

ВНУТРИ: 8-ГБ диск, полный дистрибутивов и программ

LINUX FORMAT

Главное в мире Linux

Сентябрь 2010 № 9 (135)

Сможет ли web ► m убить Flash?

Новый видеокодек от Google в сети с. 50

В ЭТОМ НОМЕРЕ

Патенты на ПО: за и против

» Они тоже могут быть полезными?

Обзор: Qt Creator 2.0

» Создавать мобильные приложения стало проще?

Bash: развивайте навыки

» Освойте язык командной оболочки Linux и пишите скрипты

Hugin: клеим панораму

» Если вам не дают покоя лавры Франца Рубо

Сравнение торрент-клиентов

» Качайте дистрибутивы правильно



Firefox

Отправит ли Chrome самый успешный открытый проект в истории на свалку истории?



“ Курт фон Финк
По-моему, кое-кто склонен создавать антагонизм на пустом месте ”

Больше о MySQL и NoSQL на с. 28

Плюс! Launchpad

Начните свой открытый проект сегодня!

РАБОЧИЙ СТОЛ:

Доки

» Превратите Linux в Mac OS X

MANTIS

Контроль ошибок

» Неполадки в коде: от регистрации до устранения

HARDCORE LINUX

Procmail

» Рецепты для... сортировки входящих сообщений?

Подписной индекс в каталоге агентства «Роспечать» 20882
Подписной индекс в каталоге «Пресса России» 87974
Подписной индекс в каталоге «Почта России» 16572

Linux center
www.linuxcenter.ru



Лучший свободный проект России – 2010



С вашей помощью мы хотим узнать, какие новые проекты по разработке свободного ПО появились в нашей стране и ее ближайших соседях за минувший год, и отобрать из них наиболее достойные инициативы, создатели которых будут награждены внушительными денежными премиями.

Мы понимаем, что в мире существуют как проекты-тяжеловесы, развиваемые коллективами из десятков и даже сотен человек, так и «малютки», насчитывающие одного-двух основных разработчиков. Исходя из этого, указанные премии будут вручаться в двух номинациях — для индивидуальных и для групповых проектов.

В порядке эксперимента, мы также решили **учредить в этом году еще одну, третью номинацию: «Осенний марафон»**, ориентированную в первую очередь (но не только!) на студентов. Начинается новый учебный семестр, а значит — пора браться за курсовые проекты (вы ведь не из тех, кто делает их в ночь перед защитой?). Выбрав для «курсовой» тему, связанную с одной из предложенных организаторами конкурса, и успешно справившись с поставленной задачей в срок, вы получите право на денежный приз. Более того, плоды вашего труда не лягут пыльной папкой в архивах любимой кафедры, а будут активно использоваться тысячами людей. Каждый из них скажет вам

«спасибо», а кое-кто может даже оказаться потенциальным работодателем — иконка вашей программы на панели быстрого запуска скажет о вас больше, чем строки резюме!

Мы верим, что наш конкурс — это хороший способ отблагодарить людей, чьими разработками пользуемся не только мы, но и все сообщество, и надеемся, что эти премии помогут им продолжать их важное и нужное дело.

Чтобы номинировать проект на конкурс, вам не обязательно быть его разработчиком! Просто заполните анкету на нашем сайте — и можете начинать болеть за свою любимую программу. Итоги конкурса будут подведены в декабре 2010 года и опубликованы на нашем сайте, а также в журнале *Linux Format* за январь 2011 года.

Победители будут определены решением компетентного жюри, в состав которого войдут представители компаний-организаторов, являющиеся экспертами в области свободного ПО и GNU/Linux.

Зарегистрируйте ваш любимый проект здесь:
www.linuxformat.ru/foss-contest

Мы приглашаем спонсоров поддержать наше начинание и увеличить призовой фонд Конкурса! Обращайтесь по адресу info@linuxformat.ru

Что мы делаем

» Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.

» Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.

» Мы выпускаем весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.

» Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.



Кто мы

Firefox славится богатой подборкой расширений. Поэтому мы решили спросить у команды LXF: о каком самом-самом дополнении вы всегда тайно мечтали?



Грэм Моррисон
Режим ретро-web: добавляет теги BLINK и MARQUEE к каждой странице.



Майк Сондерс
AutoStallman. Оно ищет на странице все вхождения слова «Linux» и предваряет его словом «GNU».



Нейл Ботвик
Как пользователь Gentoo, я жажду расширения, содержащего, когда скомпилируется Firefox. Минуточку...



Эфраим Эрнандес-Мендоса
«Режим босса». Клик – и с виду я как бы работаю, а не наблюдаю слив очередного матча сборной Мексики.



Эндрю Грегори
Индикатор UK Threat Level не особо-то страшен. Пусть мой дом окружает спецназ, если я не щелкаю по нему раз в 5 секунд.



Знди Ченелл
Фильтр неправых мнений – удаляет все страницы, содержащие высказывания, с которыми я не согласен.



Дэвид Картрайт
Как насчет эмулятора Internet Explorer? Не может отобразить страницы, падает и сливает номера кредиток на сторону.



Энди Хадсон
Firefox? Web? Что это? Не знаю таких слов – дайте что-то, чтобы просматривать Gorper, и ладно.



Ник Вейч
Злое расширение. Например, чтобы сразу помечало все видео с котятками на YouTube как нелегальный контент.



Сюзан Линтон
Инициатор войнушек – автоматически добавляет «И поэтому мой дистрибутив лучше» к каждому сообщению.



Шашанк Шарма
Режим ZX Spectrum: чтобы пиликал и мигал бордюром во время загрузки страниц.



Боб Мосс
Это неважно – я держусь подальше от Web с тех пор, как мне начали снится хищные коты.



Firefox или Chrome?

» Я впервые подключился к Всемирной сети (попросту говоря, Web) в 1997 году. Занятно осознавать, что примерно в то же время компания Red Hat представила свой менеджер пакетов RPM, которым многие из нас пользуются и по сей день, но речь сейчас не об этом. Мои первым браузером был Netscape версии, если не ошибаюсь, три с чем-то, и я еще помню те времена, когда за него надо было платить деньги. Пресловутая война браузеров разворачивалась у меня на глазах, и журналы с крупными надписями на обложке «Internet Explorer или Netscape Navigator?» все еще пылают где-то на антресолях.

Шли годы, Netscape очевидно старел – меня почему-то особенно огорчала невозможность использовать псевдокласс :hover в селекторах CSS. Мои друзья и знакомые переходили на Internet Explorer: сначала 4.x, затем 5.x... Когда технологический разрыв стал угрожающим, я пересел на Mozilla 0.6, потом – на Firefox 1.0. А потом появился Google Chrome.

Не стану плести интригу дальше – нет, в отличие от многих, я по-прежнему пользуюсь Firefox. Мне нравится WebKit как технология, мне нравится дизайн Chrome и скорость. Я нисколько не боюсь «всевидящего ока» Google. Я даже пробовал поставить Chrome как второй браузер – но дружбы из этого знакомства не вышло.

К чему я все это рассказываю? Большая часть материалов «Firefox против Chrome», включая и тот, что вы найдете на стр. 20, написана с точки зрения технологий, и почему-то мало кто рассматривает такой аспект, как привыкание. Согласен, что мой случай нетипичный, но ведь я, наверное, не один во всем мире? Верные сторонники Firefox, объединяйтесь!

Валентин Синецын, главный редактор

» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru

Подписка и предыдущие номера: subscribe@linuxformat.ru

Техническая поддержка: answers@linuxformat.ru

Проблемы с дисками: disks@linuxformat.ru

Общие вопросы: info@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

Web-сайт: www.linuxformat.ru

» Адрес редакции: Россия, Санкт-Петербург, Лиговский пр., 50, корп. 15

» Телефон редакции: (812) 309-06-86. Дополнительная информация на с. 112

Содержание

В котором мы с гордостью представляем свои труды в этом месяце.

Обзоры

Eschalon: Book II 8

Мы выпустили Алекса Кокса из клетки, чтобы он мог оценить ролевую игру в духе старой школы.



➤ Это похоже на пост-апокалиптического Властелина Колец, но для Linux.

Eclipse 3.6 9

Самая сложная интегрированная среда разработки в мире добавила новых функций. Что мы про это думаем?

WordPress 3 10

Это блог? Или система управления содержанием? Нет, это WordPress. Теперь – с новыми темами.

Qt Creator 2 11

Познакомьтесь с лучшим редактором исходных текстов C++ по эту сторону скорости света.

KOffice 2.2 12

А теперь познакомьтесь с нелучшим текстовым редактором, электронной таблицей и прочим.



➤ Графический редактор Krita – лучшая часть интегрированного офисного пакета KDE.

Книжные новинки 13

Искусство оптимизации MySQL и создание на его основе превосходных web-сайтов с Django.

Firefox vs Chrome

Выигрывает открытое ПО! с. 20

Что за штука...

WebM

Google пытается уничтожить H.264, пока он не вырос слишком большим. с. 50



Hugin с. 34

Склейте из своих отпускных снимков что-нибудь красивое.



Люди говорят



Монти сказал:
«Неспособным освоить SQL нечего и пытаться работать с NoSQL»

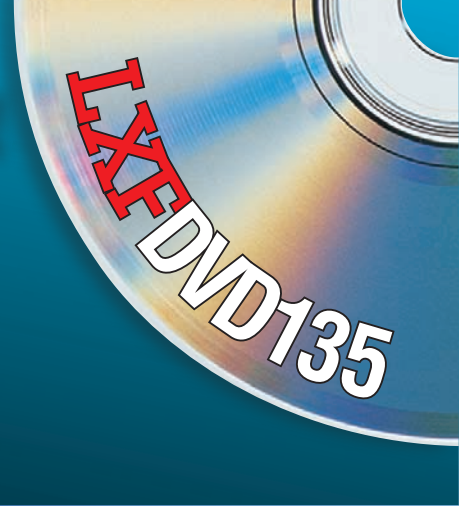
Курт фон Финк цитирует классиков с. 28

Диск с тройной загрузкой!

Три дистрибутива Linux + один в подарок

- » **Slackware 13.1** Система для настоящих линуксоидов
- » **Ubuntu Netbook Edition** Подойдет и для старых ПК
- » **Linux Gamers DVD** для игры и игровых
- » **PCLinuxOS 2010.07** Новый настольный дистрибутив

ПЛЮС: KOffice 2.2 и Firefox 4 Beta! **с. 100**



Ищите в этом номере...

Патенты на ПО 38

Сказать «не нужно» просто – сложнее разобраться, почему не нужно и действительно ли не нужно. Совершим небольшую экскурсию на территорию гражданского права.

Репозиторий Debian 42

Можно придумать множество причин для создания собственного репозитория – мы перечислим некоторые из них и снабдим вас необходимыми инструментами.

Linux в доке 46

Правильнее было бы сказать «доки в Linux» – рассмотрим несколько способов воссоздать одну из главных черт Mac OS X в открытой ОС.

Пропустили номер?

Узнайте на с. 107, как получить его прямо сейчас!



Постоянные рубрики

- Новости** 4
События мира Linux глазами наших экспертов.
- Сравнение** 14
В котором мы выбираем для вас лучший торрент-клиент.
- Интервью LXF** 28
Курт фон Финк, MySQL.
- Что за штука** 50
WebM: новый видеоформат, который может стать фатальным для H.264.
- Рубрика сисадмина** 52
Копирование файлов, циклические базы данных и Ubuntu Tweak. Такую
- микстуру мог приготовить только доктор Браун.
- Ответы** 88
Проблемы Linux решены: от загрузки с USB-брелка до web-камер.
- Hotpicks** 94
Лучшие в мире новинки свободного ПО.
- Диск Linux Format** 100
Содержимое двустороннего DVD этого месяца.
- Пропустили номер? ...** 107
Закажите себе один из предыдущих выпусков журнала.

Школа LXF 108
Свободное ПО и новый образовательный стандарт.

Через месяц 112
LXF136 выйдет заведомо раньше, чем релиз-кандидат GNU/Hurd.



» А вы не забыли оформить подписку на LXF?

Учебники

Начинающим Хитроумная мышь 58

Всем знакомый манипулятор умеет много больше, чем кажется. Плюс: управление фотографиями без Mono.



» KDE: ищем горшочек с золотом под одним из концов списка опций настройки.

OpenOffice.org Calc Снова оптимизируем 62

В прошлые разы мы научились писать быстрые функции – а теперь узнаем, что можно выжать из самого офисного пакета.

Подстройка Измерим скорости 66

Действительно ли ваш компьютер с Linux так быстр? Вряд ли. Этот разговор беспредметен, пока в нем не появятся числа.

