

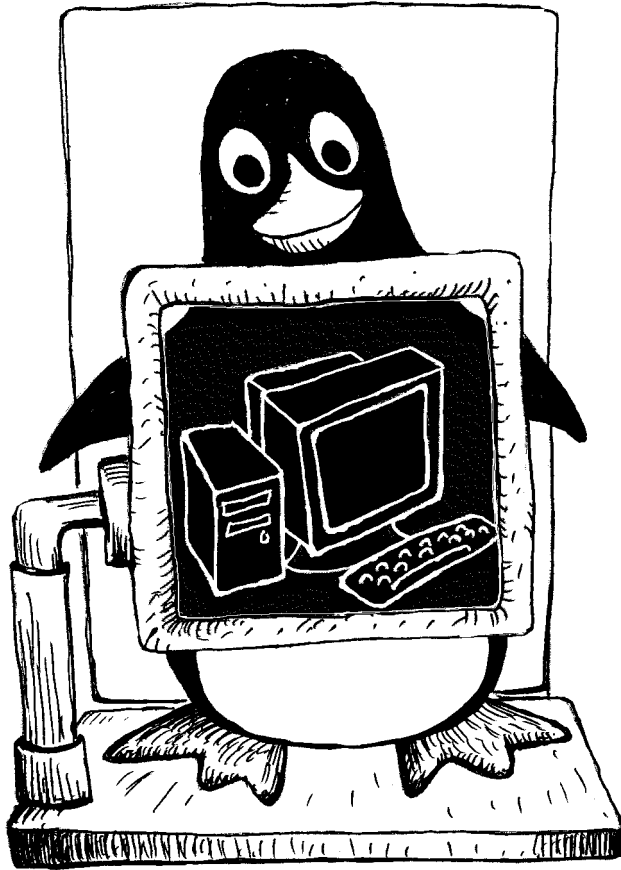
Дмитрий Ляхов

# Linux для начинающих

ИЗДАТЕЛЬСТВО  
**Бестселлер**<sup>®</sup>

Москва  
2003 год

## ЧАСТЬ ВТОРАЯ. Подробно об устройстве Linux



## Глава четвертая. Как выбрать дистрибутив?

Как вам уже известно из глав первой части, Linux — это вовсе не обязательно сама операционная система. Конечно, именем Linux называются все системы, основанные на одноименном ядре, и это название достаточно широко распространено, однако в свою очередь является не совсем верным. Если говорить строго, то словом Linux обозначается только ядро, созданное и поддерживаемое Линусом Торвальдсом. Понятие же операционной системы как таковой намного шире. Операционная система представляет собой, прежде всего совокупность ядра, особой файловой системы, графической или командной оболочки и, наконец, большого числа утилит и программ, поставляемых вместе с системой и позволяющих пользователю с максимальным удобством решать свои задачи. Это касается абсолютно всех операционных систем, не являются исключением и операционные системы, основанные на ядре Linux. Да, именно так, во множественном числе: «операционные системы». Все дело в том, что в мире на сегодняшний день существует огромное количество разновидностей Linux: удачных и не очень, популярных и используемых только узким кругом лиц, развивающихся и «замороженных» на определенной стадии своего развития. И для подобных систем есть свое название: дистрибутив. Именно так называется операционная система, распространяемая на CD/DVD-дисках или доступная для свободной загрузки через Интернет и содержащая все необходимые для работы программы и приложения.

Слово «дистрибутив» произошло от английского «distributive», что можно перевести как «доступный для распространения». Действительно, пользователь, желающий получить работоспособную операционную систему, будет искать именно дистрибутив, а не одно лишь ее ядро. Дистрибутив — это и есть операционная система в традиционном понимании этого слова. Чтобы нагляднее это представить, нужно понять, что диск с дистрибутивом можно установить или удалить (инсталлировать или деинсталлировать). При этом также должен установиться или оказаться удаленным и тот минимум программ, необходимый для комфортной работы.

На самом деле, возникновение большого числа дистрибутивов Linux (для удобства здесь и далее будет использовано тривиальное название этой операционной системы) обусловлено исторически. Поскольку первый вариант ядра Linux был написан именно в соответствии с идеями FSF и распространялся в Интернете по Стандартной общественной лицензии, то неудивительно, что в процесс разработки и совершенствования системы стало включаться все больше и больше программистов из разных стран мира. А это в свою очередь привело к тому, что создание системы оказалось разделено: разработчики объединились в определенные группы, каждая из которых стала создавать свою операционную систему на основе имеющихся разработок. Так и возникли все современные дистрибутивы.

С одной стороны, такое положение дел довольно удобно и обосновано с практической точки зрения, ибо позволяет конечному пользователю подобрать дистрибутив «под себя», заточенный под необходимые задачи и оптимизированный для конкретных приложений и программных пакетов. С другой стороны, выбор как действие, предполагает, что пользователь, перед которым стоит задача выбора дистрибутива, должен принять во внимание как можно большее число параметров и нюансов. А если добавить к этому, что выбор — это еще и поиск компромисса между несколькими решениями, не все из которых можно попробовать в действии, то вывод напрашивается сам собой: выбор операционной системы, а в частности разновидности дистрибутива Linux, — дело довольно сложное. Эта глава призвана помочь разрешить данную проблему.

#### **§4.1. Основные принципы выбора дистрибутива**

Итак, перед пользователем, решившим перейти на Linux или начать знакомство с информационными технологиями именно на основе этой операционной системы, стоит достаточно сложная задача: выбрать дистрибутив. Логично предположить, что этот выбор обусловлен двумя основными факторами: средой применения операционной системы и ее техническими характеристиками.

Как уже было сказано, дистрибутивов Linux существует бесчисленное множество. Впрочем, в определенные перио-

ды времени их всегда было разное количество из-за появления новых и исчезновения старых, оказавшихся либо менее удачными, либо ориентированными на слишком специфическую аудиторию и среду применения. С тех пор, как Linux начала миграцию на домашние и офисные компьютеры, явно выделилась определенная группа дистрибутивов, получивших наибольшее распространение. Но вообще, основных разновидностей дистрибутивов можно выделить три.

- *Популярные.* К ним относятся завоевавшие мировое признание и ставшие брендами дистрибутивы, имена которых на слуху. Это — Red Hat, Mandrake, Debian, SuSE и Slackware.
- *Локализованные.* Это дистрибутивы, команда разработчиков которых ориентируется на одну конкретную страну. Применительно к отечественной действительности, такими дистрибутивами являются ASP Linux и ALT Linux.
- *Прочие.* К прочим дистрибутивам относятся, соответственно, те, которые не завоевали большой популярности в силу разных причин (экономических, технических) или которые не стремились к популярности, зато были ориентированы на определенную группу людей и отточены для выполнения определенных задач (например, дистрибутивы Black Cat Linux, Lunar Linux, Gentoo и другие). Подробнее о некоторых из них будет сказано чуть позднее.

Помимо этого, можно выделить и иные параметры, по которым стоит классифицировать дистрибутивы. Прежде всего, нужно отметить трех «прародителей» всех современных дистрибутивов: Red Hat, Slackware и Debian GNU/Linux. Именно на их основе были созданы иные разновидности операционной системы Linux. Другой критерий сравнения — строение. По этому параметру все дистрибутивы делятся на:

- *пакетные* (Red Hat, Debian, их клоны и т. п.);
- *базирующиеся на исходных текстах* (пример — Slackware);
- *созданные на основе «портов»* (таковым является дистрибутив Gentoo. Сходная технология присутствует и в дистрибутивах Sorcerer, Lunar Linux и некоторых других).

Об особенностях каждой из этих групп будет сказано немало позднее, когда речь пойдет о конкретных дистрибутивах, входящих в них.

Итак, дистрибутивов много, но работать все равно придется только с одним. Как же выбрать? Конечно, по большому счету, Linux есть Linux, и все основные характеристики этой системы, о которых будет сказано в последующих главах, сохраняются в любом дистрибутиве. Тем не менее, сами дистрибутивы могут очень сильно отличаться, поэтому к выбору дистрибутива, тем более, если он первый или выбор необходимо сделать для определенных нужд (домашних, офисных, разработки, создания сети или сервера), нужно подходить осознанно. И в связи с этим можно выделить несколько основных критериев, по которым следует выбрать дистрибутив или хотя бы иметь их в виду. Вот они:

- Самым первым советом по выбору дистрибутива будет тот, что выбирать нужно все-таки операционную систему от разработчика, уже завоевавшего популярность, имеющего систему технической поддержки и работающего с постоянной группой программистов. Несложно догадаться, что под эти параметры попадают дистрибутивы из первой и второй групп. Что касается «маленьких» дистрибутивов, то они, безусловно, могут быть в чем-то очень хороши, однако для широкого применения рекомендовать их сложно. Тем более — для дома или офиса, где важную роль играют стабильность, поддержка оборудования и изначально наличие мощного и гибкого программного обеспечения, обязательно разнопланового.

- Не менее важную роль играет и наличие локализации, в нашем случае — русификации. Русификация бывает двух типов: полная русификация, включающая русификацию интерфейса и поддержку ввода русских символов либо просто возможность ввода русских символов. Если второй тип поддерживают очень многие дистрибутивы Linux, то полную русификацию имеют лишь некоторые: Red Hat, Mandrake, Debian и ASP Linux, равно как и ALT Linux, учитывая то, что последние два дистрибутива создаются непосредственно в России.

- Если для работы критическим является наличие технической поддержки, то необходимо выбрать тот дистрибутив, компания-производитель которого такую поддержку осуществляет. Это как раз те дистрибутивы, которые являются популярными и разработка и распространение которых поставлены на коммерческую основу.
- В том случае, если дистрибутив Linux приобретается впервые и для домашнего использования, не стоит начинать знакомство с Linux ни с маленьких, ни со сложных в установке и использовании дистрибутивов, таких как, например, Slackware. По многочисленным наблюдениям, впечатление об операционной системе складывается в основном из внешних факторов: наличия подходящих программ, удобного и настраиваемого интерфейса, нескольких графических сред, хорошей справочной системы и поддержки большого количества оборудования. Конечно, знакомиться с Linux можно и на основе таких дистрибутивов, но нужно помнить, что эта операционная система, с одной стороны, намного шире и многограннее, а с другой — вовсе не является такой уж сложной в установке и использовании.
- Наконец, важным советом будет то, что при приобретении или получении дистрибутива Linux всегда нужно ориентироваться на последнюю версию и, соответственно, приобретать во владение (не в пользование, поскольку лицензия GPL подразумевает как раз владение) именно ее. Поскольку Linux — развивающаяся операционная система, пользователю вполне естественно находиться на острие прогресса и получать версию системы, в которой исправлены старые ошибки и недоработки и учтены текущие пожелания.

Теперь, когда даны общие критерии выбора дистрибутива, можно перейти и к критериям специальным. Ниже будет подробно рассказано о каждом из популярных дистрибутивов и дан общий обзор дистрибутивов, получивших меньшее распространение, что должно помочь сделать правильный выбор.

## §4.2. Распространенные дистрибутивы

*Red Hat (www.redhat.com)*

Red Hat — дистрибутив, являющийся объективно самым популярным на момент издания книги. Он не был первым созданным дистрибутивом, но, обладая большим количеством преимуществ, он стал лидером по распространенности. Red Hat был создан в 1994 году программистом Марком Эвингом (Marc Ewing). Среди достоинств Red Hat можно отметить, прежде всего, использование получивших огромную популярность RPM-пакетов. Эта технология была создана сотрудниками компании Red Hat, Inc. в 1995 году. Вообще, RPM-пакетом называется совокупность файлов одной программы, которую необходимо установить для того, чтобы программа стала работоспособной и ее можно было запускать и работать с ней. Использование таких пакетов в дистрибутиве значительно облегчает для пользователя процедуру установки программ, лишая его необходимости компилировать программу из исходных текстов. Пакетными дистрибутивами в данном списке являются большинство современных дистрибутивов — Mandrake, SuSE, ASP Linux, ALT Linux и других. Однако это вовсе не значит, что в других дистрибутивах нет возможности использования RPM-пакетов. Напротив, есть, и это делает любой из перечисленных дистрибутивов удобной основой для установки программ без лишних усилий.

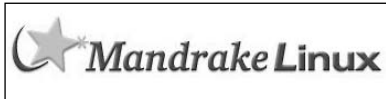
Что касается последнего, то Red Hat уже давно обладает удобным графическим инсталлятором, что делает его установку простой и удобной. Впрочем, это тоже можно сказать почти обо всех современных дистрибутивах. Red Hat известен также и тем, что на его основе наиболее часто создаются web-серверы Apache. Одновременно с этим, «Красная шапочка» (так иногда переводится название дистрибутива) нередко встречается и на компьютерах, выполняющих функции сервера сети.

Каждая новая версия дистрибутива Red Hat выходит приблизительно каждые 4–6 месяцев. Это позволяет включить в дистрибутив новейшие разработки, тем не менее, уже ставшие стабильными версиями, не имеющими ошибок в





программном коде. Компания Red Hat является одной из самых развитых компаний, занимающихся производством дистрибутивов, она даже спонсирует некоторые другие разработки в области открытого программного обеспечения. Последнее, что следует отметить, — это то, что Red Hat выпускается для трех платформ: i386/586/686 (персональных компьютеров на основе процессоров Intel и AMD), Sparc и Alpha. Впрочем, последние две вряд ли будут актуальны для конечного пользователя, не занимающегося профессиональным администрированием разноплановых компьютерных систем.



*Mandrake (www.mandrake.com)*

Разработка дистрибутива Mandrake началась в 1998 году во Франции, где молодой программист Жюль Дюваль (Gael Duval) начал разработку «клона». Название «Mandrake» переводится с английского языка как «мандрагора», однако к оккультизму и астрологии это название имеет слабое отношение. Мандрагора — это всего лишь растение, в честь которого и был назван дистрибутив.

