The Debian GNU/Linux Glossary for the NewbieDoc Project

Romain Lerallut

Revision History

Revision v.2.0 2 May 2002 Revised by: jee
You know the drill. Edited, added things, removed things, and finally bumped it up to 2.0 to designate a release version and
Revision v0.04 8 April 2001 Revised by: rl
Typos, mostly.
Revision v0.03 23 June 2001 Revised by: rl
added a few stuff, polished a few definitions.
Revision v0.02 30 March 2001 Revised by: jwg
Converted to docbook <glossary> format.
Revision v0.01 29 January 2001 Revised by: RL
First draft of the SGML-based version. Debian Rules!!

This is the Debian-Glossary for Newbiedoc. I try to sum up a few of the words that are usually found in debian, and more generally Linux, but scarcely explained. Thanks to Jesse Goerz for formatting it in Glossary-SGML, and to all the Newbiedocwriters who've contributed.

To share is to grow...

Hope this helps,

Romain Lerallut

A

access permissions

A set of permissions associated with every file or directory, (including "special files"), that determines who can Read, Write or Execute the file. Only the owner of the file or the super-user can modify those permissions.

The format of the access permissions, as shown with **ls -l**, is a list of 10 symbols:

```
drwxr-xr-xdrwxr-xr-x
```

The first symbol is either `-` or `d`, designating whether or not the file in question is actually a directory. After that, there are three sets of three symbols each, telling you whether a file (or directory) is readable, writable, and executable. The three sets represent the owner, group and "everything else" permissions, respectively. For example, the above example represents a file that

- Is a directory
- Gives the owner read, write and executable permissions

- Gives the group read and execute permissions
- Gives everyone else read and execute permissions

See Also: Owner, groups.

ae

Anthony's editor: a very lightweight text editor. It is the one that's included in rescue disks, because of its small size.

AFAIK

As Far As I Know

See Also: AFAIKT.

AFAIKT

As Far As I Know Today

See Also: AFAIK.

ASCII

American Standard Code for Information Interchange. The standard for representing letters, numbers and control characters on 7-bits (8 bits for accentuated character sets).

An ascii text file is a file that can directly be read by basic text editors, such as (cat, more, *joe*, *vi*, *ae*)

autocompletion

A feature offered by most *Shell*s (including but not restricted to **bash**, **tcsh**, **ksh** and especially zsh) that allows the shell to complete a command when it can be done unambiguously. It sometimes offers a list of the possible completions. Check the *Manpage* for your shell for exact syntax and possible uses. With bash and tcsh, completion is achieved by pressing once on the **TAB** key. With ksh, it's a double strike of the **ESC** key...check the man.

B

background

A process is said to be running "in the background" when it is not attached to a terminal. Such programs include daemons, as well as other stuff. You can put a running program into the background in bash with Ctrl+Z. Bash will list the program name and a number you can use to bring the program back to the foreground, with **fg <number>**.

binary, binaries

Binary files are files that are **not** written in *ASCII* (human-readable) format, but in succession of bytes. Binary files include compiled executable files, compressed files (including images), and some data files that don't need/allow easy review.(Or that have special requirements: ciphering, size, etc...)

BTW

By The Way

C

C Language

One of the most powerful computer languages ever invented. Linux is written in C, as well as all UNIX systems: it is thus very portable among UNIX architectures. In fact, C was invented as a langauge which would make writing an operating system easier, and it was used to create UNIX in the first place. A programmer will type C *Source code* source code in a text editor and then compile it into a binary executable form.

See Also: Interpreters.

chown

The **chown** command changes the owner of a file or directory to another user. Usage: **chown** <*`new owner`*> <*`file/directory`*>

Compiler

A compiler translates a *Source code* (ie human readable, *ASCII*) into binary (machine) code, to be executed. GCC is the *GNU* C Compiler, it is issued with most linux distributions.

Curses

What you issue when commands fail. :-)

Seriously, the "curses" thing is a collection of display routines for the old white-on-black terminals. When you're in CONSOLE mode and you edit a file with VI or Emacs, you're using "curses" routines. In the old days (1970's, give or take) computers displayed a CURSOR (hence the curses name, from what I can tell) and the computer guys figured out ways to move the cursor around, up, down, erase, print, reverse and so forth, so other computer guys could develop full-screen text editors (later called WYSIWYG). X windows uses pixel-oriented graphic drawing routines; but the text-based terminal CURSOR-position mechanism is called CURSES.