Perl Улучшим фото 68

Известный как язык обработки текстов, Perl справится и со сложной графикой. И даже может где-то заменить Photoshop.

POV-Ray Применим текстуры 72

Чтобы созданные вами объекты выглядели реалистично, их надо как следует замаскировать. Текстуры для этого идеально подходят.

Mantis Отслеживаем ошибки 76

Слово «багзилла» скоро станет именем нарицательным, как и «гугл», но в мире не одна система регистрации инцидентов.

Bash Пишем скрипты 80

Новая серия статей о программировании на языке оболочки начинается с заявления, что вы уже на этом языке программировали.

Hardcore Linux Рецепты для почты 84

Если давно не обновлявшаяся программа по-прежнему на пике популярности – значит, в ней что-то есть. Узнайте, что это такое есть в Procmail, а заодно рассортируйте свои письма.



ГЛАВНОЕ Linux в ростовских школах » Westinghouse Digital Electronics заплатит за BusyBox » Oracle судится с Google » Релиз Gnome 3.0 перенесен



Рубрику вел Евгений Крестников

Перед вами моя последняя новостная колонка для LXF – из-за высокой занятости в других проектах я передаю эстафету Ирине Матюшонок. Читатели журнала Ирину пока не знают, но в нашей

тематике она далеко не новичок. Мне же остается только поблагодарить своих читателей и коллег, а также предложить вашему вниманию свой «дембельский аккорд».

Linux в школах Ростова

Наш призыв поделиться опытом внедрения Linux в школах на общественных началах принес свои первые плоды.

В одном из предыдущих номеров мы упомянули инициативу ростовской группы пользователей Linux и попросили читателей сообщать интересные примеры некоммерческих инициатив. В редакцию LXF обратился один из активистов RLUG Сергей Ирьюпин – он рассказал о внедрении СПО в ростовских школах.

По его словам, идея помочь учебным заведениям витала в воздухе, но не встретила понимания со стороны школ и членов LUG до 2009 года. Затем ситуация в стране изменилась, но работать бесплатно захотели не все. В «школьном» проекте согласилось участвовать человек 10 (стабильно и регулярно – 3–4). Внедряли только ALT Linux: представитель LUG говорит, что работать с единой платформой удобнее.

О свободном ПО в ростовских школах особо не задумываются. Не все работники образования знают о возможной альтернативе коммерческим продуктам, хотя слово «Linux» на слуху у директоров и учителей. Относительно качества продукта – по мнению участников проекта, ALT Linux 4.0 имеет полный набор программ для преподавания информатики. Тут есть что показать и на занятиях по другим предметам. К тому же система хорошо оптимизирована под устаревшие школьные машины.

Основные технические сложности вызвала аппаратная совместимость: скажем, с интерактивной доской SmartBoard не было проблем, но попадались также

Promethean Activboard и Interwrite, пока не поддерживаемые в ALT Linux. Возникли трудности и с другой периферией.

Основную роль играет человеческий фактор: сотрудники школ плохо информированы об СПО и его преимуществах. Учителя инертны и не хотят изучать что-то новое: «Ну вот если нам открытым текстом скажут... если нас заставят... тогда конечно – куда же мы денемся, будем внедрять эту самую “линекс”».

Преподаватель школы №78, Оксана Михайловна, сказала LUG'овцам: «Мы всеми

силами будем стараться остаться на Windows. Будем искать деньги, покупать лицензии. Если нас будут заставлять внедрять этот Linux – ну, поставим второй системой, лишь бы отстали. А работать все равно будем на Windows». В школе №55 директор предоставил решение этого вопроса учителю информатики. А тот с порога заявил: «Зачем мне Linux? Я не собираюсь свои учебные планы переделывать».

Иногда собеседники (директора или завучи) не желали общаться с волонтерами. В школе №107 Сергею пришлось выслу-



«Проблема в том, что у нас люди потеряли веру во многое.»

» Сергей Ирьюпин: «Все больше учителей и учеников будет узнавать о свободном ПО».

шать от директора длинную тираду на тему «Windows хороший, а Linux плохой».

Есть и другие примеры. Директор школы №72, Оксана Юрьевна Чиплиева, обрадовалась помощникам: «Как здорово, что вы нам позвонили... мы сами пробовали-пробовали – ничего не получается; искали, кто может помочь – никого не нашли... вас прямо Бог послал». Инсталляция в этой школе была одной из самых удачных – все оборудование (доска, принтер, сканер) заработало сразу.

На западе к волонтерам привыкли. В США выстраиваются очереди желающих

GPL: Сумели отстоять

По решению суда, производитель бытовой техники должен заплатить разработчикам BusyBox около \$100 000.



➤ **Внутри этого устройства тоже может оказаться Linux. Причем — не всегда законно.**

бесплатно поработать в приютах для бездомных животных или сдать кровь в донорском пункте (донорам там тоже не платят). У нас на таких людей смотрят с опаской: «Для многих, да почти для всех, основная загадка — почему мы помогаем бесплатно? Проблема в том, что у нас в стране люди потеряли веру во многое. В том числе в идею бескорыстной, качественной, профессиональной помощи. Ведь не секрет, что и за деньги работают неохотно и некачественно. Ну хоть сектантами не считают — и слава Богу», говорит Сергей Ирюпин.

Тем не менее, результат обнадеживает: активисты ростовской LUG посетили 13 школ, где установили Linux в кабинетах информатики второй системой. Проведены беседы о преимуществах свободного ПО с учителями и директорами. В кабинете информатики школы №50 установлен только ALT Linux, и преподавание ведется на базе свободного ПО с октября прошлого года. В марте были организованы пятидневные, сертифицированные компанией ALT Linux, курсы для учителей Советского района города. Пять человек из «школьной» команды получили сертификаты «Специалист по внедрению ПСПО в системе образования», а Сергей Ирюпин (он, кстати, является членом ALT Linux Team) — еще и сертификат преподавателя.

Ростовские активисты будут продолжать обзванивать и посещать школы, беседовать с директорами и учителями, помогать им осваивать Linux, а также проводить бесплатные семинары и курсы для учителей. Сергей Ирюпин не сомневается в результатах: «Все больше учителей и учеников будет узнавать о свободном ПО. Все больше и больше школ будут использовать Linux. Будущее — за свободным ПО», сказал он нашему корреспонденту.

«**В**иновником торжества» стал набор утилит для встраиваемых систем BusyBox. Этот продукт — рекордсмен по числу подобных исков: его активно применяют производители устройств, не всегда готовые открыть исходные тексты своих продуктов, как требует лицензия.

Центр защиты свободного ПО (Software Freedom Law Center) сообщил об успешном завершении судебного процесса против Westinghouse Digital Electronics, включившей BusyBox в состав прошивок телевизоров высокой четкости и отказавшейся открыть сопутствующий исходный код. Компанию признали виновной в нарушении условий GPL, и она должна выплатить разработчикам программы \$90 000 компенсации и возместить судебные издержки в размере \$47 685. Кроме того, суд запретил дальнейшее распространение устройств, а уже выпущенные экземпляры будут переданы на благотворительные цели.

В США суд впервые признал неправоту нарушителя GPL, и это очень важный прецедент: ранее подобные иски заканчивались мировыми соглашениями с производителями. Пока правозащитники ликуют, хочу обратить внимание читателей на вопросы, оставшиеся за кадром. Почему суд круто обошелся с Westinghouse, но не запретил продавать устройства Samsung, Best Buy, JVC, Zухel или Western Digital? Иски против 13 производителей бытовой электроники были поданы в декабре прошлого года, но пострадал пока только один ответчик. Не просматривается ли тут связь с тем, что Westinghouse — фактически компания-банкрот, вынужденная распродавать свои активы, чтобы вернуть долги кредиторам? Взять с нее нечего (сомневаюсь, что разработчики BusyBox смогут получить свои \$90 000), и договариваться с таким оппонентом правозащитникам нет никакого смысла. Зато на роль кандидата для показательной порки Westinghouse подходит идеально — компании уже нечего терять, кроме своих телевизоров. Иное дело — серьезные игроки: Samsung, Comtrend, Dobbs-Stanford и GCI уже согласились с позицией истца, и судебное разбирательство с ними вскоре будет прекращено. Вероятно, ответчики пойдут на мировую, выплатят зную сумму разработчикам и их защитникам, а также выполнят ряд требований в части соблюдения условий лицензии.

У небогатых разработчиков свободного продукта нашлись защитники, способные на равных поспорить с акулами бизнеса. Неплохо! Но на чьи средства идет борьба? Почему серьезные (и не бедные) американские юристы тратят немалые суммы на судебные процессы? Торжество справедливости — удовольствие не дешевое: издержки на разбирательство всего с одним банкротом превысили \$47 000, причем перспективы получить с компании хоть какие-то деньги эфемерны. А ведь только за один «заход» было подано 13 исков! Да еще против солидных компаний, судиться с которыми — дело куда более затратное. Такова традиция стран победившей демократии: пусть ты кругом прав — денег на спор с «богатым дядей» тебе не хватит. И вдруг — хватило. Интересно, правда?

Я не обвиняю этих людей вговоре с корпорациями, оплаченным участии в их конкурентных войнах и прочих грехах. Этим (по слухам) занимаются защитники природы. Там ведь борются за чистоту окружающей среды, а здесь среды исключительно программные, и подобная аналогия не совсем уместна. Полагаю, читателям стоит самим поразмыслить над этой проблемой.

Что касается ситуации в России: представители контролирующих органов знают, что такое СПО, и факт его использования уже никого не удивляет. Споры о юридическом статусе GPL продолжаются до сих пор, а вопрос о возможности принуждения российских производителей к соблюдению ее условий остается открытым. Отвечать на него мы будем в следующих номерах журнала.

Я же обратился в ГНУ/Линуксцентр, где на базе популярного встраиваемого дистрибутива OpenWRT разрабатывают ПО для маршрутизаторов. Альтернативная прошивка позволяет их подключать к мобильным сетям через USB-модем. В компании не видят смысла в новой ветке открытого продукта, но готовы по запросу предоставить код своих модификаций любому желающему. Устройства сопровождаются письменным предложением, что третья сторона может получить исходные коды у разработчика ПО — в компании ГНУ/Линуксцентр. Впрочем, на вопрос о возможности принуждения «нарушителей» мне здесь не ответили — в Линуксцентре GPL соблюдают, и эта проблема компанию не затрагивает.

Oracle vs Google: Битва гигантов

Пока одни подают иски о нарушении патентных прав на технологии Java, другие начинают ответвление OpenSolaris.

Самое время вернуться к судьбе разработок Sun, тем более, что Oracle дала мощный повод – в опубликованном недавно пресс-релизе корпорация обвиняет Google в нарушении связанных с Java патентов. Поисковый гигант пока не отреагировал на этот выпад, хотя к моменту выхода журнала ситуация должна проясниться. Пока сторонники открытых технологий высказываются в духе «а мы предупреждали», а аналитики строят версии, рассмотрим причины конфликта.

Зачем Oracle нужна эта война? Авраам Линкольн однажды сказал: «И патентное, и авторское право нужны для поддержки пламени гения топливом интереса». Патенты давно переросли заложенные в них изначально функции и стали орудием конкурентной борьбы корпораций. По словам одного из ведущих российских аналитиков, директора компании Elashkin Research Михаила Елашкина, бизнес-модель Oracle всегда предполагала первичность данных. Корпорация обеспечивает управление ими через свои СУБД, но если данные хранятся на серверах Google – зачем тогда Oracle нужен? Михаил выделил здесь два основных

момента. Прежде всего, корпорация пытается удержать контроль над технологией Java, ради которой во многом и была куплена Sun. На втором плане – личность Ларри Эллисона и его отношение к тому, кто должен быть на вершине: «Всемирно известна его “нелюбовь” к Биллу Гейтсу. Наивно думать, что она строится на личных отношениях – речь о власти и деньгах. Сейчас Гейтс вышел из игры, и Microsoft перестает быть главным раздражителем на рынке. А вот Google, безусловно, стремится стать единственным богом. Вот Брину с компанией и поставили флажки. Показали, что на рынке есть зубастые и кроме них», говорит эксперт корреспонденту «Компьютеры-онлайн».

Получается, что Oracle по-хозяйски обходит бывшие владения Sun и метит территорию. По словам Михаила Елашкина, «она купила некоторые активы Sun, которыми те не могли распорядиться наилучшим образом – не хватало денег, харизмы, влияния... а Oracle добавляет их себе в обойму и стреляет очередями». Тем не менее, патенты нельзя считать только поводом для драки – на первоначальном этапе Oracle

защищает стандарты, которые затем использует, чтобы придержать Google.