Дистрибутив Mandrake является клоном вышеописанного дистрибутива Red Hat. Что такое клон? Фактически, это тот же самый дистрибутив, но подвергшийся небольшим изменениям. В основном эти изменения касаются программного обеспечения, программы-инсталлятора, а также некоторых внешних признаков. Все программные технологии, составляющие основу исходного дистрибутива, обычно остаются неизменными и переходят к «клону». Впрочем, Mandrake — очень популярный клон Red Hat. Он завоевал известность прежде всего своей невероятно удачной программой установки. Ранее Linux пугал пользователя сложностью установки и постинсталляционной настройки системы. Mandrake же разрушил это предубеждение, создав инсталлятор, позволяющий установить дистрибутив не сложнее, чем операционную систему Windows. Самой важной характеристикой инсталлятора Mandrake является то, что в нем присутствует хорошо развитое средство контроля пакетных зависимостей. Оно необходимо для того, что-

бы при выборе пакетов пользователь «не перестарался» и не получил после установки нерабочую операционную систему. Но это далеко не все: Mandrake обладает еще рядом преимуществ.

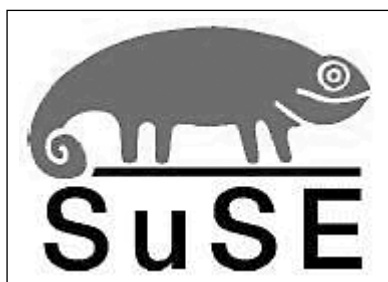
В частности, он всегда содержит сбалансированный набор программ, пригодных к использованию сразу после установки. Mandrake — относительно небольшой дистрибутив. Его дисковые версии обычно могут уместиться на трех или четырех дисках, однако это не сказывается на его функциональности: список программ, традиционно входящих в дистрибутив, позволяет удовлетворить почти все возникающие потребности. Помимо этого, Mandrake обладает несколькими предустановленными степенями защиты компьютера от внешних и внутренних посягательств — пользователь вправе сам выбрать ту степень защиты, которую ему хотелось бы использовать. Это, конечно, не означает, что таких функций нет в других дистрибутивах, просто в Mandrake они реализованы на интуитивно понятном уровне. В дистрибутиве есть приложение, названное Mandrake Control Center (центр управления Mandrake). Подобные центры управления есть в графических средах KDE и Gnome, входящих в состав любого дистрибутива, однако в Mandrake в нем сосредоточено гораздо больше настроек. Например, добавлена возможность настройки оборудования. Таким образом, дистрибутив Mandrake можно легко рекомендовать начинающим пользователям Linux, однако в настоящее время его преимущество перед иными дистрибутивами несколько стирается: почти все из них с каждой новой версией становятся ближе к начинающему пользователю Unix-подобных операционных систем, хотя первый шаг был сделан, конечно, разработчиками Mandrake.

*Debian ([www.debian.org](http://www.debian.org))*

Проект Debian по созданию новой операционной системы с использованием ядра Linux был основан Яном Мердоком (Ian Murdock) в 1993 году при значительной поддержке уже существовавшего в то время проекта GNU Ри-



чарда Столмена. Во многом благодаря этому дистрибутив Debian является прямым наследником проекта GNU, наиболее ярко выразившим всю суть свободного программного обеспечения. Дистрибутив Debian является единственной истинно открытой разработкой: к работе над дистрибутивом может подключиться любой желающий в любое время, в то время как другие дистрибутивы Linux имеют относительно постоянный состав разработчиков. Тем не менее, наиболее важным отличием этого дистрибутива от других является то, что все программы и приложения, которые в нем используются, отлаживаются исключительно тщательно. В Debian никогда не используется программное обеспечение из нестабильной ветви разработки — только самые стабильные версии. Именно это качество позволило Debian GNU/Linux приобрести славу самого надежного дистрибутива, причем подходящего не только для домашних и офисных нужд, но и для организации на его основе локального или web-сервера. Доказательством этого факта является то, что Debian достаточно медленно меняет номера версий своих дистрибутивов. За десятилетнюю историю существования дистрибутива их сменилось всего три, что говорит о надежности каждой из этих версий и серьезном подходе разработчиков к задаче.



*SuSE (www.suse.com)*

SuSE — достаточно удачный пакетный дистрибутив, тем не менее, мало распространенный в нашей стране. Наибольшую популярность он приобрел на своей родине — в Германии, где он был создан в середине девяностых годов. В России же он не популярен потому, что не имеет русификации, хотя его технические характеристики ничуть не хуже, чем у Red Hat или Debian. SuSE не является «исходным» дистрибутивом, таким как Red Hat, однако он не является и его клоном: дистрибутив был создан на основе Slackware, но оказался, в отличие от своего «прародителя», гораздо более ориентированным на домашнего или корпоративного пользователя. SuSE обладает прозрачной и понятной про-

граммой установки. Комплекс настроек, производимых после инсталляции, тоже не должен вызвать сложностей даже у неподготовленного пользователя. Самая яркая особенность дистрибутива SuSE — встроенный комплекс конфигурационных утилит под названием Yast! Yast! — почти идеальное средство для настроек системы, касающихся как настройки производительности, так и оборудования и даже внешнего вида. Словом, SuSE — довольно удачный дистрибутив Linux, однако на момент написания книги не имел приемлемой (следует читать — официальной) русификации, поэтому не совсем актуален в нашей стране. Напоследок нужно отметить одну интересную особенность данного дистрибутива: разработчики SuSE одними из первых начали выпускать версию своего дистрибутива на DVD. Во многом это было обусловлено большим размером операционной системы — профессиональные редакции SuSE занимают десять и больше дисков.

#### *Slackware (www.slackware.com)*

Slackware — самый первый дистрибутив Linux. Его история начинается в июне 1993 года. Именно тогда Патрик Волькердинг (Patrick Volkerding) решает создать первую операционную систему на основе ядра Linux. Slackware, равно как и Debian, полностью придерживается философии свободного программного обеспечения, однако, в отличие от последнего, Slackware завоевал значительно меньшую популярность. Прежде всего, это обуславливалось тем, что программисты хотели создать как можно более Unix-подобную операционную систему, а следовательно, явно не рассчитанную на массового пользователя. Конечно, со временем многое изменилось, но если вам требуется выбрать систему для дома или офиса, Slackware вряд ли будет идеальным выбором. Если же необходима операционная система для организации web-сервера, Slackware вполне может подойти. С официального сайта разработчиков дистрибутива можно загрузить последнюю версию дистрибутива или получить интересующую вас информацию: прочитать отзывы пользователей и FAQ и найти ответы на часто задаваемые вопросы.



### *ASP Linux (www.asplinux.ru)*

Начать рассматривать отечественные дистрибутивы Linux лучше всего с лидера, которым на момент написания книги является ASP Linux. Дистрибутив ASP выпускается в нескольких модификациях: Express Edition, Standard Edition, Deluxe Edition и Server Edition. Последняя — исключительно серверная разработка, не предназначенная для конечного пользователя, тем более, для установки на домашнем компьютере. Что касается трех остальных изданий, то они хорошо подходят для этой цели. Помимо этого, компания ASP выпускает и другие дистрибутивы, во многом способствующие популяризации Linux (один из таких дистрибутивов приложен к настоящей книге).

Наиболее важным достоинством ASP Linux является наличие в этом дистрибутиве хорошей русификации. Русифицировано практически все, начиная от графических сред KDE и Gnome и заканчивая отдельными man-страницами (справочная система, доступная в командном режиме работы Linux).

Другой особенностью дистрибутива является наличие хорошей службы технической поддержки, а также возможности обновления дистрибутива прямо с официального сайта компании. Для этого в дистрибутиве предусмотрена специальная утилита. Что касается установки, то дистрибутив обладает достаточно удобным инсталлятором, позволяющим редактировать структуру разделов жесткого диска, производить выбор пакетов с хорошим контролем зависимостей и выполнять постинсталляционные настройки. Помимо этого, в состав дистрибутива входят три основных загрузчика, позволяющих выбирать операционную систему при загрузке компьютера: стандартные LiLo и Grub, а также собственный загрузчик ASP Loader (его внешний вид можно рассмотреть в главе седьмой).

Дистрибутив ASP Linux полностью совместим с дистрибутивом Red Hat. Это понятно, поскольку ASP является клоном последнего, однако настолько полного взаимодействия удалось добиться только разработчикам ASP. С каждой новой версией этого дистрибутива, в него включается все больше

драйверов оборудования, упрощаются установка и разрешение проблем и конфликтов (конечно, если таковые возникают, что в современных дистрибутивах встречается все реже).

В последнее время компанией ASP Linux разработан дистрибутив, основанный на принципе так называемого *LiveCD*, иными словами — работающий прямо с компакт-диска без установки на винчестер компьютера. Он носит название Greenhorn и позволяет работать с большинством стандартных приложений Linux.

*ALT Linux (www.altlinux.ru)*

ALT Linux — дистрибутив, созданный компанией ALT Linux Team, специализирующейся на открытых разработках. В частности, компанией были осуществлены локализации популярного браузера Mozilla, дистрибутива Debian GNU/Linux, офисного пакета OpenOffice.org, но главная заслуга компании — конечно, создание собственного дистрибутива. Характерной особенностью здесь является то, что ALT Linux Team выпускает много разноплановых дистрибутивов. Это ALT Linux «Утес-К», являющийся максимально защищенной операционной системой, пригодной для использования в качестве сервера, ALT Linux Junior — однодисковый дистрибутив для новичков, ALT Linux Master — более продвинутый вариант, который возможно использовать как в качестве настольной операционной системы, так и сервера сети, web-сервера и рабочего места разработчиков. Впрочем, это касается практически всех «больших» дистрибутивов, таких как Debian GNU/Linux, ASP Linux Deluxe Edition и других.



### §4.3. Прочие дистрибутивы

Несмотря на то, что вышеописанные дистрибутивы Linux являются самыми распространенными, удобными и удачными, количество вариаций Linux этим небольшим списком не исчерпывается. Существует еще немало удачных (или просто оригинальных) дистрибутивов, тем не менее, не получивших широкого распространения. С некоторыми

из них тоже необходимо познакомить читателя.

*Gentoo (www.gentoo.org)*

Gentoo является достаточно необычным дистрибутивом Linux. Это относительно небольшой и быстрый дистрибутив, однако главная его особенность состоит в том, что он не является пакетным. Вернее, пакеты им используются, однако это не RPM-пакеты, на которых основан дистрибутив Red Hat и его многочисленные клоны. Для инсталляции программ в Gentoo используются «порты» особая система установки программного обеспечения, которая в оригинале называется porteges. Ее суть заключается в следующем: после компиляции программы для Gentoo, «порт» сохраняется в каталоге /usr/portege. Для того, чтобы установить программу, необходимо найти ее «порт» в этом каталоге и дать команду emerge (слияние), после чего программа оказывается установленной и готовой к работе. В остальном же дистрибутив Gentoo сходен с прочими операционными системами на основе ядра Linux.

*Stampede (www.stampede.org)*

Дистрибутив Stampede был создан в 1997 году исключительно как операционная система для компьютеров, использующих процессоры Intel. В результате этого, все приложения, входящие в состав дистрибутива, были оптимизированы именно для этой архитектуры. Stampede — пакетный дистрибутив, но он использует собственную систему пакетов, называемую SLP (Stampede Linux Packages), заявленные производительность и сила сжатия данных которой превышают аналогичные параметры RPM-packages.

*Lunar Linux (www.lunar-linux.org)*

Lunar Linux — тоже необычный дистрибутив. Конечно, его функциональность приблизительно равна функциональности всех «маленьких» дистрибутивов, однако Lunar Linux имеет одну интересную особенность: так же, как и

---

<sup>1</sup> На самом деле, подобный принцип был использован даже раньше — в дистрибутиве Sorcerer, на основе которого и был создан Lunar Linux.

Gentoo, он использует систему «портов» для установки программного обеспечения, но в Lunar Linux эта процедура была реализована гораздо проще. Для этого в дистрибутиве были изменены некоторые команды, при запуске которых с определенным параметром, устанавливается необходимая программа<sup>1</sup>. Команды Lunar Linux вполне соответствуют антуражу системы: чего стоят, например, команды `lunar`, `moonbase` или `module`.

#### *Knoppix (www.knoppix.ru)*

Knoppix — первый в мире дистрибутив Linux, относящийся к классу LiveCD-дистрибутивов. Это означает, что он может работать напрямую с компакт-диска и не требует инсталляции на жесткий диск компьютера. Разработанный командой программистов, возглавляемой Клаусом Кноппером (Claus Knopper), дистрибутив быстро приобрел популярность прежде всего среди тех, кто ранее не использовал Linux, но пожелал опробовать свободную ОС. Дистрибутив предназначается не только для образовательных целей или демонстрации возможностей Linux, но и для создания постоянной рабочей среды, которую можно использовать на любом компьютере — достаточно вставить компакт-диск в привод. Основная сфера применения — работа с пользовательскими приложениями: офисным пакетом OpenOffice.org, программами для работы в Интернет, аудио и видео плеерами и играми. Knoppix частично русифицирован, поэтому проблем с использованием русских символов не возникает.

Вот, пожалуй, и все, что необходимо знать пользователю о многообразии дистрибутивов Linux. При желании, интересующую информацию всегда можно найти в сети Интернет. Тем же, кто ищет в книге конкретного совета, какой дистрибутив выбрать для личного пользования, можно порекомендовать практически любой из семи популярных, за исключением разве что Slackware (ориентированность не на начинающего пользователя) и SuSE (отсутствие русификации).

После прочтения этой главы наверняка возникнет во-



прос: на основе какого дистрибутива написана данная книга? Тем не менее, однозначного ответа нет и быть не может. В процессе создания книги были использованы разные дистрибутивы, в большей степени Red Hat 8.0, Mandrake 9.0 и ASP Linux 7.3 Deluxe Edition. Как уже было сказано, разница между дистрибутивами заключается в основном в наличии или отсутствии того или иного программного обеспечения. В книге говорится об очень многих программах для Linux, но большинство из них входят в любой стандартный дистрибутив, для других же даны ссылки на дистрибутив, в котором их можно найти. Что касается программы-инсталлятора, различающейся от одной модификации Linux к другой, то в главе об установке Linux описаны самые яркие из них. Это доказывает, что современные развитые дистрибутивы Linux имеют практически все необходимое для того, чтобы считаться операционной системой многопланового применения, пригодной для домашнего и офисного использования. А о различиях между дистрибутивами и критериях выбора было много сказано выше.

