hai chấm)
- Thư mục cá nhân (home directory) ký hiệu ~: Mỗi người dùng có một thư mục cá nhân nơi mà người dùng có toàn quyền (thêm, sửa, xóa tập tin thư mục).

Lưu ý: Tên thư mục và tập tin có phân biệt chữ hoa và chữ thường.

3.9 Một số lệnh cơ bản trên thư mục

- Xem thư mục hiện hành: `pwd`
- Xem nội dung thư mục `ls [dir]`
- Chuyển thư mục: `cd newdir`
- Tạo thư mục: `mkdir newdir`
- Sao chép thư mục `cp -r old-dir new-dir`
- Xóa thư mục rỗng: `rmdir a-dir`
- Xóa thư mục: `rm -rf a-dir`

3.10 Một số lệnh thao tác trên tập tin

- Sao chép tập tin `cp old-file new-file`
- Đổi tên tập tin `mv old-name new-name`
- Di chuyển tập tin `mv file-name dir-name`
- Tạo liên kết `ln -s file-name link-name`
- Tạo/Cập nhật tập tin `touch file-name`
- Xóa tập tin `rm [-f] file-name`
- Hiển thị nội dung `cat file-name`

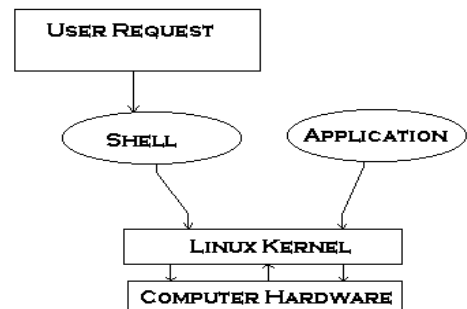
Lưu ý: Chi tiết về các lệnh được trình bày trong 2 tài liệu sau:

- Ngô Bá Hùng - Linux - Các lệnh cơ bản
- Ngô Bá Hùng - Linux - Hệ thống tập tin

Địa chỉ download : <https://sites.google.com/site/nbhung/open-source>

3.11 Bộ thông dịch lệnh

- Là một chương trình chạy ở mức người dùng
- Thông dịch và thực thi các lệnh nhận từ thiết bị nhập chuẩn (bàn phím) hoặc từ tập tin
- Chuyển các lệnh người dùng đến kernel
- Không thuộc kernel



Một số Shell thông dụng dưới Linux:

Tên shell	Người phát triển	Nơi	Ghi chú
BASH (Bourne-Again SHell)	Brian Fox and Chet Ramey	Free Software Foundation	Phổ biến nhất trên Linux
CSH (C SHell)	Bill Joy	University of California (For BSD)	Cú pháp gần ngôn ngữ C
KSH (Korn SHell)	David Korn	AT & T Bell Labs	
TCSH	Ken Greer		Nhiều tính năng hơn CSH

Một số lệnh liên quan đến thông tin về Shell:

- Xem toàn bộ shell của hệ thống: `cat /etc/shells`
- Xem shell đang dùng: `echo $SHELL`

Có hai chế độ sử dụng shell:

- Chế độ tương tác: Thông qua một terminal; Người dùng nhập lệnh từ bàn phím; Shell thực hiện từng lệnh một.
- Chế độ kịch bản (shell script): Một chuỗi lệnh được lưu trong một tập tin văn bản, gọi là một shell script; Yêu cầu shell thực thi tập tin shell script

Những tiện lợi của shell script:

- Có thể nhận đầu vào từ người dùng hoặc tập tin và xuất kết quả ra màn hình
- Là phương tiện để tạo ra các lệnh riêng của người dùng
- Tiết kiệm thời gian vì không phải nhập lại lệnh nhiều lần
- Cho phép tự động các thao tác thường nhật
- Cho phép tự động hóa các tác vụ quản trị hệ thống

3.12 Lập trình shell

3.12.1 Tạo một shell script

- Dùng một trình soạn văn bản để biên soạn shell script
- Gán quyền thực thi cho shell script vừa biên soạn
 - `chmod +x shell-script-name`
 - Hoặc `chmod 755 shell-script-name`
- Thực thi shell script
 - `bash shell-script-name`
 - `./shell-script-name`

Ví dụ:

Dùng trình soạn thảo văn bản lưu nội dung sau vào tập tin có tên là `script-1.sh`

```
#!/bin/bash
# File name: script-1.sh
clear
echo "Hello World !"
```

Cấp quyền thực thi:

```
$chmod 755 script-1.sh
```