Говоря о развитии ситуации, эксперт не берется делать глобальных прогнозов: «И для Oracle, и для Google Java и Android – не основные продукты, обеспечивающие прибыль. В соответствии с правилами корпоративно-патентных войн, Google выкатит Oracle встречный иск о нарушении патентов, и через лет пять работы сотен очень дорогих юристов компании мягко договорятся о сумме компенсации», считает он.

Свободные продукты и технологии не являются мишенью атаки – здесь мы имеем дело с очередной корпоративной «разборкой». Но Java очень важна для отрасли, а потому инициатива Oracle неизбежно скажется на свободном ПО, вот только каким образом – пока неизвестно. Не стоит сейчас говорить и о судьбе прочих проектов Sun (в первую очередь, OpenSolaris и его новым ответвлением, Illumos): ситуация слишком нестабильна, и любые выводы устареют еще до появления журнала на прилавках. Но мы непременно вернемся к этой теме в следующих выпусках.

Ubuntu и Gnome: Скандал в благородном семействе

Canonical опять обвинили в том, что она вносит в свободные проекты слишком мало кода.

С 26 по 30 июля в Гааге прошла конференция пользователей и разработчиков Gnome GUADEC. Были озвучены планы по переносу выхода третьей версии на март следующего года и показаны новые макеты GNOME Shell.

Джон Мак-Канн [Jon McCann] из Red Hat сказав, что сейчас Gnome – по сути, набор отдельных продуктов, предложил сделать среду более целостной. О собственной ОС речи не идет: хотя название Gnome OS прозвучало, это следует понимать как предложение сделать из Gnome платформу для разработчиков, включающую различное промежуточное ПО (т.н. Middleware OS).

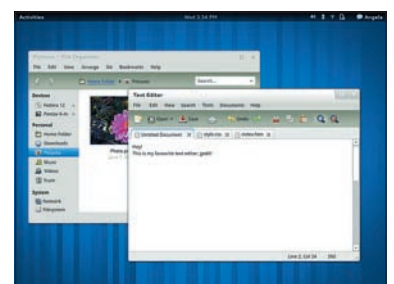
Все очень логично – Gnome является основной средой для коммерческих дист-

рибутивов, и корпорации боятся повторения ситуации с KDE 4. Им не нужен сырой продукт, который придется доводить до ума несколько лет – отсюда перенос сроков выпуска и инициатива г-на Мак-Канна.

Не обошлось и без курьезов: согласно опубликованным результатам опроса разработчиков, вклад Canonical в создание Gnome на порядок меньше, чем у Novell или RedHat. В ответ на критику, Джоно Бэкон [Jonno Bacon] сообщил, что многие проекты Canonical не включают в основную ветку Gnome, и найти их можно только на Launchpad. Компания никогда и не утверждала, что вносит в Gnome больше кода, чем Red Hat или Novell. Офици-

альной реакции Canonical не последовало, если не считать таковой запись о вреде трибализма в личном блоге Марка Шаттлворта. Основатель Canonical критикует людей, считающих других некомпетентными по умолчанию. **LXF**

➤ Gnome 3.0 должен стать гораздо более целостным, чем его предшественники.



Новинки программного и аппаратного обеспечения в описании наших экспертов



АЛЕКСЕЙ ФЕДОРЧУК
Его слабости — mass storage, разметка диска и файловые системы.

KDE 3: Реанимация или гальванизация?

Для многих старых пользователей KDE, начинавших ещё с 1-й версии, выход KDE 4.0 был шоком. И дело не в недоработках — для первого представителя новой ветки они естественны; но это было не то KDE, которое они знали и любили. И потому с тоской вспоминали доброе старое время — даже два корректирующих релиза KDE 3.5.x от этого не спасали: было очевидно, что дни 3-й ветки сочтены.

И вот недавно Тимоти Пирсон [Timothy Pearson] решил исправить положение коренным образом, начав проект Trinity KDE. В двух словах, это дальнейшее развитие 3-й ветки с некоторыми заимствованиями из 4-й — теми, что не идут вразрез с исконной идеологией KDE. С первыми результатами его работы можно ознакомиться посредством Kubuntu 10.04, где они объединены под именем KDE3/Trinity 3.5.11.

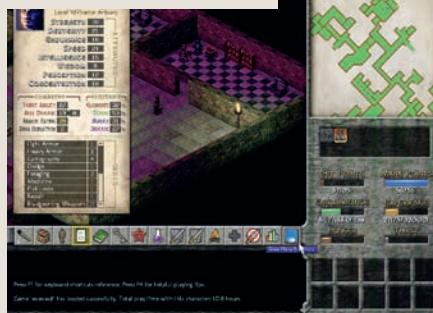
Увы, успех этого безнадёжного предприятия видится сомнительным. Кроме очевидных проблем с версиями библиотек Qt и привлечением разработчиков, главным камнем преткновения будет пользовательская база. За истекшие два с половиной года KDE 3 растеряло изрядную часть своих некогда преданных поклонников: одни, скрепя сердце, погрузились в мир плазмойдов 4-й ветки, другие же мигрировали на иные рабочие столы. Новые же пользователи воспринимают KDE 4 как данность. И проект Тимоти может оказаться просто не востребованным широкими народными массами. Тем не менее, я очень хотел бы пожелать ему удачи.

alv@posix.ru

Сегодня мы рассматриваем:

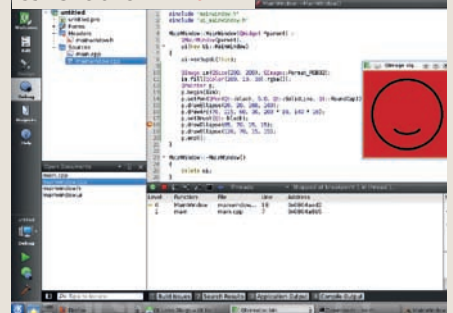
- | | | | |
|--|-----------|--|-----------|
| Eschalon: Book II | 8 | Qt Creator 2 | 11 |
| Это сиквел к RPG, которая весьма понравилась нам в прошлом году. Алекс провел 20 часов, познавая новый мир, но сказать по правде, мы сомневаемся, что он сам возьмется за новую серию. | | Причиной того, что новый KOffice оказался не лучше старого, может быть использование разработчиками KDevelop 4. Переход на Qt Creator 2 мог бы кое-что принести. | |
| Eclipse Helios | 9 | KOffice 2.2 | 12 |
| Этот релиз — неплохой повод увязать между собой одну из самых популярных в мире интегрированных сред разработки и необходимые в Linux инструменты разработчика. | | Энди Ченел немало натерпелся с предыдущей версией этого офисного пакета, но все же нашел в себе силы взглянуть на новую. Оправдала ли команда KDE его доверие? | |
| WordPress 3 | 10 | Книжные новинки | 13 |
| WordPress 2.x вышел вечность назад, и вот самая популярная платформа для блогов увидела крупное обновление. Будут ли работать с ним ваши раскладки и расширения? | | Две книги для разработчиков: настройте MySQL на максимальную производительность и разберитесь с Django. | |

Eschalon: Book II с. 8



» Вернитесь в старый добрый мир изометрических проекций, языческой символики и плохой гигиены.

Qt Creator 2 с. 11



» Каким бы хорошим ни был Qt Creator, даже ему слабо сделать из Грэма классного программиста.

Наш вердикт: Пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибальной шкале (0 — низшая оценка, 10 — высшая). Как правило, учитываются функциональность, производительность, простота использования и цена, а для бесплатных программ — еще и документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.



Выдающиеся решения могут получить престижную награду «Топ Stuff». Номинантами становятся лучшие из лучших — просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но если разработчики рекомендуют Autopackage, мы следуем этому совету.

LINUX FORMAT Вердикт

Google Earth

Разработчики: Google

Сайт: <http://earth.google.com>

Цена: Бесплатно по закрытой лицензии

Функциональность 10/10

Производительность 9/10

Простота использования 9/10

Оправданность цены 9/10

» Если весь мир — сцена, то Google Earth — театр. Простая в использовании, захватывающая и ободряющая практическая программа.

Рейтинг 9/10

Eschalon: Book 2

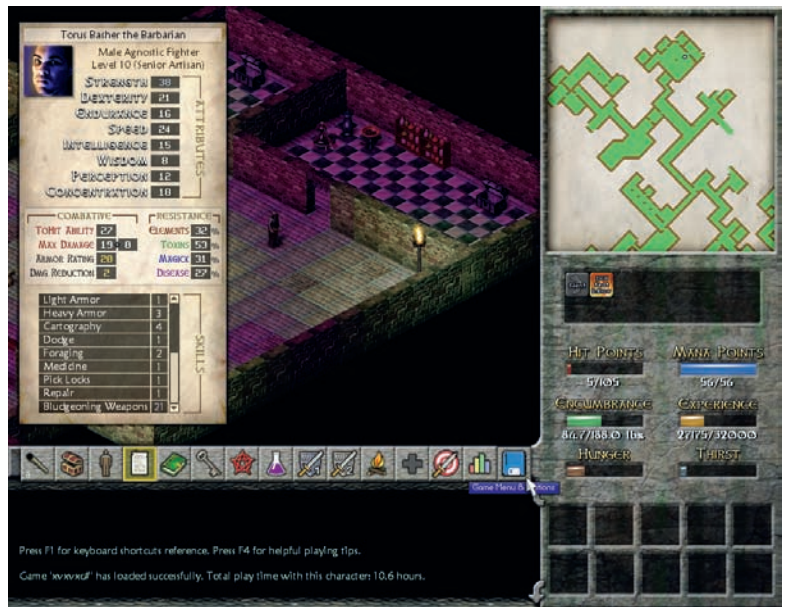
Алекс Кокс – один из тех чудиков, что готовы умереть с голоду, воюя с крысами в подземелье. Превосходный обозреватель для этой игры.

Вкратце

» Игра изометрического формата в духе классики 80-х. Мечи, магия и кучадохлых крыс. Напоминает *Sacred Gold* или *Nethack*.

Тампи, грозный воин Эвердейла. Загрызен крысой: 17 раз подряд лупил палицей – мимо! Пол Дэниелс, маг из Порт-Куудада. Умел извлекать шаровые молнии из воздуха, но умудрился погибнуть от жестокого обезвоживания прямо на берегу озера. Мистер Сникс, мошенник из Вольфенвуда, перестал существовать при попытке надеть брюки. И всё это в один день.

Eschalon Book II выросла на том же ком-посте, что и предыдущая серия. Это смесь почвы от игр *Ultima* и нашинкованных книжек AD&D под перебоем из напыщенных фэнтези-романов. Квесты по большей части стандартны: пойдти туда, убей того – и всё это в сопровождении библейски-нудных поучений. Впочем, вполне в духе игр типа *Ultima*.



» Это RPG: ударное оружие под инвентарными номерами, жуткое тёмное подземелье и 5 очков на дорожку.

«Приятно, когда RPG оставляет вам свободу выбора.»

Поскольку это сиквел ролевой игры (RPG), действие начинается предсказуемо. Вы потеряли память вместе с бойцовскими и магическими навыками, нажитыми непосильным трудом в предыдущей игре. Тут следует долгий период смертельных поединков, истребления монстров и игрового становления, с одной целью: узнать, кто же вы такой и что, чёрт возьми, происходит.

Нарращивать мускулатуру персонажа далеко не просто. «Баланс благосклон-

ности» *Eschalon* выглядит чертовой учительшей на фоне более снисходительных современных RPG. Например, хранителям квестов нет никакого дела до того, как вы справитесь со своей миссией – а сила монстров не согласуется с вашими возможностями. Кончится тем, что вас пошлют в командировку без шансов на выживание. Поэтому почаще сохраняйте игру (тем более что автосохранение отсутствует).

Ретушь для квеста

Начало игры затруднено нехваткой ресурсов. Здоровье, даже без битв, восстанавливается мучительно долго: на ранней стадии лекарства неэффективны, разорительно дороги и попадают редко.

Но можно поломать игровые правила, создав нового персонажа: например, отключить нудную систему заготовки пищи или отменить сохранение при приближении к врагу и при болезнях. Приятно, когда RPG оставляет вам свободу выбора.

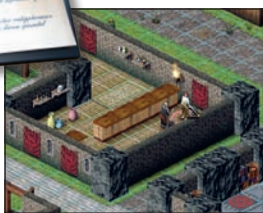
Эта версия *Book II* – сравнительно «молодая» сборка для Linux, поэтому встречаются довольно странные ошибки. Самый озадачивающий трюк – «штаны смерти»: стоит приобрести брюки, как игра завершается аварийно. Но мы твердо надеемся, что когда вы доберётесь до нее, эта

и другие ошибки будут исправлены: форум *Basilisk* весьма активен и хорошо поддерживается разработчиком.