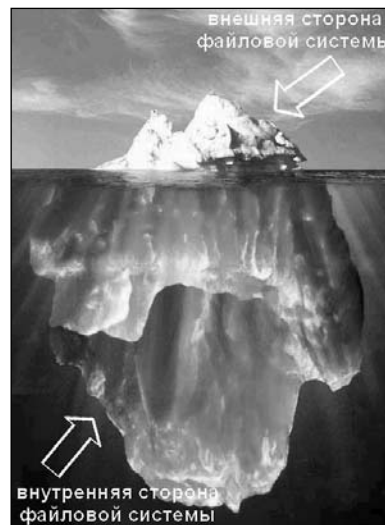
## Глава пятая. Файловая система Linux

### §5.1. Файловая система: что это такое?

Как известно, любой компьютер, для каких бы целей он бы ни был предназначен, является не только машиной для произведения вычислений, но и хранилищем информации самого разного рода — от документов до конфигурационных файлов операционной системы, на нем установленной. Что касается информации, то хранить ее можно по-разному, в частности в оперативной или постоянной памяти. Ко второму случаю относится хранение информации на жестком диске. Особенности этого способа и будут рассмотрены ниже.

Как можно догадаться, для того чтобы хранить на жестком диске разнообразную информацию и осуществлять к ней доступ как со стороны пользователя, так и со стороны системы, необходимо ее структурировать. И именно такая структура, возможно, даже иерархия, данных носит название файловой системы. Тем не менее, файловая система понимается по-разному, точнее, имеет две стороны — внешнюю и внутреннюю. Дабы нагляднее себе их представить, взгляните на *рис. 5.1*. Действительно, любую файловую систему можно изобразить в виде айсберга, имеющего свою надводную и подводную части. Причем надводной частью является как раз то, что видит человек, севший за компьютер, — файлы, папки и прочие иерархические атрибуты. Нижняя же часть — это то, что видит операционная система. Она представлена кластерами и иными единицами распределения информации, понятными операционной системе. Впрочем, о внутреннем устройстве файловой системы и видах файловых систем, поддерживаемых Linux, будет сказано чуть позже. А пока рассмотрим вершину айсберга, то есть ту сторону файловой системы, которая обращена к пользователю.

**Рис. 5.1.**  
Наглядное изображение файловой системы.



Любая операционная система, существующая на сегодняшний день либо даже существовавшая ранее, имеет свою структуру размещения файлов, однако во всех этих структурах можно выделить как общие, так и особенные черты. Общим является то, что все свободное пространство жесткого диска понимается как корневой каталог — самое большое по объему «помещение» для хранения информации. В нем можно размещать как файлы, так и другие каталоги. Разумеется, никто не запретит вам создать в одном каталоге ровно столько подкаталогов, сколько потребуется. Такая организация называется деревом каталогов. Впрочем, если вы работали в Windows, равно как и в DOS, вы наверняка знали все вышесказанное раньше, поэтому мы переходим к рассмотрению того особенного, что отличает устройство файловой системы Linux от устройства других файловых систем.

### §5.2. Особенности файловой системы Linux

Итак, в Linux точно так же, как и во всех иных операционных системах, существует корневой каталог, который обозначается как «/» (косая черта или правильный слэш) и не носит ни какого другого названия, будь то имя каталога или буква диска, как это принято в Windows. Кстати, необходимо отметить, что слэш, используемый в Linux, — именно «правильный», то есть имеет наклон слева направо, и именно такой, какой используется при написании адреса веб-страницы в строке браузера, а не обратный (back slash), какой используется в среде DOS.

Помимо этого, в Linux, в отличие от Windows, нет привычного для многих пользователей отображения физических устройств, в частности тех, на которых может храниться информация, — жесткого диска, флоппи-дисковода, привода CD-ROM или DVD. В Linux существуют только файлы устройств, которые располагаются в своем каталоге.

Операционная система Linux отличается и от DOS, если уж сравнивать операционные системы, умеющие работать в режиме командной строки, в частности тем, что в первой несколько смазывается роль расширения файла. Конечно, оно присутствует, и по нему можно определить, какой род ин-

формации заключен в данном файле, однако для самой операционной системы расширение не имеет особого значения. Нет в Linux и исполняемых файлов, в том виде, в котором они присутствуют в DOS (обычно они имеют расширение \*.exe). По большому счету, любой файл в Linux является исполняемым, если быть более точным, тот, в атрибутах которого указано право на его исполнение. Вообще, об атрибутах и свойствах файлов необходимо поговорить особо, поскольку зачастую эти параметры требуют к себе обращения.

### §5.3. Свойства файлов Linux

Как вы уже могли понять, читая эту главу, файловая организация в Linux имеет значительные отличия по сравнению с файловой организацией других операционных систем. Но различия эти касаются не только особенностей размещения файлов и каталогов на диске. Касаются они, возможно даже в большей степени, еще и свойств файлов. Эти свойства называются в Linux атрибутами файла.

Вообще, атрибуты всех файлов, какие только можно встретить в Linux, условно делятся на две группы: *атрибуты принадлежности* файла и *атрибуты прав доступа* к нему.

Атрибуты принадлежности — это атрибуты, указывающие, кому может принадлежать данный файл или каталог, поскольку каталог также является единицей файловой системы. В связи с тем, что Linux — система многопользовательская, то и определяется принадлежность файла по-разному. Во-первых, файл может принадлежать только одному человеку — его владельцу, создавшему файл либо скопировавшему его из внешнего источника. В данном случае файлу назначается атрибут «owner». Во-вторых, файл может принадлежать определенной группе пользователей, в частности той, к которой принадлежит владелец файла. При этом в его атрибуты добавляется слово «group». И, наконец, в-третьих, файл может быть доступен для всех. И если это так, то в его свойствах можно найти значение «other».

Что касается прав доступа к файлу, то их тоже существует три основные разновидности: *право на чтение (просмотр)*, *право на изменение (редактирование)* и *право на исполнение (запуск) файла*.

- *Право на чтение (read)* выражается в том, что пользователь системы вправе просмотреть файл, используя команды просмотра либо любое предназначенное для подобных целей приложение, и скопировать его без изменений.
- *Право на изменение (write)* — это возможность изменить содержание файла, тем не менее, не удаляя его, поскольку это будет невозможно<sup>1</sup>.
- *Право на исполнение (execute)* распространяется, как можно догадаться, только на исполняемые файлы. Именно это право составляет основное отличие исполняемого файла Linux от подобного файла в другой операционной системе. Расширение в Linux теряет свою роль, а возможность исполнения данного файла прописывается только в его атрибутах.

Итак, это основные атрибуты файлов в Linux. Просмотреть их можно определенным комплексом команд, описанным в главе шестой, в которую при желании можно заглянуть уже сейчас. Но для начала предстоит рассмотреть, какие конкретно файлы могут существовать в Linux и какими правами они обладают.

#### §5.4. Типы файлов Linux

Прежде чем говорить о типах файлов в Linux, нужно хорошо представить себе те условия, которые на них накладываются в этой операционной системе. Прежде всего, нужно помнить о том, что имя любого файла Linux может иметь длину до 255 символов и состоять (счастье для любителей создавать экстравагантные имена файлов) из любых символов, кроме простого пробела и уже известного нам правильного слэша, которым отделяются каталоги. Но, как известно, настоящие компьютерщики пробелами не пользуются, поэтому с таким положением дел вполне можно мириться, тем более что создать файл или каталог даже с любым из этих двух символов в его имени вам никто не помешает, сто-

---

<sup>1</sup> Удалять файлы в Linux имеет право только системный администратор. Разумеется, это не касается тех файлов, которые находятся на логических дисках остальных пользователей.

ит лишь применить одну хитрость. Для этого достаточно заключить подобное имя в кавычки в процессе создания файла. Есть и другой вариант — поставить перед запрещенным символом обратный слэш «\».

Помимо выше сказанного, полезно знать и о том, что имя файла в Linux является всего лишь ссылкой на сам файл, действительно содержащий информацию, и ссылок таких может быть сколько угодно. Из этого нетрудно сделать вывод, что один и тот же файл может называться по-разному. И это позволяет упорядочивать структуру каталогов, благодаря различию процедур удаления самого файла и ссылки на него. Если проводить аналогию с Windows, то аналогом ссылки на файл, хотя и очень условным, является его ярлык, который можно размещать в нескольких каталогах сразу.

Перейдем к типам файлов. Основных типов файлов, характерных для файловой системы Linux, три:

- *обычные файлы;*
- *символические ссылки;*
- *файлы физических устройств.*

Конечно, есть и другие, но сфера их применения достаточно узка и лежит в основном в плоскости профессионального системного администрирования, поэтому их рассматривать мы не будем.

Итак, с обычными файлами все понятно — это те файлы, в которых действительно содержатся данные. Многие считают, что поскольку Linux имеет свои собственные типы файлов для каждого вида информации (текстовой, графической, звуковой, видео), то она не синхронизируема с Windows. Однако это далеко не так. Linux поддерживает многие файлы с характерными для Windows расширениями: \*.bmp, \*.jpg, \*.html, \*.pdf, \*.txt, \*.doc, \*.rtf, \*.wav, \*.mp3 и многими другими, поэтому проблемы передачи данных между этими системами не существует или почти не существует.

Что касается символических ссылок, то это — имена файлов, которых, как уже говорилось, может быть сколько угодно.

но много. Сами ссылки можно использовать в целях упорядочивания иерархии файлов и поддержании в ней образцовой системности, когда администратору системы и простым пользователям точно

известно, где найти тот или иной файл, поскольку они сгруппированы, а символические ссылки на него облегчают поиск. Так же с помощью символических ссылок указываются файлы, находящиеся на съемном носителе либо на логическом разделе жесткого диска, отформатированного в другой файловой системе, отличной от той, которая используется Linux.

Файлы физических устройств — это те файлы, которые соответствуют устройствам, подключаемым к компьютеру (принтер, сканер, модем), синхронизирующимся с ним (КПК, Mp3-плеер) или входящим в его основной состав (жесткий диск, звуковая и видео подсистемы). Все эти файлы находятся в каталоге /dev (от англ. Device — устройство), и их легко можно там найти. А чтобы понять, что есть что, см. таблицу 5.1.

### §5.5. Каталоги Linux

Рассмотрев файлы Linux, переходим к рассмотрению каталогов. Тем, кто имеет

**Таблица 5.1.**  
**Файлы устройств**

<i>/dev/console</i> —	<i>системная консоль, а точнее монитор и клавиатура, являющие собой основную систему ввода и отображения информации, понимаемую Linux.</i>
<i>/dev/bd</i> —	<i>жесткий диск с интерфейсом IDE. Следует отметить, что в этом каталоге скорее всего находятся и другие файлы, например, <i>bda</i>, <i>bdb</i>, <i>bdc</i> и <i>bdd</i>, где <i>a</i> — <i>primary master</i> (устройство-хозяин на первичном носителе), <i>b</i> — <i>primary slave</i> (устройство-раб на первичном носителе), <i>c</i> — <i>secondary master</i> (устройство-хозяин на вторичном носителе), <i>d</i> — <i>secondary slave</i> (устройство-раб на вторичном носителе). Скажем, <i>bda</i> может подразделяться и на <i>bda1</i>, <i>bda2</i>, <i>bda3</i>..., где 1, 2, 3 — порядковые номера логических дисков на одном физическом, разумеется, если вы таковые используете.</i>
<i>/dev/fd</i> —	<i>флоппи-дискетод.</i>
<i>/dev/tty</i> —	<i>последовательные порты, например, COM1, COM2, COM3.</i>
<i>/dev/null</i> —	<i>это не совсем устройство, скорее, это аналог «корзины» Windows, с той разницей, что любая информация, записанная на это устройство, исчезает навсегда.</i>

опыт работы в Windows, известно, что все файлы, используемые этой операционной системой, расположены либо прямо на диске, либо в различных папках. Причем любые программы и приложения там можно размещать и устанавливать с большой степенью свободы. В Linux же существует гораздо более четкая структура каталогов. При установке Linux изначально создается больше каталогов, чем в какой бы то ни было другой операционной системе. Каждый из них имеет свои определенные функции. Это создает интересное ощущение присутствия внутри системы и возможности повлиять на ход протекающих в ней процессов.

Чтобы получить более или менее четкое представление о том, что хранится в каждом из основных каталогов, рассмотрите *таблицу 5.2*.

### §5.6. Виды файловых систем, используемых Linux

Настала пора поговорить о внутренней стороне файловой системы Linux. Вообще, когда речь идет о файловой системе, используемой Linux, обычно вспоминается ext2fs, как самая распростра-

**Таблица 5.2.**  
**Основные каталоги Linux**

/	<i>Корневой каталог</i>
/root	<i>Каталог администратора системы, называемого также суперпользователем. Доступ других пользователей к этому каталогу обязательно защищен паролем.</i>
/home	<i>Каталог, содержащий подкаталоги всех пользователей системы.</i>
/boot	<i>Каталог, содержащий файлы, необходимые для загрузки системы, а также ядро Linux.</i>
/etc	<i>Здесь находятся основные конфигурационные файлы, а также некоторые из тех файлов, которые требуются на начальной стадии загрузки системы.</i>
/lib	<i>Этот каталог содержит так называемые библиотеки, необходимые компилятору C (C++). Они используются многими программами, что позволяет ускорить их работу.</i>
/mnt	<i>Каталог, содержащий все смонтированные в Linux устройства. Разумеется, это касается только тех устройств, которые имеют свою файловую систему и позволяют хранить информацию.</i>
/tmp	<i>Папка для временных файлов, создаваемых системой. Этот каталог самоочищается.</i>
/lost+found	<i>Каталог, где складываются потерянные в результате нештатной ситуации файлы. Сюда обращается программа восстановления системы.</i>



<i>/dev</i>	<i>Каталог файлов устройств. Подробнее о том, что в нем находится, можно узнать из таблицы 5.1.</i>
<i>/usr</i>	<i>Это самый большой каталог. В нем располагается львиная доля всех установленных в системе программ и приложений.</i>
<i>/bin</i>	<i>В этом каталоге содержится информация обо всех командах Linux.</i>

ненная и наиболее используемая файловая система. Однако список файловых систем, с которыми может работать Linux, гораздо шире. В этом и заключается одно из главных преимуществ Linux — поддержка большого количества файловых систем, а, следовательно, и гибкость использования операционной системы в целом.