D

dameon

A program that stays in the background, and provides services for users. Daemons are usually asleep, and wake up only when a task is assigned to them.

See Also: background.

Debian University

It's a "complete training course" about Debian GNU/Linux, where info is stored in courses that depend on each other. A good way to get a panorama of a task, before diving into more technically direct info.

Check it out here: The Debian University (http://home.xnet.com/~darogers/debian_university.txt)

[Remember: straight A's required to get the right to su as root :-)]

Debian

One of the many distributions of Linux. Debian's major difference is that it allows only *Free software*.

For any and all info, check out http://www.debian.org (http://www.debian.org)

There are simultaneously 3 releases of Debian GNU/Linux. One is the "stable", one the accordingly aptly named "unstable", the middle being held by "testing". The stable one is one in which development is stopped and packages are upgraded only for bug fixing purposes. The "unstable" is the development one. In the "testing" release can be found packages that have survived 14 days of hard labor in the unstable area and that have been deemed fit for the service.

Please, do note that "testing" is NOT the same as "stable". Most packages that can be found in testing are OK but there can be a few surprises. A package goes into testing only if it has had no bug reports filed for it while in unstable for two weeks. Being in testing only guarantees that nobody has found any serious problems during standard use.*Use caution,* especially if uptime and stability are important to you (air traffic control, nuclear power plants, and the likes should definitely stay with "stable"...please do :-)

A few Debian releases:

- **Slink:**

  Debian version 2.1

  kernel 2.0

- **Potato:**

  Debian version 2.2

  kernel 2.2

- **Woody:**

  Debian version 3.0

  kernel 2.2 or 2.4 depending on the architecture

- **Sid:**

  Debian version "Still in Development"

  This version is always the "unstable" version.

  The current kernel-du-jour

[little story: The names comes from the "Toy Story" (copyright to its owner and so on) movies. (In the past there were Rex, Ham, Buzz, Bo, and I think the next one may be Sarge, but I'll check). One of the early developers/project leaders of Debian was Bruce Perens, author of the Open Source definition used by the Open Source Movement. At the time, he was employed by Pixar Studios, creator of "Toy Story", and implemented Debian as the development environment for their movies.

[little story: Debian was created in 1993 by Ian Murdock. It was named after him and his wife Debra: Deb-ian (pronounce it deb-ee-ann :-)]

And last, a quote from the wise:

> a "fortune" cookie
>
> Debian Linux is a solid, comprehensive product, and a genuine pleasure to use. It is also great to become involved with the Debian collective, whose friendliness and spirit recalls the early days of the Internet and its sense of openness and global cooperation.

Dependencies

A dependency is a "link" between two packages, mainly when a package is required for another to work or to be installed. (*debconf* is used to configure a number of software. It is written in perl so it depends on package *perl*. Packages depending on *debconf*, also then depend on *perl*, which depends on.....).hen a package is required for another to work or to be installed. (*debconf* is used to configure a number of software. It is written in perl so it depends on package *perl*. Packages depending on *debconf*, also then depend on *perl*, which depends on.....).

Dependencies also refer to conflicting software or versions. You can't easily remove a package if other packages depend on it. No package management programs will allow this to happen without a command to ignore dependencies.

See Also: Packages.

/dev

/dev contains "pseudo files" that are really gateways into peripherals. Most peripherals are "talked to" by one or more /dev/* files. Among them:

- **/dev/fd0** = The first floppy disk drive (MS-DOS floppy A:)

  /dev/fd1 = The second floppy disk drive (MS-DOS floppy B:)

- **/dev/hd[a-d]** = IDE drives...
  - /dev/hda = controller 1 master drive
  - /dev/hdb = controller 1 slave drive
  - /dev/hdc = controller 2 master drive
  - /dev/hdd = controller 2 slave drive

  to get to the partitions, append partition number:

  /dev/hda1 /dev/hdb6 /dev/hdd19

- **/dev/sc[a-...]** = SCSI drives
  - /dev/scd = SCSI device ID 4 (<== is this right?)

  to get to the partitions, append partition number:

  /dev/scd7 /dev/scd13

- depending on your sound driver,these devices are used to send/retrieve sound data:
  - /dev/audio
  - /dev/dsp
  - /dev/mixer
  - /dev/sequencer

- **/dev/tty*** = remote terminal interface

  /dev/tty?? shows up with the ‘w‘ command when you log in via hardwired terminal

- **/dev/pts/*** = pseudo-tty - ssh or (horrors!) telnet interface

  /dev/pts/* lets you interact with a command shell just like a real tty (above)

- **/dev/mouse** = your mouse COM port

Some of those files are symbolic links to others, for example:

/dev/mouse -> /dev/psaux for PS/2 mice

/dev/cdrom -> /dev/hdc if you have an IDE CD-Rom drive

And also many others that I don’t know and that you won’t have to know about unless you do wild things with your linux box ;-).