Несмотря на садистский замысел *Eschalon Book II*, мы с удовольствием рекомендуем её как образчик жанра. Игровой мир не особо разнообразен, но обширен. Хотя сюжет и безыскусный, при таком разнообразии игровых персонажей парочку кругов вы пройдёте с удовольствием. Однако если вы не знакомы с RPG старого типа, советуем сперва попробовать демо. **LXF**



Свойства навскидку



Гадкие торгаша

Эти парни завязали цену? Убить их! Только вот очередную дозу лекарства взять будет нелегко.



Непогода

В *Eschalon* часто дождит. И это не просто графика: биться под дождём гораздо труднее.

LINUX FORMAT **Вердикт**

Eschalon Book II

Разработчик: Basilisk Games
 Сайт: www.basiliskgames.com
 Цена: \$25

Сюжет	6/10
Графика	5/10
Увлекательность	8/10
Оправданность цены	7/10

» Достойное продолжение серии *Eschalon* по мотивам *Ultima*. Не для слабонервных.

Рейтинг 7/10

Eclipse 3.6

Ник Вейч представит в новом свете известную среду разработки – монументальную и громоздкую, но зато универсальную.

Вкратце

» Универсальная среда разработки с открытым исходным кодом, «заточенная» под Java. См. также *NetBeans*.

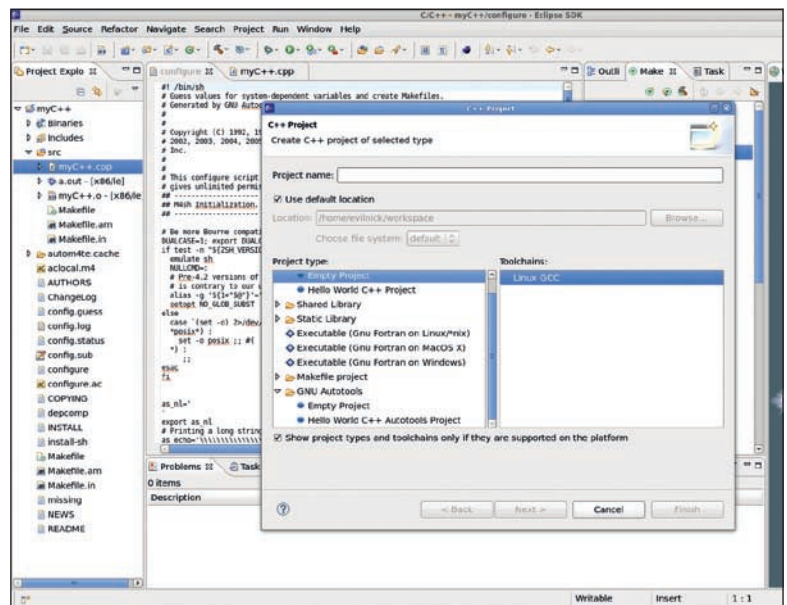
Ежегодно в конце июня происходят некоторые традиционные события. Одно из них – выход новой версии *Eclipse*, всеобъемлющей среды разработки с открытым исходным кодом, которую IBM отдала миру в 2001 году. После появления в 2004 году серии 3.x, каждый год выходит обновление – результат целенаправленного и планомерного развития.

Это позволяет всем сопутствующим проектам концентрировать усилия на выпуске одной ежегодной версии и создаёт элемент предсказуемости (чем открытое ПО обычно не блещет). Кроме уверенности в дате выхода очередной версии, бизнес-пользователи получают надёжный цикл обновления.

Новая версия *Eclipse* получила несколько долгожданных новых функций – например, применение *Git* в качестве репозитория, усиленная поддержка JavaScript и других web-инструментов, новый клиент 'Marketplace' для поиска необходимых модулей и расширенная поддержка инструментов C/C++ на Linux.

«Очевидное решение для коммерческих проектов на Java.»

В секторе Java наиболее интересны проект *Web Tools Platform*, обновлённый до новейших спецификаций *Java Enterprise Edition*, а также развитие мобильного Java-



» С развитием поддержки Linux, *Eclipse* всерьёз намерена стать универсальной средой разработки. Здесь можно разрабатывать даже KDE-приложения!

проекта *Pulsar*. Web-приложения – и облачные, и мобильные – в *Eclipse* поддержаны неплохо, но на сей раз сделана попытка создать единую среду. Для этого добавлены официальная поддержка JavaScript, включая отладчик, и другие средства web-разработки.

Не только Java

Многие ошибочно полагают, что *Eclipse* – это среда разработки только для Java. Как таковые, расширения C/C++ существовали в *Eclipse* почти с самого начала, но не были особо популярны среди Linux-программистов.

Отчасти это объясняется тем, что проекты *Eclipse* трудно встраивать в инструментарий Linux-разработки. От одного *GCC* без средств формирования пакетов, *GNU Autotools*, *Valgrind* и других Linux-инструментов пользы мало. Но с прошлого года в проекте *Eclipse Linux Tools* началась работа над официальным модулем расширения для обеспечения недостающей функциональности, и сейчас мы наблюдаем пробу пера.

Конечно, поддерживаются и другие языки. Кроме PHP и Ruby, *Eclipse* работает с Fortran и многочисленными web-технологиями, включая JavaScript и XML. Поддержка Python осуществляется *PyDev*, модулем

от сторонних лиц (удобным для проектов Google App Engine).

При таком разнообразии Java-ресурсов и расширений собственного и стороннего производства, *Eclipse* становится очевидным решением для разработки коммерческих проектов на Java – да и для всех, кто программирует на Java. Очевиден упор на web-разработку, тогда как настольный Linux несколько обделён. Да, в *Eclipse* есть несколько полезных инструментов, но для всех языков кроме Java это не более чем базовая поддержка. **LXF**

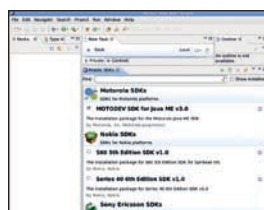


Свойства навскидку



Marketplace

Этот новый клиент позволяет запросто отыскать любой требуемый дополнительный модуль.



Мобильная разработка

Для мобильной Java-разработки можно установить *Pulsar* и как модуль расширения, и как отдельное приложение.

LINUX FORMAT Вердикт

Eclipse 3.6 (Helios)

Разработчик: The Eclipse Foundation
Сайт: www.eclipse.org
Лицензия: Eclipse Public Licence

Функциональность	9/10
Производительность	8/10
Простота использования	6/10
Документация	7/10

» Не совсем для разработки настольных приложений на C++, зато супер-выбор для любого Java-программиста.

Рейтинг 8/10

WordPress 3.0



Популярнейшее ПО для ведения блогов посягает на CMS. Энди Ченнел проверяет, не чрезмерны ли такие амбиции.

Вкратце

» Простое ПО для управления блогами, которое возомнило себя CMS. См. также Blogger, Drupal, Joomla.

Вероятно, WordPress на сегодняшний день – наиболее популярное ПО для ведения блогов. Версия 3.0 свидетельствует о нацеленности разработчиков на успех Joomla, Drupal и других CMS-брендов.

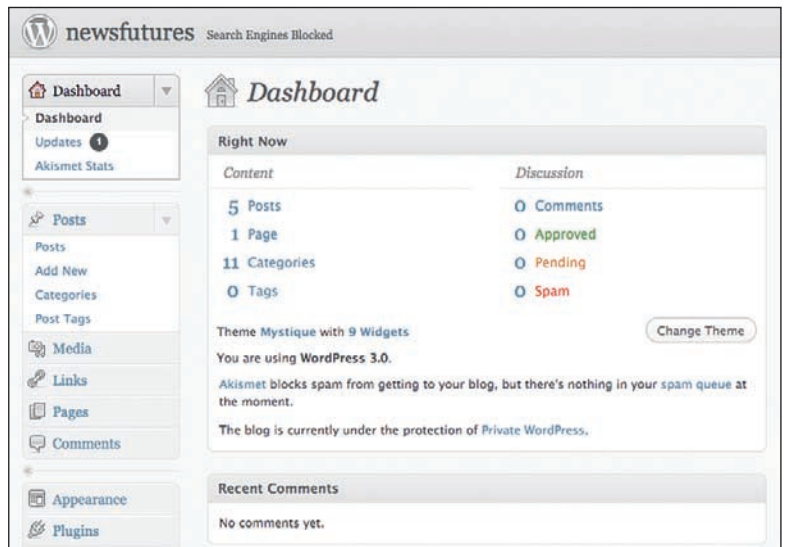
Если у вас уже есть сайт с работающим экземпляром WordPress, то для установки достаточно нажать ссылку Automatic Update на панели инструментов [Dashboard]. Новая установка немногим сложнее: если база данных и веб-сервер уже настроены, всё прочее займёт какую-то пару минут.

Панель инструментов, где настраиваются блоги и посты, стала элегантнее и выдает гораздо больше информации по одному щелчку. По ощущению, система очень динамична и полностью сосредоточена на том, чтобы предельно упростить и облегчить процесс ведения блогов.

Исчерпывающая система меню с поддержкой перетаскивания и переименования позволяет создавать весьма сложные сайты с динамичным, многообразным

«Панель инструментов более информативна, чем в прежней версии.»

представлением контента – что необычно для блогосферы. Управлять системой меню проще, чем в Drupal: можно встроить навигацию в визуальную тему или реализовать ее в виде виджетов на web-страницах. Представления и работа с запросами



» Новой панелью удобнее пользоваться, а документация превосходная.

не идут ни в какое сравнение с возможностями Drupal, но и здесь у WordPress мощный потенциал развития.

Модули и виджеты

Силами сообщества WordPress разработаны горы подключаемых модулей для расширения функциональности. По-быстрому добавить их – и можно делать почти любую работу из арсенала более сложных CMS, включая (с помощью модуля BuddyPress) создание социальных сетей типа Facebook.

Благодаря другому достижению новой версии появилась возможность использования BuddyPress. Прежде эта функция работала только с дистрибутивом WordPressMU, а теперь удачно встроена в основную систему.

Версия 3.0 укомплектована новой темой с удобными меню и множеством вариантов виджетов, колонок, фоновых рисунков и заголовков. Всё это хорошо продумано и надёжно действует, а рациональная структура и развёрнутые комментарии упрощают работу с CSS. По сути, тема TwentyTen в сочетании со встроенным в WordPress редактором CSS могут служить вполне приличным руководством по редактированию CSS для интересующихся web-дизайном.

Оснащённостью WordPress 3.0 уступает более крупным собратьям (Drupal

или Elgg), зато он удобнее в управлении и обращении с контентом. Это особенно заметно при обновлении ядра, графических тем и подключаемых модулей. Некоторые функции – например, типы постов (с помощью которых администратор может создавать пресловутые категории) – ещё не обзавелись графическим интерфейсом, но редактирование для них требуется минимальное. В целом, это замечательный экземпляр ПО – а кроме того, отличное учебное пособие для тех, кто интересуется механикой администрирования блогов или системами управления контентом. **ixbt**



Свойства навскидку



TwentyTen

Новая стандартная тема «подчиняется» и меню, и виджетам – и содержит массу настраиваемых параметров.



Вскройте CSS

С редактором CSS можно осваивать web-разработку: устройство темы больше не тайна.

LINUX FORMAT Вердикт

WordPress

Разработчик: Команда WordPress
 Сайт: <http://wordpress.org>
 Лицензия: GPL

Удобство использования	9/10
Функциональность	8/10
Производительность	9/10
Документация	10/10

» Идеальное введение в ПО для блогов и CMS. Нам очень понравилось – хотим, чтобы понравилось и вам.

Рейтинг 9/10

Qt Creator 2.0

Через месяц после выхода *KDevelop 4* Грэм Моррисон исследует его заклятого соперника. Кто победит? Посмотрим...

Вкратце

» Среда разработки для программистов C++, использующих систему Qt. См. также *Eclipse*, *KDevelop 4* или старенький *Kate*.