Ниже будет дана достаточно подробная классификация файловых систем, которые может использовать Linux, однако прежде, чем перейти к их описанию, необходимо уточнить два основных момента.

Первый из них заключается в том, что нужно обязательно различать два основных вида файловых систем. К первому относятся те файловые системы, которые имеют жесткий диск или его раздел, в котором непосредственно установлена операционная система Linux. Второй вид — это файловые системы, доступ к которым может осуществляться под управлением Linux, однако на которые установить данную операционную систему нельзя. Такие системы обозначены в таблице как поддерживаемые.

И другой момент: файловые системы для Linux подразделяются на журналируемые и не журналируемые (journalized и non-journalized). Что же такое журналируемая файловая система? Проще всего представить суть этого явления можно, сравнив ее работу с работой не журналируемой файловой системы.

Когда операционная система Linux завершает свою работу, то все данные и все изменения, произошедшие в ней за сеанс работы, сохраняются на жесткий диск. Однако такое происходит только при корректном завершении работы компьютера или выходе из системы. Если же система отключилась в результате сбоя (фатальной системной ошибки или банального перебоя в сети электропитания), то все данные сохраниться не успевают. Более того, Linux «забы-

вает» и местоположение большей части файлов, находящихся на жестком диске. В результате, при последующем включении компьютера, программе *fsck* (служебной утилите Linux) приходится методично просматривать весь жесткий диск, дабы восстановить информацию о нем. В журналируемых файловых системах эта проблема решена. Все изменения, происходящие во время работы с Linux, в этих системах записываются в некоторое подобие журнала (отсюда и название). Поэтому, в результате внезапного сбоя, важные данные с жесткого диска хоть и теряются, но их копии остаются в «журнале», что позволяет быстро вернуть операционную систему в работоспособное состояние, не прибегая к долгой проверке диска при помощи утилиты *fsck*. К журналируемым файловым системам можно отнести *ext3fs*, *ReiserFS*, *JFS* и некоторые другие их разновидности. Познакомиться с самыми популярными файловыми системами для Linux можно, прочитав врезку.

Последнее, что нужно отметить касательно файло-

## Виды файловых систем Linux

### Основные

#### Нежурналируемые

**minix** — файловая система одноименной операционной системы, созданной исследователями из Хельсинского университета. Именно на ее основе Линус Торвальдс и создал Linux. И, разумеется, *minix* стала первой файловой системой, поддерживаемой Linux. Сейчас *minix* почти не используется.

**extfs** — первая версия расширенной файловой системы, обладающей большими возможностями, по сравнению с *minix*. В отличие от последней, *extfs* поддерживает, например, жесткие диски размером больше 64 мегабайт, а также более длинные имена файлов. Тем не менее, *extfs* тоже безнадежно устарела.

**ext2fs** — вторая версия расширенной файловой системы оказалась более удачной. В частности, она гораздо лучше оптимизирована, чем *extfs* и обеспечивает более высокую производительность. Не удивительно, что она стала практически основной файловой системой для Linux, всех ее дистрибутивов и разновидностей.

**swap** — раздел подкачки Linux имеет свою файловую систему. Она называется *Linux swap* и организована так, чтобы обеспечить наиболее быстрый доступ к данным.

**proc** — вспомогательная файловая система. Целиком создать дисковый раздел *proc* нельзя, более того, файлы этой файловой системы не занимают места на диске, *proc* служит в основном для обмена данными между ядром операционной системы и ее составными частями.

**nfs** — расшифровывается как «Network File System». Это сетевая файловая система, позволяющая обмениваться данными, находящимися на жестких дисках удаленных компьютеров.

*Журналируемые*

**Ext3fs** — улучшенная версия популярной файловой системы *ext2fs*. Фактически, *ext3fs* и *ext2fs* аналогичны по своей структуре. Изменения коснулись лишь возможности ведения «журнала».

**JFS** — журналируемая файловая система, созданная специалистами из IBM. Характеризуется достаточно высокой надежностью и поддержкой файлов больших размеров.

**ReiserFS** — данная система имеет одну уникальную особенность по сравнению с другими файловыми системами Linux. Это более сложная структура данных, именуемая *B+Trees* или *Balanced Trees*. Не вдаваясь в особенности ее функционирования, можно сказать, что она позволяет осуществлять быстрый поиск данных на диске, благодаря чему повышается общая производительность системы. Помимо этого, *ReiserFS* поддерживает самый большой размер файла — 210 петабайт, правда, только начиная с третьей ее версии.

## Поддерживаемые

**FAT** — знакомая всем файловая система Windows линейки 9x (*Windows 95/98/Me*). Поскольку она поддерживается Linux, то и данные, на ней хранящиеся, вполне доступны в Linux. Разумеется, если система монтирована.

**NTFS** — файловая система Windows серии NT и XP. Доступ к ней также возможен из Linux.

**MSDOS** — файловая система DOS тоже доступна, правда, со всеми сопутствующими этой файловой системе атрибутами — короткими именами файлов, ограниченным разделом жесткого диска и т. п.

**HPFS** — файловая система уже устаревшей OS/2. Тем не менее, она тоже поддерживается Linux.

вых систем Linux, — это то, что набор поддерживаемых файловых систем практически одинаков для любого дистрибутива, однако в разных дистрибутивах включить поддержку иных систем (например, не указанных в списке, предлагаемом в процессе установки системы) бывает непросто для начинающего пользователя. Поэтому основной рекомендацией будет использовать при установке Linux стабильные и проверенные файловые системы, такие как *ext2fs*, *ext3fs* или *ReiserFS*. Они точно поддерживаются.

## Глава шестая. Команды Linux

Мы сказали что, Linux — это операционная система, осуществляющая взаимодействие с пользователем при помощи командной строки. Тем не менее, читателю, наверное, до сих пор многое в этой фразе остается непонятным. Данная глава призвана помочь определиться, для чего, собственно, нужна командная строка Linux и что с ее помощью можно делать. А самый главный вопрос, на который предстоит ответить: как, каким образом осуществляется подобное взаимодействие с пользователем и какую пользу можно из него извлечь?

### §6.1. Что такое команда?

Операционная система Linux в начале своего исторического пути была объективно недостаточно удобна для неподготовленного пользователя. Неудобство это заключалось прежде всего в том, что пользователю, дабы управлять системой, необходимо было изначально знать большое число команд. После установки он не попадал в графический режим X Window или графическую среду KDE, как это происходит после стандартной установки современных дистрибутивов Linux, а оказывался перед черным экраном монитора, на котором высвечивалось только приглашение к работе в командной строке. Тем не менее, уметь работать с командной строкой необходимо и в настоящее время — в век графичес-

#### Когда использовать командную строку?

*Использование командной строки может быть целесообразным в следующих случаях:*

- Система не загружается в графическом режиме из-за неполадок.
- С системой можно работать в графическом режиме, но в ней обнаружены неполадки. С целью их устранения можно загрузиться в относительно беспробойном режиме командной строки и попытаться их устранить.
- Если компьютер имеет слабую конфигурацию, использование командной строки может оказаться единственным возможным вариантом работы с ним.
- Иногда быстрее запустить программу через терминал, чем искать ее в многочисленных пунктах меню.
- Командная строка зачастую оказывается более гибким инструментом. Наличие огромного количества параметров к командам — прямое тому подтверждение.

ких интерфейсов. Во-первых, это может оказаться единственным выходом из сложившейся в операционной системе внештатной ситуации, когда устранение неполадки в графическом режиме просто невозможно, а во-вторых, командная строка Linux имеет немало преимуществ. Основные варианты, когда целесообразно использовать командную строку, представлены во врезке.

Итак, что же такое команда? Говоря упрощенно, командная строка операционной системы — это набор символов, при помощи которых можно выполнить любое действие: запустить необходимую программу или даже продуктивно работать с ней посредством исключительно командной строки. Логично предположить, что этот набор символов, составляющих команду, должен быть понятен как пользователю, так и операционной системе. В конце концов, команды — это не язык программирования, а упрощенные письменные конструкции. Конструкции эти пришли, естественно, из английского языка, поэтому при определенных языковых познаниях можно легко догадаться, для чего предназначено подавляющее большинство этих команд. Например, команда *find* служит для поиска определенных файлов, *mount* — для монтирования (подключения) файловой системы и т. д.

Тем не менее, поскольку команды — это не порядок знаков в двоичной системе счисления, то собственно ядро операционной системы их понять, а, следовательно, и правильно интерпретировать не может. Поэтому на помощь приходят специальные программы-посредники. Они называются командными оболочками.

### §6.2. Что такое командная оболочка?

Командные оболочки Linux, называемые также командными интерпретаторами или по-английски, shell, это программы, задача которых заключается в обработке вводимых пользователем команд с тем, чтобы их правильно поняла операционная система. Командная оболочка — это приложение, работающее исключительно с текстом. В его окне (или полноэкранном режиме, если загрузка системы происходит без использования графического режима

X Window) вводятся команды. Там же отображается текстовая информация, выдаваемая интерпретатором команд. Если вы уже имеете опыт работы с операционной системой DOS или хотя бы запускали сеанс MS-DOS из-под Windows, то вполне представляете себе, как должно происходить подобное общение с пользователем. Однако нужно заменить, что командные оболочки Linux по своему устройству гораздо сложнее упомянутого сеанса MS-DOS. Их достоинство заключается в том, что они не только интерпретируют классические команды Linux, но и работают со специальными небольшими программами — *скриптами*. Скрипты значительно расширяют функциональность классической командной строки Linux (правильнее было бы сказать Unix), однако для работы с ними требуются хотя бы минимальные познания в программировании.

Командных оболочек для Linux существует много. Они различаются по функциональности, наличию возможности интерпретирования различных языков программирования и прочим параметрам. Самой популярной из них заслуженно является оболочка BASH. Расшифровывается ее название так же оригинально, как и все названия, встречающиеся в Linux — Bourne Again Shell. О том, почему для нее было выбрано именно такое название, можно прочитать во врезке.

BASH является командным интерпретатором Linux по умолчанию. Если вы ничего не изменяли в глубоких настройках системы, то он один загружается вместе с Linux. Эта оболочка является достаточно удобной и продвинутой, поэтому работа всех команд Linux будет показана именно на примере их интерпретации BASH.

### Почему BASH?

*Как известно, прародительницей Linux является Unix. Что касается командных интерпретаторов в Unix, то изначально все они назывались Shell, то есть оболочкой. Некоторое время спустя, были разработаны новые, модифицированные версии Shell. Одной из наиболее успешных оказалась Bourne Shell, разработанная Стивом Борном (Steve Bourne). Эта оболочка не вошла в проект GNU при его зарождении, однако специально для него была создана новая, открытая командная оболочка, получившая название Bourne Again Shell. В этом названии содержится игра слов: слова Bourne и bourne произносятся одинаково, однако последнее означает «рождаться», «появляться на свет». Иными словами, BASH — это почти та же Bourne Shell, однако появившаяся на свет вновь, и уже — в рамках проекта GNU.*

### §6.3. Общие сведения о командной строке

Прежде, чем перейти к перечислению команд Linux и начать приводить примеры из использования, необходимо озвучить хотя бы минимум сведений о том, как устроена командная строка, какой она использует синтаксис и как, собственно, вводятся команды.

Итак, работая с командной строкой Linux, необходимо знать, что часть символов, которые вы, скорее всего, будете вводить с клавиатуры, являются служебными, то есть используются для создания дополнительных описаний к командам, для задания параметров и т. п. Вот они:

```
~ ! @ # $ % ^ & * ( ) - _ [ ] { } ; : ' " / \ < >
```

Однако все из них по большому счету годятся и для использования в имени файла. Если заключить имя создаваемого файла или каталога в 'одинарные кавычки', то все символы, находящиеся между ними, потеряют свое специальное значение.

Касательно командной строки Linux, нужно отметить еще один важный факт: команды Linux часто имеют параметры или ключи (так подобные функции называются в среде DOS). Они расширяют действие той или иной команды, а некоторые команды вообще не исполняются без заданных параметров, например, команды копирования файлов или каталогов. Параметр отделяется от самого текста команды пробелом и знаком «-», стоящим вплотную перед ним. Однако существуют и параметры «второго уровня», которые отличаются от параметров первого уровня прежде всего сферой действия. Они отделяются от команды знаком «- -». Параметр может служить для самых различных целей, которые будут рассмотрены ниже, применительно к конкретным командам.

Отдельно следует сказать и о внешнем виде командного интерпретатора shell. Если вы не используете режим X Window или любую из соответствующих ему графических сред (KDE, Gnome), то после загрузки системы, на черном экране, вы увидите приглашение к вводу команд, которое может быть двух типов:

`[user@localhost user]$` — это приглашение для обычного пользователя, который был создан на стадии установки системы и от имени которого вы загрузились.

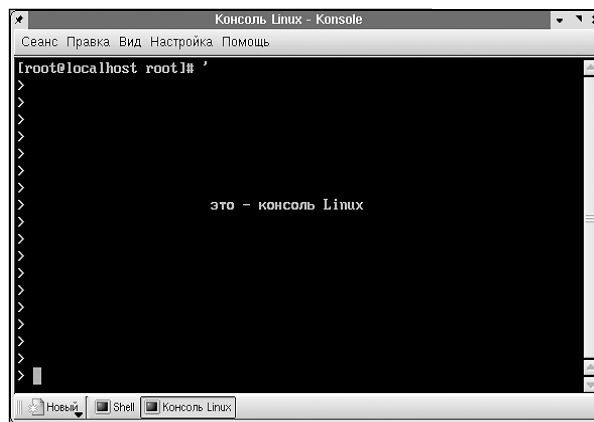
или

`[root@localhost root]#` — это приглашение для суперпользователя или администратора системы.