Distribution, distro, dists

"Linux" does not exist by itself. A distribution is

1. a Linux *Kernel*, and
2. applications (software).

The kind of software, the price of it (or lack thereof), details of the implementation, all depend on the distribution.

Among them: *Debian, Red Hat, SuSe, Slackware, Mandrake* ... and lots of others.

They mostly differ by the tools they offer to manage and or install the system.

E

Emacs

One of the most famous text-editors in the UNIX world. Part of the *GNU* project, there is a text-only version as well as a graphical one, with LOTS of "plugins" available to customize your own version. Emacs has the reputation of being able to do everything, up to but not excluding being the proverbial kitchen sink. Keep in mind that it is a fairly difficult editor to use, especially with all of the plugins installed. Many people stay away from Emacs, using the argument that an editor's job is to let you edit, and programs should do one thing well instead of everything.

See Also: vi*,* joe*,* ae.

e2fs, Ext 2 FS

Second-Extended Filesystem, currently the standard filesystem used in Linux. Future distributions may move toward e3fs, which has some extra features not available in e2fs.

See Also: Filesystem.

F

FAT

The standard name for the family of filesystems that use Microsoft's File Allocation Table format. For example, FAT16 is the old MS-DOS file system, VFAT is the filesystem introduced with Windows 95, allowing filenames longer than 8 characters, and FAT32 is the current filesystem for Windows systems, with some improvements on VFAT. Windows NT uses a different filesystem known as NTFS instead of a FAT system. Of course, all of these filesystems can be accessed from Linux, as long as you use the right tools.

See Also: Filesystem.

Filesystem

a way of storing files on a medium which can be hard drives, floppies, cdroms etc,etc...

ext2 (2nd Extended) is currently the "native" FS for linux

ReiserFS is an *experimental* filesystem, designed for both speed and survivability.

fat (and vfat [CHECK]) are the linux names for the standard Microsoft(r) Windows(r) filesystems: fat16 (fat) and fat32 (vfat).

iso9660 is usually the standard for CD-ROMs.

proc,smb,dev,... are also [special] filesystems.

Linux supports a great number of other file systems, at least in reading but most of them in read/write.

Free software

free is taken in the "freedom" sense.

It is a way of creating and distributing software with the source code and with the right to modify, upgrade and (re-)sell it as one sees fit to do, as long as the license (usually the GNU Public License (GPL)) comes along.

The term *Open Source* is used more and more often, to stop the confusion between freedom and price.

The main difference between the two terms are that when something is designated as *Open Source*, it signifies that the license used for the software has been approved by the Open Source Initiative. (http://www.opensource.org). *check http://www.gnu.org*

FWIW

For What It's Worth

G

GNU

Recursive acronym for **"GNU's Not Unix"**

The GNU (http://www.gnu.org) project was initiated in 1984 by Richard M Stallman, a software developer at MIT's AI Lab. Its goal is to provide high-quality Free Software.

groups

User groups are a feature inherited from UNIX. Groups exist mostly so that resources can be shared between users belonging to the same group. For example, while files for our project are generally owned by the authors, they are part of the newbiedoc group so that any of us can work with them.

See Also: access permissions.

H

Hard link

To the contrary of the symlink (symbolic link), here the actual data on disk is pointed to by more than one directory entry.

A symbolic link is merely a file that refers to another file; a hard link is actually the same exact information, located at the same spot on the disk.

With a hard link, you can pick any one of the instances of the file, and delete all the rest, and it'll still exist; with symbolic links, you can delete the target file and then all the symbolic links to it will point to a non-existent file!

But it's much easier to distinguish between an itty-bitty symbolic link and the target file, than the original doc and any other hard link to it. In fact, it's impossible, since they are the exact same thing!