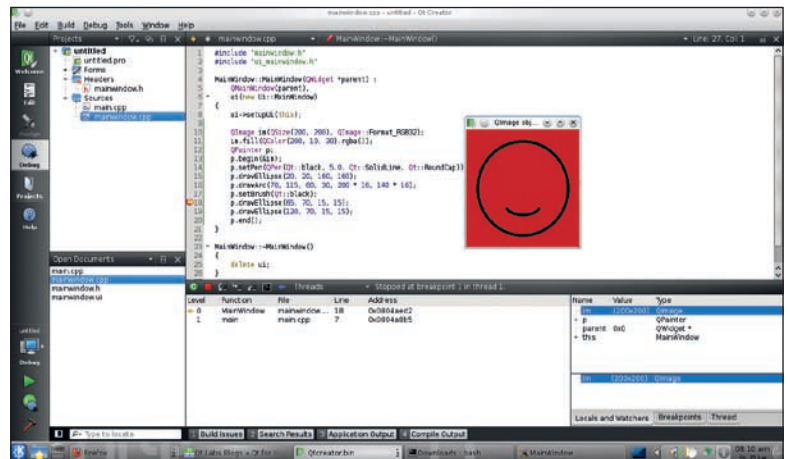
По горячим следам *KDevelop 4*, на несколько месяцев раньше, чем мы предполагали, Nokia выпустила версию 2 новаторской среды разработки для Qt. Ожидалось, что она станет частью Qt 4.7, которая пока ещё в стадии альфа-тестирования. Вместо этого Nokia включила новую версию в *Qt SDK 1.0*, слегка доработанный вариант Qt с эмулятором устройств и поддержкой MeeGo. Новинку можно также получить в составе Qt 4.6.3 или как отдельный проект.

Установка проходит гладко: берете объёмистый двоичный файл, делаете его исполняемым и запускаете из командной строки. Если вы производите это из своей учётной записи, то все необходимые файлы устанавливаются автоматически, включая значок *Creator* на рабочем столе — нажав его, вы запустите IDE. Не надо даже разыскивать библиотеки разработчика Qt: всё включено по умолчанию. Например,

«Теперь Qt Creator значительно глубже «понимает» Qt-код.»

пользователя Ubuntu, чтобы начать программирование, достаточно установить пакет **build-essentials**, а следом *Qt*.

По сравнению с версией 1.0, которая теперь воспринимается скорее как прототип, чем как основной релиз, изменилось многое. Например, новый мастер запуска — один из лучших в своём роде и впервые реально помогает новичкам. Прежде чем перейти к официальной документации Qt,



» Возможности отладки в *Qt Creator* сильно расширены — теперь можно отладить даже изображение с выводом во временное окно.

можно изучить примеры, касающиеся интерфейса, затем ознакомиться с несколькими простыми шаблонами. Всё это делается через встроенную справочную систему, прямо из главного окна.

Арсенал ПО

Creator 2.0 без запинки загрузил все наши проекты 1.x, и теперь можно работать с несколькими проектами одновременно, переключаясь между их кодами с помощью левой панели. Единственное неудобство — не поймешь, который из проектов будет собран и запущен. Выручает новый инструмент выбора цели, расположенный прямо под значком воспроизведения. С его помощью можно выбирать проекты для сборки (их названия отображаются под пиктограммой), а также переключаться между отладочной и окончательной сборками и их архитектурами.

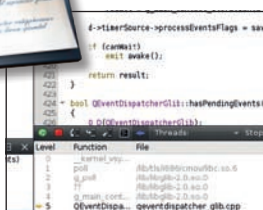
Подсветка синтаксиса была неплохой ещё в прежней версии и доработана ещё. Скажем, теперь названия символов выделяются при использовании их в строке. Эту функцию можно применять для рефакторинга. Щёлкните на символе правой кнопкой мыши, выберите команду **Rename Symbol Under Cursor** [Переименовать символ под курсором] и, по мере ввода нового названия, все остальные вхождения обновятся в режиме реального времени. Функция распространяется даже на другие файлы — например, заголовки. А если объект «потерялся», для его поиска можно использовать регулярные выражения.

Ещё одно направление стремительного развития *Creator* — это отладка, которая теперь полностью совместима с *GDB 7* и оснащена реверсом для исследования выполнения приложения по шагам обратно. Что ещё более важно для Qt-программистов — *Creator* значительно глубже «понимает» Qt-код: серия Python-тестеров автоматически проверяет некоторые стандартные и специфичные для Qt типы данных.

Можно назвать ряд неудачных дизайнерских решений (так, окно **Watch** появляется только в режиме отладки), но в целом *Creator 2.0* — крупный шаг вперёд. Добавьте кроссплатформенную совместимость и поддержку MeeGo, и *Creator* приобретает убойную силу. Ждём *Qt 4.7*. **LXF**

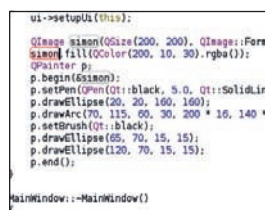


Свойства навскидку



Поддержка GDB 7

Обратная отладка и превосходная поддержка Qt-объектов ускоряют поиск ошибок.



Рефакторинг

Измените название одного символа, и оно изменится во всех его вхождениях.

LINUX FORMAT **Вердикт**

Qt Creator 2.0

Разработчик: Nokia
 Сайт: <http://qt.nokia.com>
 Цена: Бесплатно на условиях LGPL

Функциональность	9/10
Производительность	9/10
Простота использования	7/10
Документация	10/10

» Лучшая среда разработки для всех, кому нравится Qt. Кому не нравится, советуем попробовать снова.

Рейтинг 9/10

KOffice 2.2



KOffice постоянно совершенствуется, но KDE, похоже, не до него. Есть ли будущее у этого офисного пакет как такового? Рассуждает Энди Ченнел.

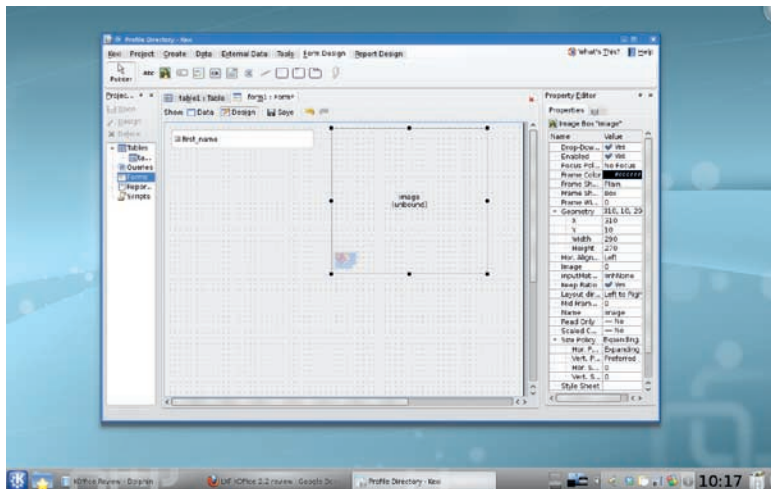
Вкратце

» Универсальный офисный пакет для KDE. См. также OpenOffice.org, MS Office/Wine и Google Docs.

За последние 12 лет KOffice вырос и масштабами, и амбициями. Были удачные и неудачные шаги; были и громкие заявки без малейшего шанса на реализацию.

Версия 2.2 в составе KWord, KPresent, KSpread, KPlato (менеджер проектов), Krita (редактор изображений) и базы данных Kexi вошла в новый мир. Настольные приложения ныне серьезно страдают от «облачной» конкуренции. Монополия Microsoft шатается, а мобильная платформа входит в жизнь.

Интерфейс KOffice тяготеет к гибкой системе палитр Adobe, а не к лентам в стиле Microsoft, и это гораздо разумнее, чем горизонтальные панели соперников. По сравнению с предыдущей версией интерфейс во многом улучшен: теперь органы управления не уползают за экран, а приложения объединяет общий стиль.



» Kexi снова в KOffice: что за офисный комплект без базы данных?

«Компоненты KOffice сравнимы с собратьями из OpenOffice.org.»

Всё запускается и действует стремительно, хотя сложные диалоги (например, менеджера стилей) открываются по несколько секунд даже на мощных машинах. В работе приложения вполне сравнимы с эквивалентами из OpenOffice.org и побивают MS Word в большинстве задач по редактированию и созданию документов.

Импорт документов, созданных другими приложениями, не всегда успешен — но и здесь по сравнению с прежними версиями заметен прогресс. KWord справился

даже с точным импортом простого файла .docx, созданного в Word 2010: компоновка и стили уцелели полностью. Правда, при редактировании документов MS Word со сложным стилевым оформлением возникали экранные помехи, но в процессе нормальной работы они самоликвидировались.

Обмен файлами

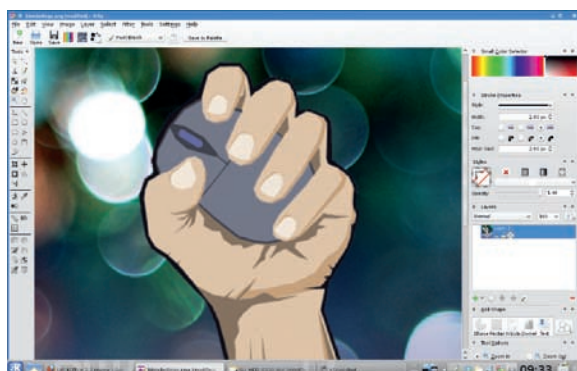
Наверное, самая надёжная часть KOffice в отношении совместимости — редактор изображений Krita, который работает со стандартными форматами типа JPEG и GIF (хотя с новейшими творениями Photoshop он не справился). Импорт PDF действует отменно: даже самые сложные страницы воспроизводятся быстрее, чем в Photoshop — правда, закладки, гиперссылки и прочие интерактивные элементы остаются «за кадром».

Возвращение Kexi пришлось весьма кстати: для работ с небольшими базами данных это просто находка. Элементы конструктора не так сложны, как у OOo Base, но большего и не надо, если вы хотите наскоро создать базу данных, соорудить работоспособную форму и тут же приступить к заполнению. Корешки вкладок для быстрого вызова конструктора, редактора отчётов и прочих инструментов обеспечивают интуитивное управление. Не помешало бы добавить шарма (пользователи ценят красоту), но как отправной пункт Kexi очень даже на высоте.

Новая версия KOffice значительно прибавила, но достаточно ли этого сейчас, в эпоху роста онлайн-офисов — большой вопрос. Социальная стратегия KDE построена таким образом, что предстоящий переход на «облачные» решения — Google Docs, MS Live Office или что-там-удумает Facebook — не будет проблематичным. С поддачи Nokia, ныне поддерживающей проект как часть своей мобильной технологии, разработчики сосредоточились на закреплении достигнутого, а не на модных штучках — и онлайн-хранилище неплохо вписывается в программу.

Работа над KOffice продолжается, и не всё может быть идеальным. Пробыйте его в работе — и помогите в доводке. LXF

» Krita обладает достойным набором инструментов, а импорт PDF превосходен.



LINUX FORMAT Вердикт

KOffice 2.2

Разработчик: KOffice Team
Сайт: www.koffice.org
Лицензия: GPL

Функциональность	7/10
Производительность	6/10
Простота использования	7/10
Документация	5/10

» Обновлённый пакет, укрепляя существующие позиции, одним глазком смотрит в будущее.

Рейтинг **6/10**

MySQL. Оптимизация производительности

Юлия Синицына пять лет оптимизировала коммерческие СУБД и выяснила, что *MySQL* в этом смысле не хуже.

Всё тормозит? Web-приложение не успевает выдать ни страницы? Босс не дождался вывода отчета? На сверхнадёжном RAID-массиве с репликацией кто-то удалил базу данных? Нет, эта книга не для вас. Вам нужно было прочесть её гораздо раньше.

Кажется, что, одолев её 800+ страниц, читатель сможет манием руки заставить работать быстрее любой *MySQL*-сервер. Это не совсем так: внимательно прочтя их, вы будете способны на большее.

Первые главы книги вполне предсказуемы: это архитектура *MySQL*, технология тестирования и профилирования, оптимизация схемы, основы и стратегии индексирования, оптимизация запросов. Затем авторы переходят на более высокие уровни: настройка параметров сервера в целом, ОС и оборудование, репликация, резервирование, масштабирование, безопасность и оптимизация архитектуры приложения.

Несмотря на обилие авторов (их шесть), книга удивительно цельная. Это не справочник, но и не сборник рецептов –

ни одного пошагового руководства или даже категоричного утверждения «всегда делайте только так» здесь нет. Она сошла бы за хороший, глубокий и всесторонний учебник по разработке и поддержке высокопроизводительных баз данных на *MySQL*, будь в ней упражнения и домашние задания. Но нет и их – остаются чистый опыт и знания специалистов с многолетней практикой. Не давая универсальных решений, они задают вам направление поиска своих: описывают проторенные пути, обязательно указывают на подводные камни и не чураются недокументированных возможностей и «хаков».