Как нетрудно заметить, конечный символ этих двух строк отличается, дабы с первого взгляда было понятно, кто работает с системой — простой пользователь или администратор.

Может возникнуть резонный вопрос: что делать, если с системой необходимо работать в современном графическом режиме с современными же приложениями? Как в таком случае запустить командную строку? На самом деле, все просто. В графическом режиме существует приложение, называемое терминалом. Его-то (рис. 6.1) и нужно запустить, чтобы попасть в shell и начать работу с командной строкой. Терминал запускается с панели задач KDE или Gnome, если там есть соответствующий значок. Если такого значка там нет, то ссылку на приложение можно найти в главном меню (меню «К» в KDE или меню а la «ступня гнома» в Gnome). Приложение достаточно удобно, и в связи с этим нужно отметить, что по сравнению с архаичной командной строкой DOS, командная строка Linux предоставляет немало удобных возможностей. Во-первых, это возможность в некоторых случаях перетаскивать отдельные участки текста мышью, во-вторых — наличие функции автодополнения, дописывающей команды или логически под-

Рис. 6.1.  
Консоль Linux





ходящие параметры. Наконец, в-третьих, оператор BASH русифицирован в подавляющем большинстве современных дистрибутивов, так что с отображением русскоязычных названий файлов и каталогов проблем быть не должно.

Таким образом, разобравшись с особенностями командного режима Linux, перейдем непосредственно к рассмотрению команд и их параметров, рассортированных по тематическим группам для удобства запоминания (а многие команды было бы неплохо именно запомнить). Список самых важных команд Linux, к некоторым из которых добавлены наиболее употребительные параметры, дан в приложении #1, для большего удобства поиска необходимой команды во внештатной ситуации.

#### §6.4. Команды Linux и их параметры

Ниже приведен список основных команд, которые могут потребоваться начинающему пользователю Linux, а также пользователю среднего уровня. На самом деле, команд Linux значительно больше — несколько сотен — но для первого знакомства с командной строкой операционной системы и выполнения базовых операций их вполне хватит.

##### §6.4.1. Базовые команды

К базовым командам ОС Linux относятся те команды, которые ответственны за выполнение простых системных операций, в основном, таких, которые позволяют получить определенные сведения о системе. В большинстве своем они не имеют дополнительных параметров.

- *X*           наверное, одна из самых используемых команд в Linux. С ее помощью запускается графический режим X Window.
- *whoami*   выводит имя пользователя системы, от имени которого она в данный момент загружена. Им может быть как суперпользователь (root), так и любой иной пользователь системы.
- *who*       выводит на экран список пользователей, работающих в системе в данный момент. Разумеет-

- *info*      ся, это актуально только при наличии сети из нескольких компьютеров или терминалов. команда, выводящая информацию по использованию какой-либо команды. Если вы не знаете, для чего служит команда, набирайте `info` и нажимайте Enter.
- *man*       название этой команды происходит от английского `manual` — руководство по эксплуатации, справочник. Фактически, `man` — это лучшая справка по Linux. Она подразделена на несколько глав, переключение между которыми осуществляется при помощи простых нажатий определенных клавиш на клавиатуре (рис. 6.2).
- *xman*      эта команда — почти полный аналог справки `man`, за тем исключением, что позволяет просматривать страницы пользовательского руководства и в графическом режиме, что, конечно, более удобно.
- *help*      еще одна команда для получения необходимой справки по какой-либо из команд, но не только. Фактически, при помощи команды `help` можно получить доступ к почти любым формам документации.
- *echo*      это команда, получившая свое широкое распространение еще в Unix-системах. В те времена она позволяла быстро и легко обмениваться сообщениями между пользователями внутренней сети. Дело в том, что `echo` выводит на экран написанное сообщение. В частности, если набрать в командной строке следующее:

Рис. 6.2.  
Команда `man`

```

root@localhost:~ - Shell - Konsole
Сеанс Правка Вид Настройка Помощь

<standard input>:15: realpath on `man1/bash.1' failed: Нет такого файла или каталога
man
BASH_BUILTINS(1)                                BASH_BUILTINS(1)

NAME
  bash, :, .. [, alias, bg, bind, break, builtin, case, cd,
  command, compgen, complete, continue, declare, dirs, dis-
  own, echo, enable, eval, exec, exit, export, fc, fg, for,
  getopts, hash, help, history, if, jobs, kill, let, local,
  logout, popd, printf, pushd, pwd, read, readonly, return,
  set, shift, shopt, source, suspend, test, times, trap,
  type, typeset, ulimit, umask, unalias, unset, until, wait,
  while - bash built-in commands, see bash(1)

BASH_BUILTIN COMMANDS
SEE ALSO
  bash(1), sh(1)

GNU Bash-2.05a                2001 October 29                BASH_BUILTINS(1)
(END)

```

*echo any text you like*

то на экран будет выведен именно ЭТОТ текст:

*any text you like*

Впрочем, команда `echo` и в настоящее время имеет практическое применение. Одним из наиболее известных вариантов ее использования является проверка используемой командной оболочки. Если набрать в командной строке

*echo \$shell или echo #shell*

то выведенным на экран результатом будет, например,

*/bin/bash*

Это означает, что в данном случае оболочкой является BASH, файлы которой расположены в каталоге `/bin`.

#### ***§6.4.2. Управление загрузкой компьютера***

Этот раздел включает в себя те команды, которые используются в основном для завершения работы системы, поскольку при загрузке системы никаких команд, собственно, и не требуется — достаточно ввести имя пользователя и пароль. Все остальное ОС Linux сделает сама.

Что касается завершения работы, то для этой процедуры в Linux существует две команды:

- *halt* Это простая команда. Она не имеет ни опций, ни синтаксиса и служит для полной остановки системных процессов и выключения компьютера.
- *shutdown* Эта команда уже несколько сложнее, поскольку имеет свой синтаксис и свои параметры. Вообще она переводится как «завершение работы системы», которое в Linux бывает двух типов (соответственно, задано двумя опциями):  
-r от *англ.* reboot — перезагрузка. С этой

опцией компьютер перезагружается.

`-h` от *англ.* halt — остановка. При использовании этой опции происходит окончательное выключение компьютера.

Однако нужно заметить, что если вы используете команду `shutdown`, вы не сможете ее выполнить, пока не пропишете еще один параметр — время, через которое должно происходить выключение или перезагрузка компьютера. Время здесь выражается в минутах. Вот несколько примеров использования команды *shutdown*.

`shutdown -h 0` немедленное выключение компьютера.  
`shutdown -r 0` немедленная перезагрузка компьютера.  
`shutdown -r +5` перезагрузка компьютера ровно через пять минут.

*Примечание.* Вместо числа «0» в качестве параметра времени, командная оболочка BASH разрешает использовать и слово «now», однако первый вариант представляется более удобным и быстрым.

#### §6.4.3. Работа с файлами

Работа с файлами в Linux сходна с аналогичной процедурой в других операционных системах. Файлы в Linux можно создавать, удалять, переименовывать, копировать, переносить, искать и делить на части. Однако помимо всех этих простых команд, последней в нижеследующем списке стоит команда `mount`, которая в операционной системе Linux имеет очень большое значение.

- `ls` Эта команда может оказаться полезной, когда потребуется вывести на экран список файлов, содержащихся в том или ином каталоге. Это одна из простейших команд, используемых при работе с файлами в Linux, однако ее незаменимость в определенных ситуациях очевидна: знать, что находится в нужном каталоге необходимо, особенно, когда работа с системой

ведется исключительно в режиме командной строки. Кстати, использование этой команды имеет свои особенности. Например, если ввести команду в следующем виде:

```
ls *.txt
```

на экран будут выведены все текстовые файлы, имеющие расширение \*.txt. Удобно, не правда ли? Ну, а если потребуется просмотреть содержимое вашего домашнего каталога, не обязательно набирать:

```
ls $home
```

хотя это тоже правильно; достаточно набрать всего лишь:

```
ls ~
```

где тильда (~) — общепринятое в Linux обозначение домашнего каталога пользователя. Ну, а если потребуется более полно ознакомиться со всеми возможностями команды ls, всегда можно обратиться к справке man, введя в данном случае:

```
man ls
```

- *cp*

Название этой команды происходит от английского сору — копировать. Поэтому нетрудно догадаться, что команда предназначена для копирования файлов или каталогов. Синтаксис программы прост: сначала пишется сама команда, затем имя файла, который нужно скопировать, а потом имя конечного файла, который и будет являться копией исходного. Выглядит это примерно так:

*cp имя\_файла1 имя\_файла2*

Однако, это самая простая конструкция. В данном случае файл будет скопирован в тот же каталог, в котором находился исходный. Но что делать, если файл нужно скопировать в другой каталог? Командой *cp* предусмотрена такая возможность. Вот как это делается:

*cp ~/каталог1/имя\_файла1 ~/каталог2/имя\_файла2*

Кстати, если в конце команды не задать конечное имя файла, то файл скопируется в другой каталог под тем же самым именем. Если же задать имя отличным от исходного, то команда копирования совместится с командой переименования.

- *mv* От английского *move*, означает «передвижение». Следовательно, при помощи этой команды, можно перенести файл из одного каталога в другой. Синтаксис этой команды точно такой же, как у команды копирования:

*mv ~/каталог1/имя\_файла1 ~/каталог2*

- *Rm* Команда отвечает за удаление файла из файловой системы. Интересная особенность команды состоит в том, что при удалении файлов она может использовать так называемые «маски». Например, если написать

*rm файл\**

то после нажатия клавиши <Enter> будут удалены все файлы из каталога, начинающиеся на «файл», однако, имя которых этим не заканчивается: будут удалены и «файл1», и «файл2», и «файлы». Причем файлы могут иметь разные

расширения, например, TXT и JPG, это на процесс удаления никак не повлияет.

Естественно, при использовании команды *rm*, можно указать и несколько разных имен файлов, все они будут удалены.

Важное замечание: в операционной системе Linux есть такое понятие, как рекурсия при выполнении команды. Это означает, что при добавлении, скажем, к команде *rm* атрибута *-r*, при ее исполнении будут удалены все файлы из каталога, в котором в данный момент находится пользователь. Также будут удалены и все подкаталоги с содержащимися в них файлами, поэтому использовать команду *rm* с этим атрибутом нужно осторожно и только при уверенности в необходимости ее выполнения.

- *rmdir* Эта команда отличается от предыдущей только тем, что с ее помощью удаляются не файлы, а целые каталоги. Синтаксис команд *rm* и *rmdir* одинаков:

```
rmdir имя_каталога
```

Так будет удален определенный каталог, находящийся в данном случае в домашнем каталоге пользователя.

- *cd* Эта команда служит для перехода в определенный каталог. Зачастую такой переход бывает необходим, чтобы запустить программу или скрипт, который не установлен в системе. В таком случае простой набор имени программы не поможет — ее нужно запустить именно из того каталога, где находится исполняемый файл. Синтаксис программы достаточно прост. Например, для того, чтобы перейти в каталог */mnt/cdrom*, нужно набрать

```
cd /dev/cdrom
```

- *find* Это команда, которая служит для поиска файлов и каталогов. Команда *find* является достаточно продвинутой: с ее помощью можно осуществить поиск как конкретного файла, так и поиск по типу файла, дате его изменения или размеру. Вообще, синтаксис поиска для этой команды выглядит следующим образом:

*find* *каталог* *критерий\_поиска*

где каталог — это название каталога, причем как одного, так и нескольких, в которых нужно искать необходимый файл. *Критерий\_поиска* — это собственно та опция команды, которая необходима на данный момент (поиск по имени, дате, размеру и т. п.).

Вот несколько основных опций:

–*name* при ее использовании поиск осуществляется по точному имени файла.

–*type* осуществляется поиск по типу файла. Однако тип файла не нужно путать с его расширением, как это принято в среде DOS. Файлы в Linux различаются по следующим типам:

§ *b* — блок-ориентированные устройства

§ *c* — байт-ориентированные устройства

§ *d* — каталоги

§ *f* — простые файлы

§ *l* — символические ссылки.

Обо всех этих типах файлов говорилось в главе «Файловая система Linux».

–*mtime* осуществляется поиск файлов, измененных указанное число дней назад. Например, *-mtime 7* — неделю назад.

Ниже следуют несколько примеров поиска файлов по команде *find*:

*find /home/catalog -name file.bmp* Такую команду можно задать, если необходимо найти файл с названием *file* и рас-



ширением BDP в подкаталоге `catalog` домашнего каталога `/home` пользователя.

`find /root/docs /root /usr/bin -mtime 5 -type l` По этой команде будет осуществлен поиск всех символических ссылок, измененных пять дней назад и находящихся в трех указанных каталогах.

- *mkdir* Эта команда отвечает за создание каталога. Синтаксис этой команды тоже очень прост. Если написать

```
mkdir catalog
```

то будет создан каталог с одноименным названием в текущем каталоге, из которого вы работаете. Можно поступить иначе и создать каталог в абсолютно любом каталоге, находящемся сколько угодно глубоко по структуре. Например, команда

```
mkdir /root/files/new_files/catalog
```

создаст каталог `catalog` в глубинных каталогах суперпользователя (`root`). Кстати, если каталоги `files` и `new_files` не существовали, команда *mkdir* создаст и их.

- *cat* Очень интересная команда. По большому счету, она служит для вывода содержимого файла на экран. То есть, если записать

```
cat файл
```

то на экран будет выведено содержимое указанного файла. Однако особенность команды состоит в том, что она позволяет копировать содержимое файла в другой файл, что приведет к созданию его копии. Например:

```
cat файл1 > файл2
```

приведет к созданию копии файла (1) в том же каталоге. Впрочем, можно поступить и по-другому. Например, если нужно восстановить файл, предварительно разбитый на несколько частей (скажем, командой *split*), то и в этом случае можно воспользоваться командой *cat*. Ее синтаксис будет выглядеть так:

```
cat файл1 файл 2 > файл3
```

что, конечно, очень удобно, особенно для переноса файла на носителях маленькой емкости.