Hard links are becoming more and more obsolete, in favor of soft links. I strongly suggest you use symlinks, unless you know what you are doing.

(the reference count , ie the number of hard links can be seen on the second column of an ls -l.

If you delete a file with a >1 reference count, it means the data is still stored somewhere, and accessible through another hard link,

if the reference count ==1, and you delete the file, the data will be lost and the space will actually be freed.

Don't mention reference counts <1 they are bad news :-)

HOWTO

Some of the best sources of information available, HOWTOs exist many usual topics and some for unusual ones. (HOWTO make coffee using the parallel port to trigger the coffee machine, for instance). HOWTO files can be found in several places:

- `/usr/share/doc/HOWTO` on your system
- The Linux Documentation Project (http://www.linuxdoc.org/docs.html#howto), and in particular its HOWTO section.
- The Debian FAQ (http://www.debian.org/doc/FAQ/), for debian-specific questions

Always try to check the HOWTOs before asking a question to a newsgroup or mailing list, or else you might get a few s.

For specific info about a command, do a man command:

```
man ls
man csh
man ps
```

or a

```
[your_command] --help
```

for a summary.

Also read our NewbieDoc on the help system (http://newbiedoc.sourceforge.net/tutorials/help-system.en/index-deb-help-sys.html)! :)

HTH

Hope This Helps

I

IIRC

If I (Recall|Remember) Correctly

IMHO

In My (Honest|Humble) Opinion

Image

Computer Graphics: an image is a bidimensional array of colored/gray/BW points. It can be stored in a large numbre of formats (gif,png,jpg,tiff,bmp...) and can be displayed with an even larger number of tools. Web browsers, image viewers (GQview, xv, ImageMagick's display), of image processing tools, such as The Gimp.

Files: an image is an exact copy of a chunk of data. Usually, images are used when creating bootable CDs or floppies, by copying an image of a linux kernel on them. Another use is when burning CDs: you first create a hard disk image of whatever you want to copy (files or raw audio data) and then you copy this image to the CDR.

Interpreters

a program that can executes commands read from a text file.

The shell is an interpreter, so are perl, tcl/tk, python, scheme, and to some extent, Java.

A shell (or perl, or any other) script does not need to be compiled, so it is very portable. However interpreted languages are not as fast as compiled languages such as C, and they depend on the presence of the interpreter.

J

Java

"A highly portable object-oriented language whose syntax is based on C++'s"

Java is currently a very popular language. It is object-oriented and was originally designed by Sun Microsystems to provide developers with a very portable language for use on a multitude of devices.

Traditionally, a Java program is interpreted by a Java Virtual Machine: a software emulation of a standard processing environment (processor, memory, stack, etc.).

The JVM is different on every architecture, but the Java program doesn't need any modification or compilation when switching from one architecture to another. This makes some Java programs exceptionally slow on some occasions, compared to equivalent programs in C++. However, its very high degree of portability make it ideal to expand the client-server software concept.

The latest advances in the development of Java now include direct compilation, making the programs much faster, and expansion in the computer graphics area.

joe

A text editor. (Actually, it's "Joe's Own Editor). **joe** is based primarily on WordStar, a formerly very popular program for MS-DOS.

See Also: vi, Emacs, ae, Recursive.

K

Kernel

The core of a linux system, the kernel is loaded right after LILO starts booting linux, and starts the system according to your configuration.

It is the "conductor" of the linux orchestra.

Versions of the Kernel:

- First (major) number: the "Great Number", that changes once every so often.
- Second (minor)number: the release number.It is even for stable releases and odd for development ones.
- Third number (patch): the update number, usually bug-fixes, or small upgrades.

[Do NOT (as I once did :-) install a kernel with an odd minor number (i.e. 2.3 and hope it will work completely as you expect. Being odd means means this is a development kernel, paving the road for the next even, stable release. (i.e. 2.4).

L

LILO

AKA the LInux LOader

A small program that starts at boot time, before any OS, and that

1. boots linux according to specifications in /etc/lilo.conf
2. allows you to boot other OSes instead of Linux.

check /etc/lilo.conf for configuration

There are other tools in substitution of lilo, such as *grub* but lilo is the most widely used and supported.

Linux

in the truest sense of the word, "Linux" refers only to the kernel.

What most people people call "Linux" is in fact usually "GNU/Linux", at least according to the Free Software Foundation (http://www.gnu.org/gnu/linux-and-gnu.html).