Книга предполагает «средний» уровень подготовки. Браться за нее, не зная SQL и основ устройства ОС, не стоит. Но лёгкий и живой язык не отпугивает от текста, и здесь явно найдется кое-что и для гуру.

Минусы книги неизбежны для детища многих авторов – стиль и (что хуже) представление о читателе местами заметно



» Не ждите, что книга решит проблемы за вас – но она научит, как сделать это самостоятельно.

различаются: то походя упомянуто сложное явление, то разжёвываются тривиальные вещи. Не всегда изложение сохраняет стройность, и не везде за нагромождением полезных фактов сразу видна структура.

Но книга стоит трудов. Сотни серверов, тысячи подключений, миллионы транзакций, миллиарды строк... По плечу – всё.

LINUX
FORMAT **Вердикт**

MySQL. Оптимизация производительности, 2-е издание

Авторы: Б. Шварц, П. Зайцев, В. Ткаченко, Д. Заводны, А. Ленц, Д. Боллинг
Издательство: Символ-Плюс/O'Reilly
ISBN: 978-5-93286-153-0
Цена: 850 руб.
Объем: 816 стр.

» Море полезной информации для тех, кто хочет научиться выжимать из *MySQL* максимум.

Рейтинг 10/10

Django. Подробное руководство

Евгений Крестников подзаброешил свои web-проекты: может быть, эта книга будет ему стимулом?

D*jango* представляет собой набор шаблонов web-разработки высокого уровня абстракции. Он упрощает создание web-приложений на языке Python и позволяет минимизировать затраты на организацию сложных web-проектов. И мы не раз писали о нем в **LXF**.

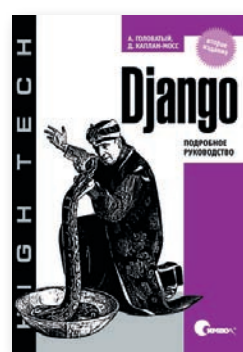
Для начала представим авторов: Андриан Головатый [Adrian Holovaty] и Джейкоб Каплан-Мосс [Jacob Kaplan-Moss] – ведущие разработчики и Великодушные Жизненные Диктаторы *Django*. То есть книгу писали профессионалы, и один этот факт заставляет отнестись к ней с уважением. Перед нами второе издание руководства (первое неформально называют «Django Book»), учитывающее особенности *Django 1.1* – предыдущая книга, к сожалению, давно устарела.

Материал разбит на четыре больших раздела, первый из которых содержит начальные сведения об инструментарии. Второй посвящен вопросам профессионального использования *Django*, а в третьем собраны описания возможностей, не вошедших в первые два раздела (такие как

генерация документов PDF или интернационализация). В разделе приложений читатели смогут найти различные справочники (по моделям, API и т. д.). Текст хорошо структурирован и снабжен подробным оглавлением, алфавитным указателем, а также примерами и иллюстрациями.

Изложение рассчитано на людей, не знакомых с *Django*, но имеющих базовые представления о программировании. Подробный учебник по Python авторы в книгу не включили; и хотя они при необходимости поясняют особенности языка, читателю, собравшемуся осваивать Python и *Django* вместе, придется обзавестись второй книгой.

Я долго колебался, выбирая, к какому классу отнести это издание. Начинающие разработчики могут использовать его как учебник, а более опытные – как справочник (тем более, что здесь обсуждаются узкоспециализированные возможности). Материал изложен в простой и доступной форме, а качество перевода не раз-



» Мы хотели бы придраться к чему-нибудь, но пока не нашли, к чему.

очаровывает. Бумага достаточно плотная, а мягкий переплет выглядит надежным и не должен развалиться от частого употребления. Кажется, в данном случае придраться не к чему, и второе издание книги «Django. Подробное руководство» получит высший балл в рейтинге **LXF**. Правда, надо признать, что я все же не зарабатываю на жизнь web-разработкой. **LXF**

LINUX
FORMAT **Вердикт**

Django. Подробное руководство

Авторы: А. Головатый, Дж. Каплан-Мосс
Издательство: Символ-Плюс/Apress
ISBN: 978-5-93286-187-5
Цена: 590 руб.
Объем: 560 стр.

» Хорошее (а главное – актуальное) руководство по *Django*. Учебник для начинающих и справочник для профи.

Рейтинг 10/10

Сравнение

» Каждый месяц мы сравниваем тысячи программ — а вы можете отдыхать!

Клиенты BitTorrent

Дистрибутивы сами себя не качают, и Ник Вейч занялся торрентами.



Про наш тест...

Мы тестировали клиенты на настольном ПК Core 2 Duo 2,5 ГГц с ОЗУ 4 ГБ под Fedora 13. Кроме штатного применения, провели основной тест на скорость, потребление ресурсов центрального процессора и памяти на четырех проверочных торрент-файлах размером в 256 МБ каждый. Клиенты ограничивались одновременным скачиванием трех файлов. Трекер был запущен на той же машине, для снятия зависимости от ширины канала, а торренты раздавались тремя компьютерами локальной сети 10/100 Мбит с использованием разнообразных клиентов (*Transmission*, *uTorrent*, *Vuze*).

Активное потребление памяти или ядер центрального процессора часто бывает намеренным, ради увеличения скорости скачивания. Время скачивания усреднялось по трем закачкам, но приводится только для справки.

Наша подборка

Deluge c.15
KTorrent c.15
Monsoon c.16
qBittorrent c.16
Torium c.17
Transmission c.17
Vuze c.18

ВitTorrent — это протокол для обеспечения доступа к файлам по одноранговой (P2P) сети, и зачастую он предоставляет самый быстрый способ получить последние дистрибутивы. Его развитие ушло далеко за пределы изначальной реализации на Python и сейчас включает различные добавочные технологии, типа Distributed Hash Tables, Peer Exchange и других. Современный клиент BitTorrent должен справляться с шифрованием и работой за маршрутизаторами NAT (предпочтительно с UPnP) и прочими акронимами. Мы также ожидаем от него легкости использования, но при изобилии функций и хорошей настраиваемости.

Интернет-провайдеры ведут с торрент-клиентами войну, желая (естественно) применять технологию шейпинга трафика для расстановки приоритетов в пользу Facebook, и если вы не хотите быть уличеными, поспособствуют такие технологии, как шифрование и случайный выбор портов.

В нашем Сравнении мы рассмотрели только клиенты, созданные специально для Linux. Стандартный клиент *BitTorrent*

все еще жив, но его последний релиз для Linux был настолько давно, что мы не стали с ним связываться. Клиенты, подобные *Miro*, активно развиваются и регулярно обновляются, но выполняют узкоспециальные задачи, и их мы тоже исключили. Кроме того, есть множество клиентов, работающих из командной строки, типа *rtorrent* и *aria2*, но они выглядят ужасно на экранных снимках, вот и остались за бортом — Сравнение-то не резиновое.

«Мы ожидаем легкости использования, но и изобилия функций, и настраиваемости.»

Deluge

Выйду ль я... и промокну насквозь.

Клиент *Deluge* [англ. потоп] может работать в различных режимах. Написанный на Python, он использует графический интерфейс *GTK*, но это не приложение *Gnome* — он так же хорошо выглядит и на рабочем столе *KDE* или на любом другом.

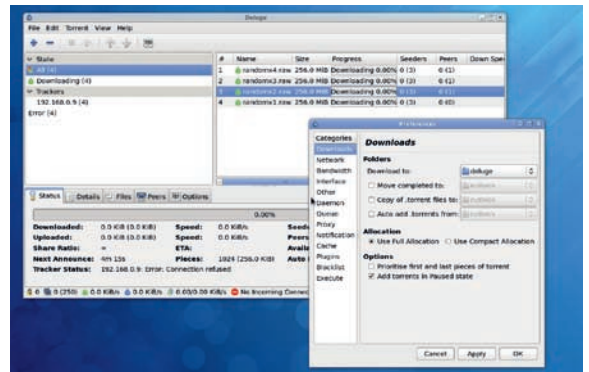
Раскладка главного окна программы довольно проста — дерево в левой части экрана дает вам выбрать нужный элемент (скажем, текущие закачки), который затем отображается в главной панели. В окне статуса внизу выводятся всякие полезные комментарии — например, текущая скорость скачивания и число подключенных и найденных источников [peers]. *Deluge* также поддерживает системный лоток, где отображается информация по скорости скачивания, и предлагает опции по контролю скачки и закачки файлов и добавления торрентов без обращения к основному интерфейсу — милый штрих, отсутствующий у других клиентов.

Функциональность базового клиента обеспечивается основным приложением,

но для многих полезных функций — например, списков блокировки или расписания — придется подключать модули расширения. Модулей этих немало, и их можно без труда установить из *Deluge*. Запустить файл непосредственно из списка скачиваемых нельзя, и, к нашему разочарованию, отсутствует поиск торрентов.

Неприятная особенность — скользкая среднее скорости скачивания продолжает «скользить» даже по завершении закачки. На время написания готовилась версия 1.3, и, возможно, когда вы будете это читать, недостаток уже устранят. Данный клиент вовсе не плох, хотя ему недостаёт интеграции с системой, характерной для других клиентов, и он выглядит среднячком, если говорить о функциях. Зато он работает и лишку ресурсов не ест.

«Для многих функций придется подключать модули расширения.»



» *Deluge* отображает информацию в системном лотке.

LINUX Вердикт
FORMAT

Deluge

Версия: 1.2.3
 Сайт: <http://deluge-torrent.org>

» Среднячок среди подобных, но окружен любовью разработчиков.

Рейтинг 6/10

KTorrent

Клиент под *KDE*, использующий библиотеку *libktorrent*.

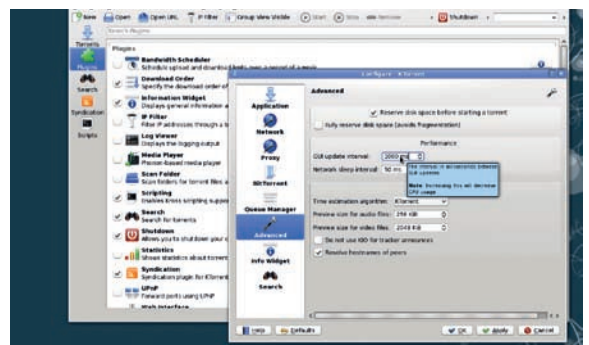
От приложений, написанных под *KDE*, ожидают множества настроек, опций и меню, и *KTorrent* — не исключение. Его не пришьешь за приложение *Mac OS X* — кажется, каждый пиксель его интерфейса стремится выдать вам важную информацию или некую опцию. Для тех, кто не боится избытка функций или недоволен отсутствием подробностей в интерфейсе своего приложения, он, вероятно, совсем неплох.

По части используемых технологий, *KTorrent* сравним с другими клиентами нашего Сравнения и даже их превосходит — кроме *Vuze*, до которого ему не хватает трекера, медиа-плеера и медиа-сервера *UPnP*. Если честно, поиск тоже не ахти. Несмотря на массу вовлеченных поисковиков, *KTorrent* просто швыряет вас на соответствующий сайт, а это с его стороны лень — *qBittorrent* гораздо эффективнее. Но в остальном, *KTorrent* имеет все базовые функции.

Благодаря модулям расширения доступны добавочные возможности, тоже разработанные весьма тонко. На огромном календаре можно распланировать ширину канала, и если вы намерены разрешить неограниченную закачку только по средам, *KTorrent* — единственный клиент, готовый вам помочь. Как и другие приложения *KDE*, он предоставляет все многообразие функций — а уж вам решать, нужны они или нет.

KTorrent — единственный клиент, позволявший установить соединения со всеми доступными пиринговыми сетями за время наших тестовых закачек, и он оказался самым быстрым. Как и некоторые другие торрент-клиенты, *KTorrent* можно запускать на системе без монитора; он также предлагает встроенный *web*-интерфейс.

«Ktorrent сравним с другими и даже их превосходит.»



» Опции, параметры и прочее — настройкой не хочу.

LINUX Вердикт
FORMAT

KTorrent

Версия: 4.0
 Сайт: <http://ktorrent.org>

» Набит различными функциями, если у вас дойдут руки их настроить.

Рейтинг 9/10

Monsoon

Для разнообразия, с библиотекой *MonoTorrent*.

Э то слегка простеющее приложение появилось как ответвление или продолжение проекта *MonoTorrent* – BitTorrent-библиотеки, написанной с использованием Mono-реализации .NET. Большинство клиентов в нашем обзоре использует библиотеку *libtorrent*, и мы ожидали от *Monsoon* некоторых отличий в работе – и получили их. Он-таки глотает циклы центрального процессора, но зато обеспечивает большую пропускную способность, с лучшим средним временем при локальной тестовой зачке (обгоняя даже *KTorrent*).