- *chown* Название команды произошло от английского *change owner* — поменять владельца. Как известно из предыдущей главы, каждый файл в Linux имеет своего владельца. Изменить этот атрибут может только владелец файла либо системный администратор, если ему открыт доступ к этому файлу (например, известен пароль к каталогу домашнего пользователя). Чтобы изменить владельца файла, нужно написать следующее:

```
chown имя_нового_владельца имя_старого_файла
```

- *chgrp* Эта команда аналогична предыдущей. Однако ее особенность заключается в том, что она позволяет менять группу, к которой принадлежит файл. Иногда и такое может потребоваться. Синтаксис этой команды точно такой же, как и у *chown*:

```
chgrp имя_новой_пользовательской_группы имя_старого_файла
```

- *split* В отличие от команды *cat*, предназначена для разбития одного файла на несколько частей. Это может потребоваться, если нужно перенес-

ти большой файл на носители меньшего размера. Например, видеофайл на flash-картах или файл mp3 или Ogg Vorbis на дискеты. Чтобы это сделать, потребуется выполнить команду *split* с определенным синтаксисом. Как это сделать — см. ниже:

*split имя\_большого\_файла имя\_полученного\_файла.*

Нужно отметить, что имя полученного файла можно указать одно. Система сама добавит к нему буквы aa, ab, ac и т. п. Иными словами, если исходный файл назывался, например, film, то полученные файлы будут носить имена filmaa, filmab, filmac и т. п. А теперь — внимание. При использовании команды *split* в том виде, как она указана выше, размеры полученных файлов будут даны по умолчанию и составят 1 мегабайт. Это не всегда удобно, поэтому, чтобы такого не случилось, можно прибегнуть к опции *-b* данной команды. Эта опция позволит указать размер получаемых на выходе файлов. Вот как это может выглядеть:

*split -b15000k film.mp4 film*

В этом случае размеры полученных файлов будут составлять 15 мегабайт каждый.

- *mount*

Заканчивая знакомство с командами, посвященными работе с файловой системой Linux, нельзя не упомянуть самую главную и, наверное, сложную из этих команд.

Mount можно перевести с английского языка как «монтирование» или «присоединение». Следовательно, эта команда Linux занимается монтированием файловой системы. Как вам уже известно из предыдущей главы, все файловые системы, используемые Linux (файловая

система жесткого диска и его разделов, CD, DVD, flash-карты и т. п.), должны быть предварительно подмонтированы. В противном случае, они не будут читаться. Конечно, файловые системы, точно необходимые для работы (жесткого диска, CD-привода), монтируются автоматически на стадии установки любого современного дистрибутива, однако необходимость монтировать (или, проще говоря, подключить) файловую систему самому может возникать очень и очень часто.

Как это делается? Предположим, что нужно присоединить файловую систему раздела жесткого диска, на котором установлена другая операционная система или хранятся данные, так, чтобы они могли быть доступны и из Linux. Для этого нужно выполнить команду *mount* в следующем виде:

```
mount /dev/hda1 /mnt/disk
```

Таким образом, файл устройства *hda1*, который на самом деле является главным разделом жесткого диска компьютера, подключается к системе. Это выражается в том, что его содержимое теперь доступно из каталога */mnt*, где, как известно, хранятся файлы всех устройств, имеющих файловую систему. Нужно отметить, что каталог */disk*, в котором будут отображаться файлы из примонтированного раздела, придется создать заранее. Иначе, команда не будет выполнена корректно. Аналогично можно присоединить любое другое устройство.

Важное замечание. Если компьютер работает под управлением двух операционных систем, например, Linux и Windows, то почти наверняка к Linux придется присоединить и файловую систему Windows-раздела. Делается это точно так же, как указано выше, однако есть один ню-

анс. Если версия Windows русифицирована, то наверняка в ней найдутся файлы с русскими именами и названиями. Но при стандартном монтировании, они окажутся нечитаемыми, поскольку в Windows, скорее всего, используется иная, чем в Linux, кодировка — CP1251 против KOI8-R. Чтобы такого не случилось, нужно выполнить команду `mount` в следующем виде:

```
mount -o iocharset=koi8-r,codepage=866 /dev/  
bda1 /mnt/disk
```

где *iocharset* — обозначение кодировки (в данном случае — *koi8-r*), а *codepage* — кодовой страницы. Далее все пишется как обычно: расположение файла монтируемого устройства — пробел — расположение конечного пути (точки монтирования).

Команда *mount* имеет несколько особых параметров, самыми важными из которых являются: *-t* при его использовании монтируются только определенные файловые системы. После этого параметра можно написать названия тех файловых систем (*fat*, *ext*), которые вы не хотите монтировать.

*-a* если добавить к команде *mount* этот параметр, то после нажатия клавиши <Enter> будут монтированы все файловые системы компьютера, которые указаны в каталоге */etc/fstab*.

*-o* с этим параметром для команды *mount* можно использовать определенные опции. Например, задать точную кодировку для всех файлов так, как это было показано выше.

#### §6.4.4. Администрирование системы

Возможно, администрирование — слишком громкое слово для заглавия последнего раздела команд, однако команды в нем собранные действительно предназначаются для администрирования системы, хотя и на пользовательском

уровне, тем не менее, достаточно для работы с компьютером под управлением Linux дома или в офисе. Вот три команды, которые посвящены управлению системными процессами и которые наверняка окажутся полезными:

- *ps* это очень важная команда для любого администратора системы, поскольку она позволяет выводить на экран список идущих в системе процессов. Принцип действия этой команды приблизительно аналогичен нажатию <Ctrl>+<Alt>+<Del> в Windows с тем исключением, что в Linux все более наглядно. Команда *ps* имеет несколько опций.

–*e* на экран выводится список всех системных процессов.

–*ef* отображение процессов выводится в виде расширенной таблицы (рис. 6.3).

–*u* при использовании данного атрибута на экран выводится информация о том, на сколько процентов загружает центральный процессор тот или иной процесс.

- *top* команда отображает список системных процессов, но в режиме реального времени. После ее запуска нажатием определенных клавиш (в английской

Рис. 6.3.

Список процессов после выполнения команды *ps -ef*

```

[root@localhost root]# ps -ef
UID        PID     PPID  C  STIME TTY          TIME CMD
root         1         0  0  00:33 ?        00:00:04 init
root         2         1  0  00:33 ?        00:00:00 [keventd]
root         3         1  0  00:33 ?        00:00:00 [klogd]
root         4         1  0  00:33 ?        00:00:00 [ksftirqd_CPU01
root         5         1  0  00:33 ?        00:00:00 [ksuapd]
root         6         1  0  00:33 ?        00:00:00 [kbfuse]
root         7         1  0  00:33 ?        00:00:00 [kupdated]
root         8         1  0  00:33 ?        00:00:00 [kdracorey]
root         64         1  0  00:34 ?        00:00:00 [khubd]
root        559         1  0  00:34 ?        00:00:00 syslogd -n 0
root        564         1  0  00:34 ?        00:00:00 klogd -x
rpcuser     594         1  0  00:34 ?        00:00:00 portmap
rpcuser     612         1  0  00:34 ?        00:00:00 rpc.statd
root        733         1  0  00:34 ?        00:00:00 /usr/sbin/apmd -p 10 -
root        800         1  0  00:34 ?        00:00:00 xinetd -stayalive -reu
root        841         1  0  00:34 ?        00:00:00 sendmail: accepting co
root        860         1  0  00:34 ?        00:00:00 gpm -t ps/2 -n /dev/mo
root        878         1  0  00:34 ?        00:00:00 crond
root        906         1  0  00:34 ?        00:00:00 /usr/bin/umnet-bridge
root        927         1  0  00:34 ?        00:00:00 /usr/bin/umnet-netifup
root        987         1  0  00:34 ?        00:00:00 /usr/bin/umnet-netifup
root       1001         1  0  00:34 ?        00:00:00 /usr/bin/umnet-dhcpd -
root       1003         1  0  00:34 ?        00:00:00 /usr/bin/umnet-natd -d
root       1022         1  0  00:34 ?        00:00:00 /usr/bin/umnet-dhcpd -
root       1051         1  0  00:34 ?        00:00:00 /usr/bin/umware-nmbd -
root       1071         1  0  00:34 ?        00:00:00 /usr/bin/umware-smbd -
xfs        1332         1  0  00:34 ?        00:00:01 xfs -droppriv -daemon
daemon    1368         1  0  00:34 ?        00:00:00 /usr/sbin/atd
root      1414         1  0  00:34 tty1      00:00:00 /sbin/mingetty tty1
root      1415         1  0  00:34 tty2      00:00:00 /sbin/mingetty tty2
root      1416         1  0  00:34 tty3      00:00:00 /sbin/mingetty tty3
root      1417         1  0  00:34 tty4      00:00:00 /sbin/mingetty tty4
root      1418         1  0  00:34 tty5      00:00:00 /sbin/mingetty tty5
root      1419         1  0  00:34 tty6      00:00:00 /sbin/mingetty tty6
root      1420         1  0  00:34 ?        00:00:00 /usr/bin/gdm -nodaemon
root      1428       1420  0  00:34 ?        00:00:00 /usr/bin/gdm -nodaemon
root      1429       1428  4  00:34 ?        00:00:34 /usr/bin/X11/X :0 -aut
root      1441       1428  0  00:35 ?        00:00:00 /bin/sh /usr/bin/start

```

раскладке) можно управлять как отображением этих процессов, так и ими самими. Например:

<A> сортировка процессов по времени и дате его начала  
<T> сортировка процессов по времени выполнения  
<P> сортировка процессов по загрузке им ЦПУ компьютера  
<M> сортировка процессов по использованию системной памяти  
<K> завершение выбранного процесса

- *kill* команда, завершающая выполнение любого процесса. Вот примеры ее использования:

*kill X* — моментальное отключение графического режима;  
*kill [имя программы или процесса]* — завершение работы программы или процесса, естественно, без сохранения данных.

### §6.5. Запуск программ из командной строки

Командная строка Linux — очень гибкий инструмент для работы с системой. Намного более гибкий, чем графический интерфейс. Однако его задача заключается не только в работе с файлами, каталогами, а также с запуском чисто системных команд, позволяющих, например, завершить работу компьютера. Помимо этого, при работе с командной оболочкой, всегда может возникнуть необходимость запустить какую-либо программу. Более того, запуск приложений — наиболее часто используемая процедура при работе с операционной системой. Осуществляется она очень просто: в большинстве случаев необходимо лишь написать название приложения, и оно будет запущено. Например, если ввести *sndconfig*, произойдет загрузка утилиты для настройки звука в Linux, а если написать *frozen-bubble*, то будет запущена одноименная игра.

Такой способ запуска программ эффективен только для тех программ, которые непосредственно установлены в Linux, то есть зарегистрированы в каталоге `/usr`. Если же приложение находится в другом каталоге, то, чтобы запустить его, потребуется указать полный путь к его файлу, который называется исполняемым (аналог `exe`-файла в DOS). Например, если программа находится в каталоге `/home/programs`, то для ее запуска в командной строке нужно прописать следующее:

```
/home/programs/имя_программы
```

либо при помощи команды `cd` перейти в этот каталог и запустить программу прямо оттуда.

Напоследок следует сказать, что командная строка Linux — действительно удобное (а иногда и единственно возможное) средство решения самых различных задач. Список команд, приведенный в данной главе, — далеко не полный. Существует очень много других команд, список которых всегда можно получить, введя команду *info*. Если вам потребуется узнать более широкий перечень того, как можно использовать ту или иную команду Linux, вы всегда сможете ввести в окне терминала или прямо в командном режиме

```
man имя_команды
```

и узнать все, что вам требуется.



## Глава седьмая. Установка Linux

Еще несколько лет назад, когда операционные системы на основе ядра Linux не получили широкого распространения, их установка была достаточно сложным делом для неподготовленного пользователя. Действительно, инсталляция ранних версий дистрибутивов, таких как Debian GNU/Linux, Slackware и даже Red Hat, осуществлялась исключительно из командной строки. Естественно, это требовало от человека, пожелавшего установить Linux, не только знания основных команд этой операционной системы, но и, почти наверняка, хотя бы небольшого опыта работы с Unix.

Позднее ситуация изменилась в лучшую сторону. Linux сделала шаг навстречу простому пользователю. Это выразилось в том, что подавляющее большинство дистрибутивов обзавелись программами установки, позволяющими провести инсталляцию в графическом режиме. В связи с этим отпала необходимость учить команды, прежде чем провести штатную установку операционной системы. Разумеется, для нетрадиционной установки знание команд может потребоваться, но подобные ситуации встречаются нечасто, и такие знания необходимы разве что профессиональным системным администраторам.

Тем не менее, установка Linux даже при использовании графического инсталлятора несколько отличается от установки Windows и обладает своими характерными особенностями. Поэтому на этой теме следует остановиться подробнее.

### §7.1. Что нужно знать перед установкой?

Правила, которым нужно следовать при установке Linux, стандартны, и то, что можно сказать об установке Linux, можно сказать и об установке других операционных систем.

Прежде всего, необходимо собрать сведения о системе. Делается это для того, чтобы точно знать конфигурацию компьютера, когда на завершающей стадии установки потребуется настроить систему и оборудование. Список того, что нужно знать о компьютере, приведен во врезке.

Кроме получения данных сведений, необходимо подготовиться к установке и другим способом. Прежде всего, сделать копии всех необходимых данных — документов, программ и т. п. Конечно, при установке Linux как второй операционной системы, данные не будут удалены, но вероятность того, что как раз во время установки в системе может произойти сбой, тоже нужно учитывать.

Неплохо иметь под рукой загрузочную дискету вашей прежней операционной системы для того, чтобы иметь возможность загрузить ее в случае, если что-то пойдет не так. Впрочем, если действовать аккуратно и осмысленно, такого не случится.

## §7.2. Способы установки Linux

Linux, как и некоторые другие операционные системы, можно установить несколькими способами. Всего существует четыре основных способа:

- установка непосредственно с CD-ROM;
- установка с жесткого диска;
- установка с удаленного ресурса;
- установка с сетевого ресурса (через Интернет).