"Linux" comes from UNIX (of course) and Linus Torvalds, who wrote the first free Unix-based kernel. It was used by the GNU project as a kernel for their system while the HURD kernel was|is in development.

[By the way, some people are still fighting over whether to say lie nucks , or lee nux. Make up your mind but choose swiftly , this battle is a bloody mess ;-). Linus Torvalds had to issue a recording (http://www.kernel.org/pub/linux/kernel/SillySounds/english.au) of how HE said it!]

M

Mailing lists

There are many (hundreds at least) of user lists, but *first browse the archives*. Some of those gurus don't like answering the same question fifty times.

They are the closest thing you may find to a "Hot-Line".

I read debian-user (subscribe and unsubscribe at `<debian-user-request@lists.debian.org>` daily, but beware of its high traffic (about 100 mails per day).

Manpage

A specially-formatted file that offers information about a specific command or topic.

```
man ls
```

calls the manpage for the command ls, showing syntax, options,etc.

Most programs install their own manual in the same place (usually /usr/share/man, though that may vary. A variable MANPATH exists and should point to all the directories containing manpages).

For more information on using the **man** command, see the NewbieDoc Help System document (http://newbiedoc.sourceforge.net/tutorials/help-system.en/index-deb-help-sys.html).

Mount

A file system is organized in a tree-form, with subdirectories.In order to access another filesystem (other partitions, disks,etc) one needs to *mount* it. That is, specify the type of the filesystem (ext2,fat...) and the directory where to put the root of the new filesystem in the directory tree.

If you have a dos disk called C in which there are the files **autoexec.bat** and **config.sys**, then the command

```
mount -t msdos /dev/hda1 /mnt/dos
```

will link **C:** to directory **/mnt/dos**. So

```
ls /mnt/dos
```

will give:

```
autoexec.bat
config.sys
```

Linux filesystems are usually automatically mounted during the boot phase,starting with the root filesystem, and then the others: /usr, /home, etc.

*check your **/etc/fstab** for more info*

Be careful when changing **/etc/fstab**, don't mount **/usr/local** before having mounted **/usr**, it does not make sense...

WARNING: The directory called the *mount point* MUST exist and it MUST ABSOLUTELY be EMPTY prior to the mounting operation, or else you will hide files possibly needed for the system's stability.

So do NOT mount ANYTHING over / unless you really know what you are doing.

If you think you feel the need for such a stunt, take a look at the **chroot** command.

O

Open Source

A general term that refers to software that has its source code included so that the consumer can modify the software if required. It is usually used as a synonym for Free Software, but it has less demands on the author than the GNU imposes. The main difference is that having software released as Open Source does not automatically give the consumer the right to redistribute the software, especially with modifications. Always check the license before trying something like that.

P

/proc/*

Pseudo-directory structure acting as gateway to communicate with processes and other 'behind-the-scenes' bowels such as networking, memory, etc.

Those files don't really exist; when read, the system reports info, and some items, when written to, change system behavior. Pretty cool!

- **/proc/$$** = info about current shell (aka $$) including commandline, present directory, environment vars, etc
- **/proc/self/** = symlink to current process info
- **/proc/sys/net/*** = ...
- **/proc/pci:** info about PCI devices
- **/proc/interrupts:** for IRQ monitoring
- ...

Packages

Linux software typically come either in tarballs or in packages.

Most distributions prefer the package form for two reasons:

- It is easier to track what is installed, if it only means checking which packages (and versions).

  (This is very important because, in the Linux world, software changes often)

- It is a very convenient way to check that dependencies are met.

Package management

On Debian systems, use **dpkg** or, even better, **apt-get** to install, monitor, or remove packages.

Do not manage packages "by hand", by directly recompiling sources, because the package tracking system will not tak them into account and your system may stop working.

Debian packages have the extension .deb , and RedHat 's have .rpm. If you want to use a RedHat package under Debian (when the package does not otherwise exist), you might consider using **alien** to convert .rpm into .deb

Partition

A partition is a chunk of hard disk on which a filesystem can be built.

A hard disk can have many partitions , each with its own filesystem. This makes it possible to have two or more operating systems coexist on the same hard disk drive, each one living in its own partition(s).

"Partitioning a hard drive" means separating it into partitions. There are a number of tools to do this, among them the native Linux tools"fdisk" and "cfdisk", as well as the controversial "Partition Magic".