Умея отслеживать каталог в ожидании новых торрентов, а также сохранять активные и завершённые зачатки в разных местах, *Monsoon* получает зачет по части управления файлами. Но в нем нельзя создавать расписание или скрипты, не предлагается модуль расширения и в помине нет DHT или Peer Exchange (хотя эти функции поддерживаются в *MonoTorrent*).

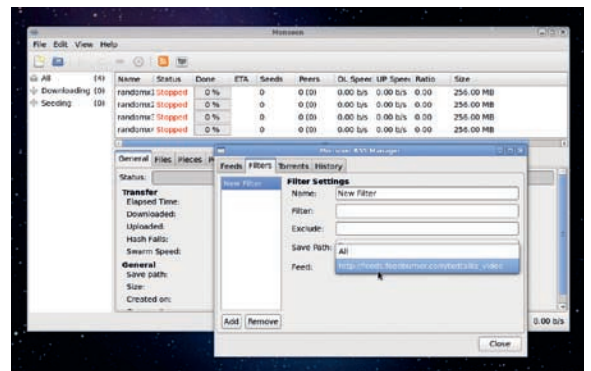
Визуально, программное обеспечение *Monsoon* выдержано в до боли знакомой

теме, с категориями слева, главным (сортируемым) списком в центре и информационным экраном на основе вкладок для выбранного торрента внизу. Как большинство торрент-клиентов данного Сравнения, *Monsoon* предоставляет разнообразные опции, включая индивидуальную настройку полосы пропускания для торрент-файла.

Как в *Vuze* и *KTorrent*, есть подписка на новостные ленты. Так распространяются многие видеоподкасты, и настройка очень проста. Предусмотрены также продвинутое опции фильтрации, для загрузки именно того, что вы хотите.

Monsoon – проект многообещающий; правда, его релизы какое-то время не выходили, хотя библиотека *MonoTorrent* недавно обновлялась.

«Глотает циклы ЦПУ, но выдает большую производительность.»



➤ Движок подписки на ленты новостей умеет выполнять фильтрацию.

LINUX FORMAT **Вердикт**

Monsoon

Версия: 0.21
 Сайт: www.monsoon-project.org

» Ощущается нехватка некоторых функций, но качок энергичный.

Рейтинг 5/10

qBittorrent

Клиент на базе Qt, с рядом приятных сюрпризов.

Данный кросс-платформенный клиент более или менее подогнан под стандарт «дерево, главная табличная панель и детали во вкладках внизу». Лишь большие и удобные кнопки вверху подсказывают, что это не *Monsoon* или один из прочих клиентов этого Сравнения.

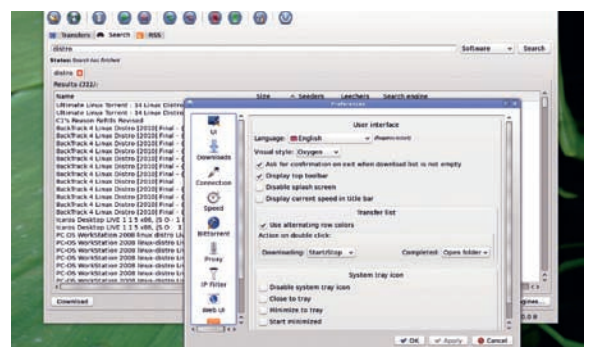
Хотя он выглядит минималистским, внутри вы найдете хорошее наполнение. Поддерживается хитроумная читалка/фильтр подписок на новостные ленты, для помощи при скачивании сериалов или видеоподкастов. Там прячутся всевозможные опции, но раскладка их удобная и легко осваивается. Правда, можно было бы и добавить подсказки по опциям и настройкам.

Есть еще консоль, которая показывает эквивалент журнала событий, а также перечень запрещенных IP-адресов (если вами был добавлен список блокировок), хотя они обычно исчисляются тысячами, что отчасти лишает эту функцию смысла.

А вот поисковая система далеко не бессмысленная. Используя механизм модулей расширения для подключения множества сайтов с торрентами, *qBittorrent* предлагает «унифицированную» стратегию поиска, которая во многих случаях полезнее, чем, например, один-за-раз подход *KTorrent*. Просто введите слово для поискового запроса (и выберите категорию для уточнения результата) – и получите полный экран результатов с поддерживаемых сайтов, упорядоченных по готовности раздачи. Здорово экономит время!

С точки зрения производительности по тестам, а также потребления ресурсов, он не блещет, но добавочные функции делают его чуть полезнее для типовой работы с торрентами, чем *Monsoon* и иже с ним.

«Гибридная стратегия поиска здорово экономит время.»



➤ Прекрасные возможности для поиска торрентов и огромное количество настроек.

LINUX FORMAT **Вердикт**

qBittorrent

Версия: 2.2.8
 Сайт: <http://qbittorrent.sourceforge.net>

» Простекого вида торрент-клиент таит массу отличных функций.

Рейтинг 7/10

Torium

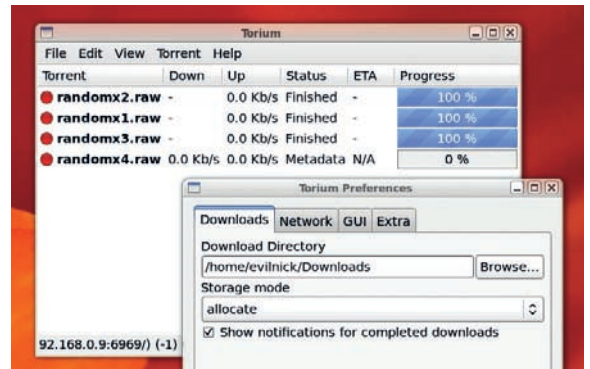
Хорош для маломощных систем, но только как последний шанс.

Клиент легчевеснее, чем *Torium*, найти трудно. Потребляя меньше памяти, чем средний MP3-файл, этот крошка способен неплохо скачивать торренты на приличной скорости; но кому подавай крутые навороты, тот обратится не по адресу.

Интерфейс выглядит несколько пустоватым, поскольку программа не богата функциями. Единственная панель представляет собой таблицу из шести колонок – достаточно для отображения названия торрента, текущей скорости и прогресса. Если вам нужны красивые графики или диаграммы – что ж, берите карандаш и рисуйте их сами. Нажав правую кнопку мыши и выбрав Properties [Свойства], вы получите дополнительную информацию по скачиваемым торрентам, что, по крайней мере, позволит назначить для них приоритеты и локации и просмотреть текущий список сетей. В меню есть опция для скачивания списка запрещенных IP-адресов, но он, похоже, мало на что влияет.

Главная беда – отсутствие шифрования. Поскольку для многих клиентов это один из способов обойти ограничения провайдеров на полосу пропускания, найти «пиров», скорее всего, будет сложно (хотя DHT имеется), вследствие того, что все больше и больше клиентов устанавливают зашифрованное соединение по умолчанию. Не исключено, что вам будет не хватать и других функций, но *Torium* ищет и скачивает торренты довольно умело, так что с базовыми задачами проблем нет.

Если вам позарез нужен торрент-клиент на крайне скудных ресурсах устройства, *Torium* вас выручит, но из-за отсутствия современных функций вряд ли он потянет что-то, кроме торрентов, самых популярных по раздачами.



» *Torium* очень легок – и по потреблению ресурсов, и по составу функций.

«Кому подавай крутые навороты, тот обратится не по адресу.»

LINUX Вердикт
FORMAT

Torium

Версия: 0.4.2
 Сайт: <http://bithack.se/projects/torium>

» Облегчен до крайности, но ценой утраты важных функций.

Рейтинг 3/10

Transmission

Как совместить простоту использования и обилие функций?

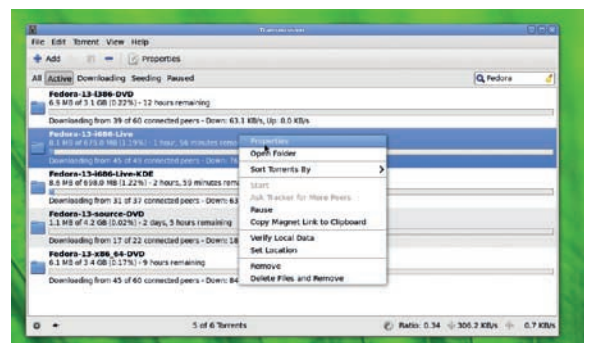
Чтобы стать торрент-клиентом по умолчанию (то есть автоматически и самым популярным) в Ubuntu, придется постараться. Для *Transmission* это произошло благодаря повсеместной рационализации и упрощению – сложно тут разве что ошибиться. Иногда, однако, сложно бывает сделать и нечто конкретное – такое случается, когда приложения упрощены до уровня защиты от дурака. Похоже, отсутствует контроль над количеством одновременно запущенных торрентов, что досадно: ведь это прекрасный способ настройки быстрого действия. *Transmission* предлагает современную поддержку прокси, все последние DHT/PEX и magnet-технологии, а также обнаружение локальных «пиров» сетей, шифрование и блок-листы.

Интерфейс клиента дружелюбен и легок в использовании. Прокрутка в главном окне показывает все торренты, активные торренты, раздачи, файлы для скачивания и отложенные торренты. Простой инди-

катор прогресса скачивания и скорости передачи – это все, что вам нужно, хотя следует отметить, что списки становятся неподъемными, если вы работаете со множеством торрентов.

Другой плюс *Transmission* – его упор на снижение нагрузки на ресурсы. Как и ряд других в нашем обзоре, его также можно запустить на системе без монитора с опциональным web-интерфейсом. Урезанные ресурсы *Transmission* позволяют легко разместить его на встроенном компьютере или NAS.

Transmission все еще в разгаре разработки; изменения и исправления добавляются на регулярной основе. Любители простоты могут на этом клиенте и остановиться.



» *Transmission* – это просто, если только у вас не сотни торрентов.

«Благодаря упрощению, в Transmission сложно разве что ошибиться.»

LINUX Вердикт
FORMAT

Transmission

Версия: 2.0
 Сайт: www.transmissionbt.com

» Очень прост в использовании, а активная разработка делает его еще лучше.

Рейтинг 7/10

Vuze

Самый популярный клиент в мире.

Vuze, который раньше фигурировал под именем *Azureus Vuze*, а еще раньше – *Azureus*, попал в лидеры на всех платформах почти сразу после релиза. Оно и понятно: он первым предложил высокую степень настраиваемости и инновационные функции. Он также был доступен под Linux с момента выпуска, поскольку написан на Java, но по той же причине процветал в основном на машинах под Windows: под Linux Java исторически работала медленно, и *Azureus* казался громоздким, неотзывчивым и алчным до системных ресурсов. Последующие версии программы работают на улучшенных версиях Java, отчего и пошли в гору.

Внешне *Vuze* сугубо индивидуален, хотя некоторые из приемов его UI выглядят заимствованными из настольной Mac OS X.

Медиа-машина

Vuze справляется со всекими файлами, при ненавязчивом уклоне в сторону медиа. Для начала заметим, что завершенные закачки перемещаются в раздел Unwatched [Непросмотренные], где вы найдете свои файлы, снабженные миниатюрой предпросмотра (если, конечно, это видео). Приятная особенность – возможность подписки на видеоподкасты прямо из *Vuze*: если вы скачали одну серию из предложенного *Vuze* списка или распознаваемой ленты, опция закачки будущих эпизодов автоматически появится под знакомой иконкой ленты. В *Vuze* есть встроенный плеер – на нашей тестовой машине

он почему-то не заработал, но файлы запустились без проблем, в проигрывателе по умолчанию. Если вы хотите посмотреть их с другого компьютера, на это в *Vuze* предусмотрен медиа-сервер, который в сочетании с методами распознавания UPnP позволяет скачать файлы на настольный компьютер и затем смотреть их, скажем, на ноутбуке. Если честно, эти функции как раз и вызвали заметное раздутие *Vuze*.

Для многих главной фишкой является включенная в *Vuze* автоматическая модерлируемая лента легального контента, доступного для скачивания. Она разделена на категории, и при всем несовершенстве предлагает немало интересных торрентов – вы можете открыть для себя кучу видеоблогов или скачать последние трейлеры к фильмам. Отдельного канала для дистрибутивов Linux нет – тут небольшой пробел...

Все мыслимые функции настраиваются через свойства. Настройка разумно поделена на разные уровни эксперти-



► Прекрасная подборка бесплатного, абсолютно легального контента стоит лишних очков, верно?