Под установкой операционной системы с сетевого ресурса чаще всего подразумевается установка системы из сети

### Что нужно знать о системе?

**BIOS:** фирма-производитель и номер версии.

*Просмотреть информацию о BIOS можно, нажав при загрузке <DEL> или иную клавишу, в зависимости от производителя компьютера.*

**Монитор:** фирма-производитель, максимальное разрешение и максимальная частота регенерации экрана.

**Жесткий диск:** фирма-производитель, тип интерфейса (IDE или SCSI), объем.

**CD-ROM:** фирма-производитель, тип интерфейса (IDE или SCSI).

**Видеокарта:** фирма-производитель, тип чипсета (например, GeForce 4 или ATI Radeon), объем видеопамати.

**Сетевая карта:** фирма-производитель. Также необходимо знать данные, требующиеся для работы в сети, а именно: ваш IP-адрес, IP-адрес шлюза, IP-адреса серверов доменных имен, сетевое имя вашего компьютера, имя домена вашей организации для того, чтобы сразу после установки системы подключить компьютер к внутренней сети, если таковая имеется.

**Мышь:** фирма-производитель, тип (Serial, PS/2 или другой). Если мышь подключена к COM-порту, то номер порта (например, COM1).

Интернет. Дело в том, что некоторые производители дистрибутивов, например, ASP Linux ([www.asplinux.ru](http://www.asplinux.ru)), позволяют не только загрузить свою систему из сети, но и установить ее прямо с сервера. Для этого нужно предварительно установить на своем компьютере программу-инсталлятор и, подсоединившись к сети, начать установку. Разумеется, данный способ применим только при наличии хорошей линии связи (dial-up вряд ли подойдет для таких целей). Нужно отметить, что программа-инсталлятор должна уметь продолжать установку и после обрыва связи. Тем не менее, установка Linux таким способом не всегда оправдана.

Установка системы с удаленного ресурса удобна в том случае, если существует внутренняя сеть из нескольких компьютеров (например, LAN) и необходимо установить Linux за короткое время на все или выборочное число машин. Действует данная схема следующим образом: программа установки запускается на сервере сети (она может находиться как на компакт-диске и считываться в CD-ROM, так и на жестком диске) и одновременно посылает данные на те компьютеры, где необходимо установить Linux. Впрочем, данный способ имеет некоторые сложности из области системного администрирования, поэтому подробно в данной книге не рассматривается.

Установка системы с жесткого диска имеет ряд особенностей. Как известно, чаще всего Linux поставляется на компакт-дисках, однако в том случае, если система была загружена из сети Интернет в готовом к работе виде, а не в виде исходников, она представлена в виде файлов (например, распространенного формата \*iso; такой файл называется образом компакт-диска). И тогда, если записать эти файлы на CD невозможно (например, ввиду отсутствия пишущего CD-привода), систему нужно будет установить прямо с жесткого диска. К этому способу еще прибегают, когда в компьютере отсутствует CD-привод или BIOS компьютера не поддерживает загрузку с CD.

Установка системы непосредственно с загрузочного компакт-диска объективно является самым удобным способом, который подходит в подавляющем большинстве случаев. Действительно, нет ничего проще, чем вставить диск из

дистрибутива в привод компьютера и запустить программу-инсталлятор. Однако и здесь есть свои особенности.

Прежде всего, необходимо определиться, будете ли вы окончательно переходить на Linux или же установите Linux как вторую ОС, параллельно с Windows. Рассмотрим оба варианта.

Установить Linux как единственную операционную систему несложно: достаточно изменить в BIOS порядок загрузки так, чтобы первым опрашиваемым устройством являлся CD-ROM. Затем необходимо загрузиться с компакт-диска и следовать указаниям программы установки. Устанавливать же Linux как вторую ОС несколько дольше. Однако эта процедура отличается от установки Linux на чистый жесткий диск только на начальном этапе. Поэтому логично будет подробнее рассмотреть установку Linux как одной из операционных систем на мультисистемном компьютере, иными словами, компьютере, работающем с несколькими операционными системами.

### §7.3. Установка Linux как второй ОС

Если вы решили установить Linux, но пока не удалять старую операционную систему, то вам потребуется проделать ряд дополнительных процедур. Ниже представлены те основные этапы, через которые нужно пройти, чтобы установить Linux так, чтобы иметь возможность пользоваться и ей, и другими установленными на компьютере операционными системами.

- Первая процедура — это *разбиение жесткого диска на логические разделы*. Как известно, две операционные системы нельзя установить на один раздел жесткого диска. В противном случае одна из них будет неработоспособной. Поэтому обычно, когда создают мультисистемный компьютер, имеют дело именно с несколькими разделами жесткого диска. Существует ряд основных способов разбиения жесткого диска на разделы. Можно прибегнуть к услугам программы-инсталлятора Linux или же воспользоваться специализированными программами. По большому счету, процедуры, выполняемые в этих двух случаях, сходны, но лучше

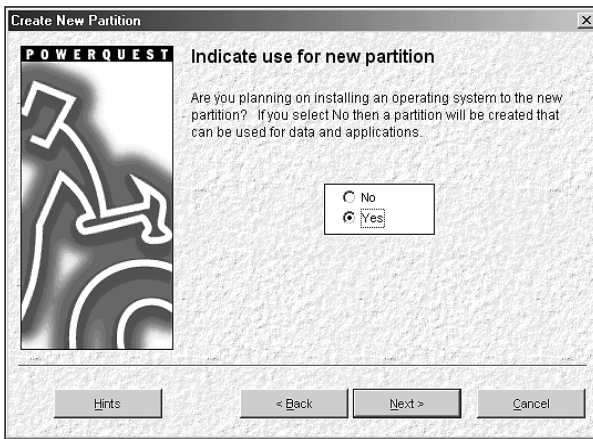


Рис. 7.1.

Создание нового логического диска в Partition Magic

Рис. 7.2.

Выбор типа файловой системы для нового логического раздела



будет показать разбиение жесткого диска на разделы именно на примере программы стороннего производителя, тем более что этот способ объективно удобнее. Возникает вопрос: какую программу использовать? Входящая в состав Windows DOS-утилита *fdisk*, к сожалению, не подойдет для этих целей, поскольку не умеет работать с теми файловыми

системами, которые поддерживаются Linux, даже такими распространенными, как ext2, ext3 и Linux Swap. Поэтому лучшим вариантом будет использование программы Partition Magic от компании Power Quest ([www.powerquest.com](http://www.powerquest.com)).

- После того, как вы выбрали то программное обеспечение, которое, по вашему мнению, больше подойдет для разбиения жесткого диска на логические разделы, необходимо перейти к следующей процедуре, а именно — *определить, сколько места нужно выделить для основного раздела Linux, а также для раздела подкачки*. Основные рекомендации по этому поводу даны во врезке.

- Третьим этапом является собственно *создание этих разделов*. В нашем случае — при помощи Partition Magic. Работа

с программой заключается в том, что вначале создается просто раздел. Далее нужно обязательно указать, что этот раздел должен быть активным. Иными словами, таким, чтобы на него можно было установить операционную систему (рис. 7.1). После этого нужно определить тип операционной системы, которую предполагается установить, и тип файловой

системы для нее (рис. 7.2). В Partition Magic последних версий можно выбрать как ext2, так и ext3. Второй вариант предпочтительнее, так как ext3 является журналируемой файловой системой (подробнее о файловых системах можно прочитать в главе шестой). После этого указываются размеры будущих разделов и сохраняются изменения. Примерный результат проведенной работы можно увидеть на рис. 7.3.

● После того, как активные разделы на жестком диске созданы, можно переходить собственно к *установке системы*. Для этого нужно просто загрузиться с первого компакт-диска дистрибутива.

#### §7.4. Ход установки Linux

Ход установки Linux является характерной особенностью каждого дистрибутива. Именно по способу установки и тем ее параметрам, которые предлагает программа-инсталлятор, часто различают дистрибутивы, выявляют их достоинства и недостатки. Тем не менее, установка Linux в любой из ее разновидностей имеет немало общих черт. Ниже будут рассмотрены общие этапы установки Linux, а также выделены самые важные отличия в ходе установки для популярных дистрибутивов:

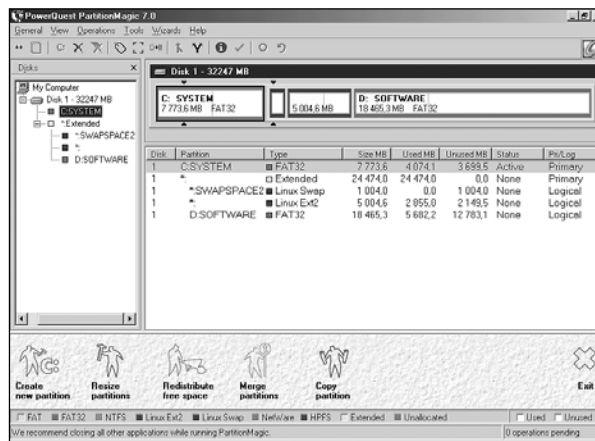


Рис. 7.3.

Результат создания Linux-раздела в Partition Magic

#### Сколько места выделить Linux?

*Как известно, Linux требует по меньшей мере два раздела: основной раздел Linux, куда будет установлена система и все приложения и программы, а также раздел подкачки, который объединяется с оперативной памятью и создает общую виртуальную память для ускорения работы операционной системы. Можно создать несколько разделов дополнительно к основному, дабы еще больше упорядочить файловую систему Linux, однако и одного раздела по большому счету достаточно.*

*Определение размера основного раздела во многом зависит от выбранных параметров установки системы. Например, если вы выбрали почти все предложенные в дистрибутиве компоненты, начиная от офисных программ и заканчивая играми, то меньше, чем гигабайтным разделом вам не обойтись. Но лучше, конечно, исходить, по крайней мере, из двух гигабайт свободного дискового пространства. Не стоит также забывать о том, что хотя в дистрибутивах Linux находится немало*

программ и программных пакетов, система также позволяет установить и новое программное обеспечение, приобретенное отдельно. Поэтому лучше сразу выделить для Linux побольше свободного места (оптимально 35 гигабайт), чтобы впоследствии не пришлось увеличивать размер логического диска посредством того же Partition Magic.

Отдельного упоминания заслуживает размер Swap-раздела. Оптимальным вариантом будет установить его в двух-трех кратном размере оперативной памяти. Иными словами, если на компьютере установлено 128 мегабайт RAM, то размер Linux Swap должен колебаться в пределах 250-300 мегабайт. В то же время, делать Swap-раздел большего размера бессмысленно, поскольку скорость считывания информации с жесткого диска относительно невысока и большие порции информации будут считываться так же медленно, как и с обычного раздела. Более того, если компьютер имеет 256 и более мегабайт оперативной памяти, Swap-раздел можно вообще не использовать.

Red Hat, Mandrake, ASP Linux, ALT Linux.

Установку Linux лучше всего будет показать именно на примере дистрибутива Mandrake. Причина, как нетрудно догадаться, заключается в наличии у данной модификации Linux наиболее наглядной программы-инсталлятора. Как было сказано, компания Mandrake первой сделала шаг навстречу конечному пользователю, специально разработав удобный и наглядный графический интерфейс установки. Наглядность состоит в отображении на экране каждого шага установки с краткими пояснениями к нему. Однако нужно отметить, что вышесказанное не стоит расценивать как похва-

лу исключительно Mandrake. Другие дистрибутивы (Red Hat, SuSE, Debian, отечественный ASP Linux и некоторые другие) тоже имеют понятные и удобные графические инсталляторы, работать с которыми можно не менее успешно, да и устройство они имеют во многом сходное с инсталлятором от Mandrake. Но перейдем собственно к процедуре установки.

Вообще, установка Linux, какого бы дистрибутива это ни касалось, состоит из нескольких этапов. Вот основные из них в порядке их прохождения пользователем, устанавливающим Linux:

#### *Запуск программы-инсталлятора*

Этот этап — самый простой. После загрузки компьютера с основного диска с дистрибутивом (или диска номер один, если дистрибутив состоит из нескольких дисков) вы попадаете в меню программы-инсталлятора. Его дизайн, внеш-

ний вид и количество отображаемых в первую очередь изменяемых параметров может варьироваться в зависимости от дистрибутива, однако суть любого графического инсталлятора одна и та же, поэтому вы всегда сможете легко разобраться в его настройках. Вот как выглядит стартовое окно инсталлятора в Mandrake (рис. 7.4), Red Hat (рис. 7.5) и ASP Linux (рис. 7.6 и рис. 7.7).

### Настройка программы-инсталлятора

После того, как инсталлятор загружен, начинается установка Linux как таковая. На этапе настройки инсталлятора вам предстоит указать несколько основных параметров:

- **Язык, на котором будет проходить установка.** Поскольку Linux — это операционная система, изначально ориентированная на многонациональное использование, в ней содержатся установки для многих языков. Естественно, на территории Российской Федерации наиболее вероятным будет использование русского языка. Здесь нужно также отметить, что многие дистрибутивы предоставляют пользователю право выбрать еще и кодировку. Если это касается только стадии установки, тут можно выбрать любую, будь то KOI8-г или CP1251.

- **Тип машины.** Этот параметр очень часто нужно указывать изначально, поскольку во время установки вышеоз-

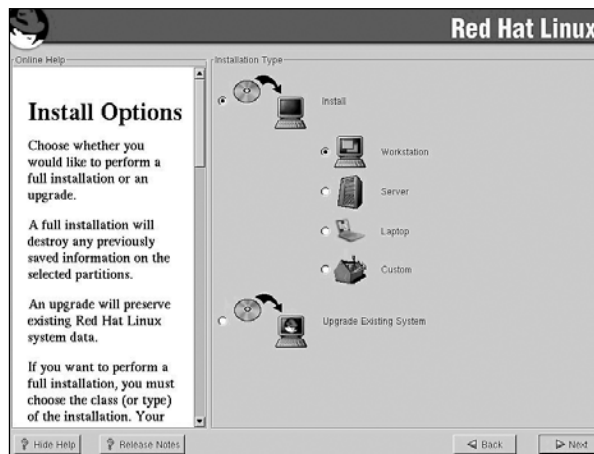


Рис. 7.4.