Linux requires at least two partitions: one is the linux swap partition, (usually twice the size of the RAM) and the other is the ext2 root filesystem. However it is sometimes recommended to mount /usr, /home, and other directories on other partitions than the one containing the root FS.

path

The list of directories from the root directory (absolute path) or from any directory (relative path) to a given file: `/usr/bin/vi` is the absolute path of **vi**. When the current directory is `/usr` the relative path of vi is `./bin/vi`.

People also refer to the PATH, which is a list of directories in which commands will run without needing to use the full, absolute path above. For example, if /usr/bin wasn't listed in your PATH, you would need to use the command **/usr/bin/vi** instead of simply **vi**. Your PATH is an environment variable for your shell, and changing it depends on the shell you are using.

See Also: Shell.

PERL

Practical Extraction and Report Language, or Pathological Eclectic Rubbish Lister ; both names are endorsed by its author, Larry Wall.

Sometimes referred to as *"The administrator's swiss army chainsaw"*

Perl is a popular interpreted language, that is used very often to build system administration tools, such as **debconf**, or parts of some servers.

Potato

Debian 2.2

See Also: Debian.

Pre-rolled Software

A euphemism for 'pay-and-pray' algorithms, i.e. commercially-developed, you-dont-get-to-see-the-source-code, hope-you-never-need-support software.

Processes

A process is the way the linux kernel represents programs that are running or are in temporary standby.

Processes are identified by a unique number called PID.They are also characterized by their owner, and their priority.

Check the current processes with **ps** or **ps aux** for all of them.

Proprietary

The approximate opposite of Open Source and Free Software. A data format is proprietary when its specifications are not made public.

Software is said to be proprietary when access to the source is not granted, or only to a selected panel of reviewers. Proprietary software is usually very expensive, compared to Open Source, and is often less tested.

R

Recursive

An acronym in which the first word is the acronym itself. For example:

GNU: Gnu is Not Unix

JOE: Joe's Own Editor

Wine: Wine Is Not an Emulator

and my favorite:

HURD: Hird of Unix-Replacing Daemons

HIRD: Hurd of Interfaces Representing Depth

["We have here, to my knowledge, the first software to be named by a pair of mutually recursive acronyms."], see here (http://www.gnu.org/software/hurd/hurd.html)

Regular expressions

One of the most powerful features of any Unix system is the "regular expression" (also called "regexp"). It is a series of special characters that are used to design patterns, that can be used to build lists of strings (usually but not always filenames). Those lists can then be used by most command line tools. *Also see wildcard...*

A few examples:

```
rm *.hop
```

will remove all files that end by **.hop** ,whereas

```
rm [Hh]op
```

will remove files hop and/or Hop.

Regular expressions are not only very useful, they are also one of the pillars of the Unix way-of-life: command line tools and regexps...

Root

1) Administrator aka "superuser"

The root account is the all-powerful maintenance account. It can read/modify/delete any and all files on the system. It is used when installing soft/hardware, for maintenance tasks, and some processes are run under root UID.

One becomes "root" by typing

```
su
```

at a normal user's shell prompt and then typing the root password, or by directly logging in as 'root'.

> *DO NOT* use the root account when you can use a normal user's. Even if you are the only user, if you don't need to do specifically administration tasks, then do your work as a normal user. Otherwise, you run the risk of deleting everything on your computer. The root account is *very* powerful.

2) *filesystem* aka "/"

The root file system is the first to be mounted. It is the root of the file system tree (*ergo* , the name). Other filesystems are mounted on sub-directories of / (eg /usr, /home and of course /mnt). The kernel needs to be able to mount a root file system (NEEDS ?? CHECK), so its location is specified in lilo.

RTFM

Read The Fine Manual

also sometimes RFD: Read the Fine Doc.

See Also: HOWTO*,* Manpage.

S

SGML

See: Standard Generalized Markup Language

Standard Generalized Markup Language

A powerful markup language for writing documents. HTML is a compact, more specific form of SGML. For information, see Docbook.org (http://www.docbook.org)

See Also: Extensible Markup Language.

Shell

The shell is an interpreter. It can either be used as a command line interpreter for a terminal, or as a shellscript interpreter. A command-line shell is started each time a [pseudo-]terminal is created (using xterm for instance). The type of shell for a certain user can be set by editing /etc/passwd: the last field of the user's line is the shell. Change from /usr/bin/bash to /usr/bin/tcsh, if you like *but make sure the file can be found at that place*, since there is no fall-back.