Окно инсталлятора Mandrake Linux

Рис. 7.5.

Окно инсталлятора Red Hat Linux





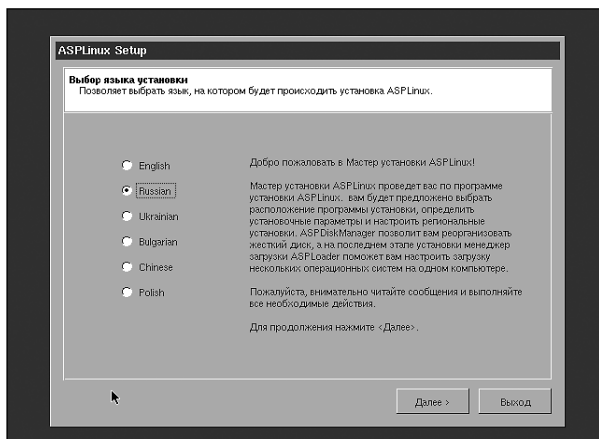


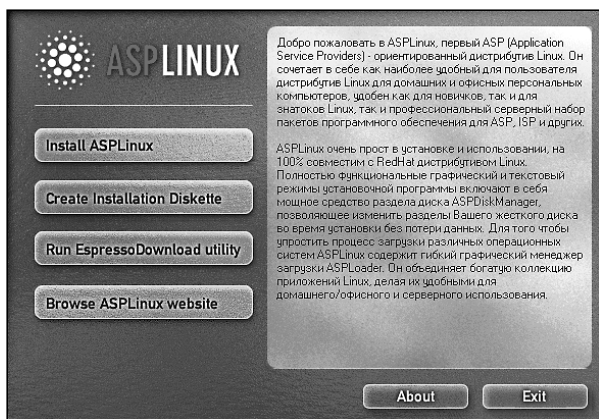
Рис. 7.6.

Окно инсталлятора ASP Linux

● **Выборочная установка.** Этот способ является самым распространенным и наиболее удобным. Конечно, он не позволяет автоматически установить систему, без необходимости указания многих параметров, но тем не менее, он является самым оправданным. При выборочной установке вам потребуется прежде всего выбрать необходимые программные пакеты для установки. О том, как это сделать, будет сказано чуть позже. Помимо этого, выборочная установка во многих дистрибутивах может предоставить и другую возможность — расширенное управление структурой разделов на жестком диске. Этот факт имеет большое значение в том случае, если вы устанавливаете Linux как вторую операционную систему. Подробнее об этом — тоже чуть позже.

Рис. 7.7.

Окно инсталлятора ASP Linux под Windows



наченный дивайс придется использовать достаточно часто.

### Выбор типа установки

На этом этапе, так же как и на предыдущем, нет ничего сложного. Всего в природе существует только три типа установки, из которых только один является широко используемым. Вот они:

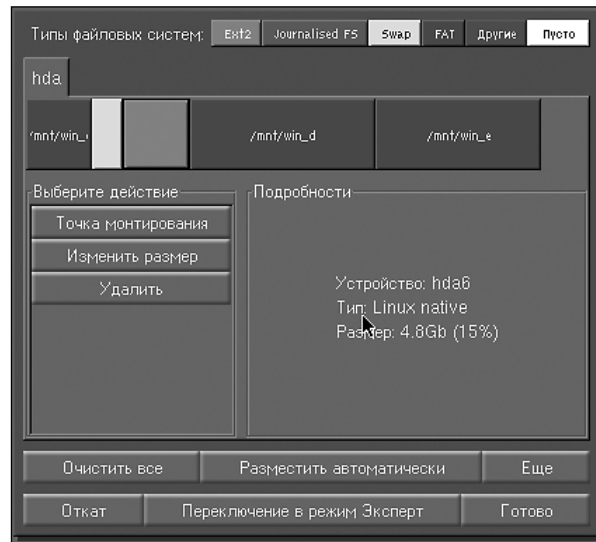
● **Быстрая установка.** Этот способ хорош, когда Linux нужно установить быстро, в ее усредненной конфигурации, например, на десяток компьютеров в офисе. Однако реальное удобство чувствуется в основном тогда, когда Linux устанавливается как единственная система на компьютере, и при этом не нужно добавлять

или редактировать логические дисковые разделы. Во всех других случаях выборочная установка объективно предпочтительнее.

● **Обновление системы.** Эта возможность предоставляется только в тех дистрибутивах, которые являются пакетными (подробнее о том, что это такое, сказано в главе четвертой) и основаны, скорее всего, на пакетах RPM — Red Hat, Mandrake, ASP Linux и некоторых других. Это означает, что установив, например, Red Hat, при использовании обновления вы сможете установить поверх него либо новую версию Red Hat (если не требуются новые, входящие в свежий дистрибутив программы, иногда проще просто заменить ядро системы на более современную версию), либо другой дистрибутив, состоящий из RPM-пакетов.

#### *Изменение структуры разделов жесткого диска*

Эта процедура несколько сложнее предыдущих, однако разобраться в ней тоже не составит особого труда. В разных дистрибутивах окно, где требуется создавать, редактировать и удалять логические разделы, выглядит по-разному. Как оно выглядит в Mandrake, можно увидеть на *рис. 7.8*. Из рисунка видно, что вам потребуется определить точку монтирования для уже созданного Linux-раздела. В случае, если вы устанавливаете систему на чистый винчестер, точкой монтирования является весь основной физический раздел. Точка монтирования (/) устанавливается обычно в одноименном пункте меню. Также необходимо подключить swap-раздел (если он будет использоваться) и иные разделы, если требуется, чтобы данные, содержащиеся в них, были доступны и под Linux. Что касается создания новых логических дисков и удаления старых (если они были), то каждый дистрибутив



**Рис. 7.8.**

**Редактирование логических разделов при установке Mandrake**

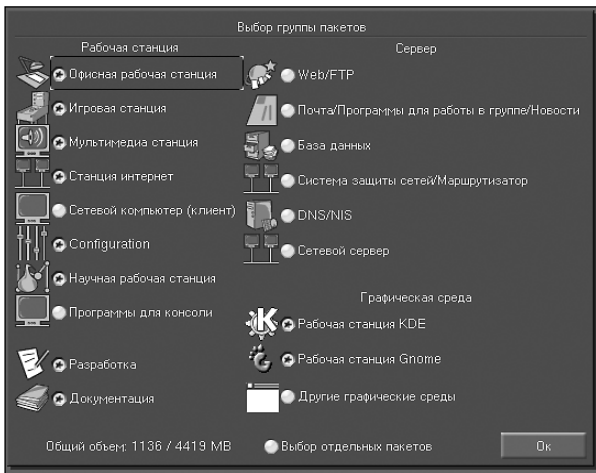


Рис. 7.9.

#### Выбор пакетов для установки

является выбор программных пакетов, то есть тех программ, приложений и других средств (драйверов, частей ядра, исходных кодов), которые необходимы для продуктивной и комфортной работы. Конечно, в любом дистрибутиве Linux есть возможность установки новых программ после установки системы, однако часто бывает удобнее сразу установить то, что точно будет нужно. На *рис. 7.9* показан пример выбора пакетов на стадии установки Mandrake. Следует отметить, что изображенный список компонентов является достаточно сильно обобщенным, а, следовательно, выбор компонентов хоть и не тонок, но позволяет избежать ошибок, таких как ошибки зависимостей между пакетами. Если же нужно выполнить более тонкий выбор, то в Mandrake необходимо переключиться в режим эксперта, в котором будет доступен список всех возможных RPM-пакетов для установки. Аналогично подобная процедура выполняется и при установке других дистрибутивов.

После завершения вышеозначенных процедур взаимодействие пользователя с программой-инсталлятором ненадолго заканчивается. Производится форматирование Linux-раздела и собственно установка системы, выраженная в копировании файлов. Время копирования зависит от количества устанавливаемых компонентов. Если дистрибутив Linux многодисковый (скорее всего, так и есть, хотя существуют и версии Linux на одном DVD, примером может слу-

предоставляет свои более или менее удобные средства для осуществления этой процедуры. Ими вполне можно воспользоваться, если по каким-то причинам было невозможно отредактировать структуру разделов заранее, перед установкой системы.

#### *Выбор пакетов программ*

Следующей процедурой, характерной именно для выборочного типа установки,

жить дистрибутив SuSE), то в процессе установке придется также заменять диски.

### §7.5. Завершение установки и настройка системы

Итак, файлы с ядром системы и выбранными программными пакетами перенесены на жесткий диск компьютера, однако установка Linux на этом не заканчивается. Предстоит еще проделать ряд процедур, причем как в программе-инсталляторе, так и впоследствии — уже после перезагрузки компьютера. Список этапов заключительного периода установки системы можно увидеть ниже:

*Настройка учетной записи root и добавление новых пользователей*

На этом этапе система запрашивает пароль для системного администратора. Именно на этой стадии можно создать сколько угодно новых пользователей системы и присвоить (если это необходимо) им пароли.

*Настройка оборудования*

После установки Linux начинается процесс распознавания установленного на компьютере оборудования. Вообще, подробно вопросы установки и настройки оборудования рас-

### Что такое пакетная зависимость?

*Помимо того, что операционная система на основе ядра Linux имеет не одну модификацию, называемую дистрибутивом, сами дистрибутивы имеют несколько разновидностей. Наиболее распространенными являются пакетные дистрибутивы. Самыми популярными из них считаются, как уже было сказано, Red Hat, Mandrake, ASP Linux и некоторые другие. Их особенностью является то, что все их составные части, включая ядро, заключены в RPM-пакеты, которые устанавливаются как изначально, так и отдельно. На стадии установки дистрибутив обычно предлагает выбрать те пакеты, которые нужно установить. И тут нужно проявить осторожность. Несмотря на то, что дистрибутив Linux состоит из отдельных пакетов, многие из них связаны между собой и образуют так называемые пакетные зависимости. Иными словами, один пакет может и не работать без другого. Конечно, ситуация во многом поправима — недостающий пакет всегда можно доустановить. Более того, все инсталляторы всех современных дистрибутивов имеют встроенную функцию контроля зависимостей. Тем не менее, совет при установке системы достаточно прост — никогда не исключайте ни один пакет из списка установки, если хотя бы примерно не знаете его назначения. В противном случае неработоспособными могут оказаться не только отдельные программы, но и вся система. Впрочем, если хорошо разбираться в названиях пакетов, можно добиться «идеальной» установки системы, оптимизированной под конкретного пользователя и не содержащей ничего лишнего.*

сматриваются в одной из глав четвертой части, однако нужно сказать, что на стадии установки необходимо иметь под рукой те данные об аппаратуре, которые были рекомендованы несколько выше, поскольку такие процедуры, как настройка монитора, мыши, клавиатуры, модема и принтера обычно выполняются именно на этом этапе.

#### *Установка загрузчика*

Даже если Linux устанавливается как единственная система на компьютере, ей потребуется загрузчик, записываемый в Master Boot Record. Этим загрузчиком обычно является LiLo, реже — Grub. Оба из них должны быть доступны для установки в любом дистрибутиве. Некоторые дистрибутивы имеют и свои оригинальные загрузчики, например, в ASP Linux это — ASP Loader. Какой из них выбрать, большого практического значения не имеет. Фактически, это дело вкуса.

#### *Настройка графического интерфейса X Window*

По окончании вышеописанных процедур, система Linux уходит на постинсталляционную перезагрузку, после которой начинается загрузка графического режима и его предварительная настройка. На этом этапе нужно выбрать такой важный параметр, как язык системы. Не только программ-инсталлятор, но и сама система (это касается подавляющего большинства дистрибутивов) являются многоязыковыми, причем даже созданные в России ASP Linux и ALT Linux предоставляют практически полный набор языков на выбор.

В период настройки X Window нужно выбрать графическую среду и графическую оболочку, ей соответствующую. Подробнее о том, какие существуют графические среды и какие преимущества имеет каждая из них, можно прочитать в главе «Графические оболочки Linux». Естественно, выбор параметров графического режима на стадии установки не является окончательным. Эти настройки можно менять при каждой очередной загрузке операционной системы.

### **§7.6. Переустановка и удаление Linux**

Переустановка Linux может потребоваться лишь в том случае, если произошел окончательный крах системы и ис-

править положение другими способами невозможно. Впрочем, если вы не знаете, как разрешить ту или иную проблему в Linux, которой раньше не существовало, вы тоже можете попробовать переустановить систему, чтобы потом разобраться в ситуации. Если Linux — единственная система, установленная на жестком диске, есть только один вариант: сохранить на надежном носителе все необходимые данные и прибегнуть к форматированию винчестера, так как установка Linux поверх существующей системы все равно уничтожает все данные. Если же Linux — вторая (третья, четвертая) система на жестком диске, то чтобы ее переустановить, вовсе не обязательно удалять Linux-раздел — программа-инсталлятор отформатирует его и установит Linux заново. Тем не менее, надо обязательно сохранить в надежном месте всю нужную информацию.

Удаление Linux (естественно, установленной как вторая операционная система) заключается в удалении Linux-раздела либо в его форматировании, чтобы использовать освободившееся пространство для другой ОС. Выполнить эти процедуры можно уже упомянутой программой Partition Magic от Power Quest или любой иной подходящей для этих целей утилитой. Но удалить Linux-раздел вместе с Linux Swap — еще не значит избавиться от загрузчика вроде LILO, позволяющего выбрать систему для загрузки. Чтобы он не появлялся, нужно восстановить старую загрузочную запись MBR. Чтобы это сделать, нужно воспользоваться DOS-утилитой *fdisk* (*fdisk.exe*). Необходимо запустить утилиту *fdisk* в самом DOS или же в сеансе MS-DOS под Windows с недокументированным ключом */mbr*. Альтернативным вариантом может оказаться запуск команды из меню «Выполнить» (рис. 7.10). Иными словами, командная строка выполнения означенной процедуры будет выглядеть так:

```
C:/>fdisk /mbr
```

Рис. 7.10.  
Запуск *fdisk* с  
ключом */mbr*