There are many different kinds of shells and choosing one is a matter of taste. Most Linux distributions offer the bash (Bourne Again SHell) as default but you can see also the csh (C-SHell), tcsh (exTended C-SHell), the ksh (Korn SHell), etc.

Shell programming using scripts is a very powerful tool for an administrator, since it was initially designed for file processing using the many command-line UNIX programs (cat, ls, ps, more, head, tail, awk, sed, etc.)

soft-link

See: Symlink

Source code

a human readable text file. It can either be compiled into an executable program , or it can be interpreted. Being able to read the source code gives *de facto* complete knowledge of the program, in order to understand it, de-bug it, replicate it, etc.

See Also: Proprietary, Open Source.

Symlink

On Windows it's called a shortcut; on Macintosh, it's an alias. A symbolic link is a directory entry that points to some other directory entry.

These are very handy. If you have several versions of a project or file, you can

```
ln  -s  actual-item-being-worked-on  current
```

to be able to refer to "current" instead of the actual item. If the version changes, just update which item your link "current" should point to. Then just keep using "current" so you'll never have to change your typing habits. Very slick!

A symlink is a standalone file, removing it will NOT remove the file it points to.

See Also: Hard link.

T

Tarballs

Many pieces of software for Linux don't come in precompiled form, but rather as a bundle of source code files. These files are appended one to another (with the tar utility) and zipped (compressed) with **gzip**. Their names are often `"hop.tar.gz"` or `"hop.tgz"`

In order to install these programs one needs to unpack them (unzipping and un-tarring) and often to compile them. GNU compliant software is usually easy to compile.

Always read the README file before doing anything other than unpacking!

U

UID, SetUID

A number representating a user. The root user's UID is 0. A program is usually run under the UID of the person who executes it (emacs, gcc, vi) except those that need to do special things (passwd, chsh) that are run as setUID root. These are the programs that are allowed to do things that only root is allowed to do, such as changing a password, or writing things to the screen of everyone on the system. These programs must be kept to a few because of the security risk that they represent.

Owner

The user with individual permissions to a file or directory. By default, the owner is the user who originally created the file, but it can be changed to someone else with the **chown** command.

See Also: access permissions, chown.

V

vi

Will Trillich (Dec. 2000)

"The other horrendous unix text editor. Almost as bad as Emacs (*) but _quite_ different. Variations include Elvis, Vim, NVI and others. Those who use VI tend to look down on those who use Emacs... and vice-versa. You've been warned. Wimps (or people with actual lives) tend to use midnight-commander (mc), anthony's editor (ae) or joe."...

[for whose affirmations I will take no responsibility. I use emacs vi and vim, and I enjoy them all.]

W

Wildcard

A special regular expression, noted ***** , the wildcard is expanded to any and all possible values.

***** means all possible values, **hop*** means all values that start by "hop", and so on...

Window Manager

see X

Woody

Debian 3.0

See Also: Debian.

X

Extensible Markup Language

A widely-used data representation format.

See Also: Standard Generalized Markup Language.

X

aka "The X Window System"

and many other (more or less correct) names (but don't just say 'Windows', argh...)

It is the system that is used to display graphical things on the screen.

An **X server** waits for "events" such as motion from the mouse, keypresses... It can then send this information to X clients such as Xterm, xv, that may run on different computers.

The way an X display looks like is set by a **window manager** whose job is to, ahem, manage your windows and their look. :-) (Borders, background image...)

Choosing a window manager is a matter of taste, and resources. Fvwm or twm have a "classical" look but eat little memory or CPU time, whereas enlightenment does definitely not run on anything other than a Pentium MMX (and THAT, if you have a good video card).

Right now, the two most popular managers are the competing Gnome (http://www.gnome.org) and KDE (http://www.kde.org). Both are available in Debian.

Y

YMMV

Your Mileage May Vary

Z

Zip

Usually, large files (or group of files) are compressed in order to save disk space. The most widely found standard is "zip".(winzip in Windows(r), gzip in Unix), these standards are two versions of an old algorithm called Lempel-Ziv (LZ)(1977).

In Unix, zipped files usually bear the extension .gz (sometimes .tgz see Tar), use **gzip file** to zip (compress) and **gunzip file** to unzip (decompress).