рез web (даже со смартфона), имеются горы доступных модулей расширения – на любой каприз. Если вам не жаль потратить несколько дополнительных циклов процессора, включите визуализацию в виде 3D-облака или запустите нечто полезное – например, самостоятельно сформируйте свой трафик, чтобы зря не гневить интернет-провайдера.

«Имеются горы доступных модулей расширения – на любой каприз.»

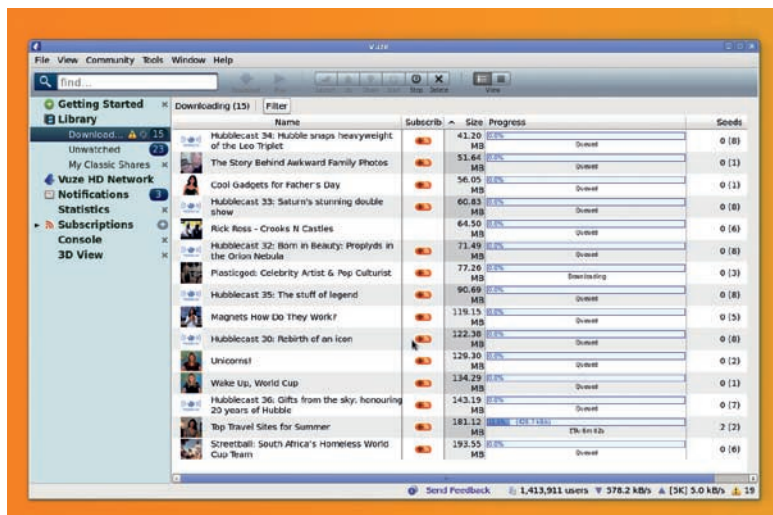
зы – не включив режим Эксперта [Expert], вы не увидите всей прорвы параметров, о которых в 90% случаев вам и знать-то незначет.

Кроме встроенных функций, куда входит удаленное управления *Vuze* че-

Время тестирования

Результаты тестов *Vuze* нас несколько удивили: он упорно демонстрировал плохую производительность. Довольно быстро находя доступные хосты и поддерживая постоянную ширину канала при установленных соединениях, он все же отставал даже от теоретических показателей. Возможно, это больше проблема с Java, чем с кодом самого *Vuze*. В реальности, когда ширина канала редко достигает мегабита, он ведет себя не хуже других клиентов (хотя и потребляет больше памяти).

Vuze, может, и раздут, но на современном рабочем столе его отзывчивость вполне достойна родного Linux-приложения. Пускай ему не хватает некоторых функций (например, он использует свою систему оповещений, а не умолчания ОС), но это, тем не менее, цельный и дружелюбный торрент-клиент.



► Симпатичные миниатюры предпросмотра для закачек привлекают к *Vuze*, несмотря на шероховатости интерфейса (из-за Java).

LINUX Вердикт
FORMAT

Vuze
Версия: 4.4.0.6
Сайт: www.vuze.com

» Ну да, большой и раздутый; зато полнофункциональный, и работает чудесно.

Рейтинг 9/10

Клиенты BitTorrent

Вердикт

Vuze 9/10

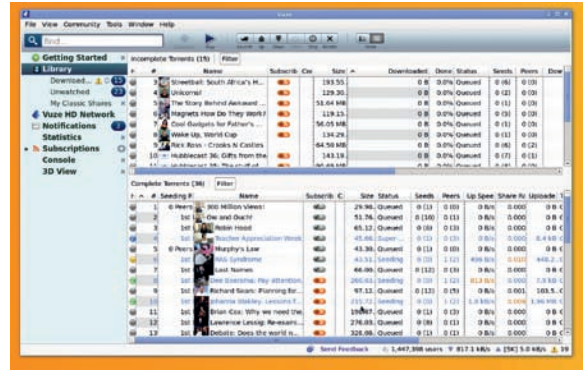
Хотя мы и провели своего рода анализ, важно не ожидать от него многого. Ради объективности, нам пришлось убрать некоторые факторы окружающей реальности, такие как Интернет, проблемы с задержками и подобные. При этом *Vuze* показал прескверные результаты в тестах: несмотря на потребление большей памяти и примерно те же загрузки центрального процессора, что и у всех, скорость его закачки не превышала 2 МБ/сек, даже при пропускной способности сети в 100 Мбит/сек. Вероятнее всего, на результаты повлияло использование Java.

Transmission, согласно тестам, очень даже неплох и предлагает большинство функций, необходимых среднему пользователю. Он довольно прост в применении, и, возможно, потому и включен в большинстве дистрибутивов по умолчанию. Но это все в ущерб параметрам, устанавливаемым

пользователем. На настройках по умолчанию приложение работает хорошо, но может настать момент, когда захочется, например, поднастроить вручную количество торрентов и соединений.

KTorrent где-то сияет, а где-то и плетется в хвосте. Поисковая система – большое удобство, но по сути она всего лишь открывает в браузере соответствующий сайт, хотя и захватывает ссылки для скачивания и предлагает добавить их в очередь на скачивание. С другой стороны, функций в *KTorrent* море и работает он быстро, хотя немного запутан в настройке.

В целом, Сравнение далось нам тяжелее, чем могло показаться. *Vuze* выглядит



» Vuze не идеален, но универсализм делает его достойным клиентом.

лучшим по характеристикам, простоте использования и интеграции с системой. А при встроенном медиа-сервере, его можно назвать универсальным решением, непревзойденным в большинстве задач. Гоняясь за скоростью, берите нечто вроде *KTorrent*, а для простоты использования, без забот о роях, обратных запросах DNS и прочем, стоит взять *Transmission*. **LXF**

Обратная связь

Какие функции торрент-клиента для вас главные? Важно ли для вас то, что клиент большой и раздутый? Мы жаждем узнать о ваших любимых торрент-клиентах, а также и за что вы их любите. Пишите нам по адресу letters@linuxformat.ru

«Пришлось убрать реальные факторы, такие как Интернет.»

Сравнительная таблица характеристик

Название	Deluge	KTorrent	Monsoon	qBittorrent	Torium	Transmission	Vuze
Версия	1.2.3	4	0.21	2.2.8	0.4.2	2.0	4.4.0.6
Сайт	http://deluge-torrent.org	http://ktorrent.org	www.monsoon-project.org	www.qbittorrent.org	http://bithack.se/projects/torium	www.transmissionbt.com	http://azureus.sourceforge.net
Инструментарий	GTK	KDE	Mono/GTK	Qt	GTK	GTK	Java/GTK
PEX	✓	✓	✗	✓	✓ ¹	✓	✓
DHT	✓	✓	✗	✓	✓ ²	✓	✓
Magnet	✓	✓	✗	✓	✗	✓	✓
Шифрование	✓	✓	✓	✓	✗	✓	✓
Лимит по расписанию	через модуль расширения	через модуль расширения	✗	✓	✗	✓	✓
Воспроизведение	✗	через модуль расширения	✗	✓	✗	✗	✓ ³
Расстановка приоритетов	✓	✓	✓	✓	✗	✓	✓
Скрипты	через модуль расширения	✓	✗	✗	✗	✗	✓
Автоматическое отключение	✗	✓	✗	✗	✗	✗	✓
IP-фильтр	через модуль расширения	через модуль расширения	✗	✓	✓	✓	✓
Поиск	✗	✓	✗	✓	✗	✗	✓
Память (в тестах), МБ	21	42,5	68,5	24	7,8	14,5	119
ЦПУ, %	28	26	68	20	25	18	24
Время ⁴ , с	172	110	108	165	184	123	439
Память (простой), МБ	17,5	35,5	13,1	11	5,1	7,4	95

^{1,2} Поддержка заявлена, но непохоже, что работает. ³ Не работало на тестовой системе, но медиа-файлы воспроизводятся на плейере по умолчанию. ⁴ Время загрузки наших четырех тестовых торрентов.



Firefox

Новая глава



Любимец мира свободного ПО находится под угрозой, причем угрожает ему бывший союзник. Расследование ведет **Грэм Моррисон**.

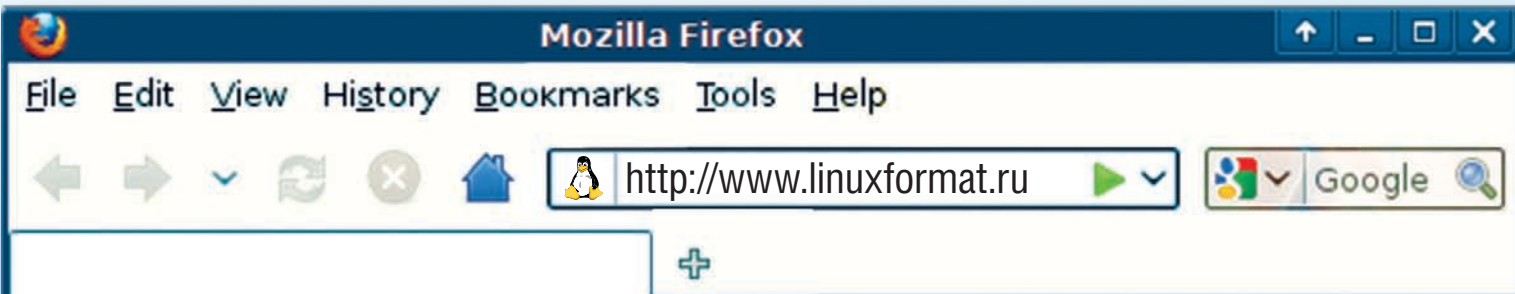
Firefox — это такой открытый проект, о котором, скорее всего, слышали даже ваши друзья, далекие от мира свободного ПО. И, скорее всего, они им еще и пользовались. Фактически, именно Firefox привлек всеобщее внимание к Open Source, попутно «подтягивая» и все обычные технологии и системы.

В то же самое время, если воспользоваться одним из собственных слоганов Firefox, данный проект буквально «отвоевал» Web. Он сделался настолько доминирующей силой, что разработчики сайтов уже не могут, как раньше, игнорировать совместимость с Firefox. Internet Explorer перестал быть единственным браузером, и пользователи Linux наконец-то получили доступ к неусеченной версии Web со своих настольных компьютеров.

«После впечатляющих первых успехов, Firefox стал пробуксовывать.»

Но после впечатляющих первых успехов, миссия Firefox начала пробуксовывать. За последний год рост его был статичным, и наблюдается тенденция к ухудшению. Лидирующий открытый браузер-конкурент сумел подняться от нуля до 7% аудитории и не выказывает признаков замедления разбега — это Chrome от Google, эксплуатирующий ту же мотивацию, что была двигателем развития Firefox; причем Google имеет всеобъемлющие и далеко идущие планы на будущее Chrome.

Однако это отнюдь не конец Firefox как браузера. Это просто конец «медового месяца». Теперь Firefox должен адаптироваться и развиваться в среде, где он уже не единственный свободный браузер, а Internet Explorer — не столь легкая мишень для атаки.



Начало истории...

Предисловие к новым приключениям Firefox.

Появившись в 2002 году, *Firefox* стал антиподом всего, что его авторам не нравилось в раздутом, громоздком, функционально сумбурном *Application Suite* от Mozilla. Mozilla унаследовала многие внешние черты и общую эстетику от своего предшественника, Netscape – компании, вынужденной монетизировать свой браузер и удовлетворять иное сообщество пользователей. В результате ей было очень трудно отстоять свою независимость.

С чувством разочарования, Блэйк Росс [Blake Ross] и Дэвид Хайатт [David Hyatt] скопировали Mozilla и урезали функциональность, а их коллега Бен Гуджер [Ben Goodger] взял то, что он считал нефункциональным пользовательским интерфейсом, и переработал его в стиле минимализма, который весьма способствовал успеху *Phoenix* (как было названо ответвление). Анонсы к релизу утверждали, что «*Phoenix* – это не старый браузер *Mozilla* вашей бабушки. Это очень быстрый браузер без излишеств, но не скудный функциями».

Успех *Phoenix* побудил Mozilla продвигать этот браузер как официальный проект. По словам Гуджера, следующие 18 месяцев были «длинной, сложной, изнурительной дорогой». Но, наконец, 9 ноября 2004 года состоялся релиз *Firefox 1.0*, и мир взликовал.

Восхождение

Firefox был обязан своим ростом и сложившимся обстоятельствам, и набору своих функций, и дальновидности его авторов. В те ранние годы *Firefox* имел преимущество на фоне сомнительной безопасности *Internet Explorer* от Microsoft и неспешности его обновлений. Все это вывело *Firefox* на траекторию, подарившую ему треть рынка браузеров, в основном за счет различных версий *Internet Explorer*.

Firefox ввел просмотр во вкладках, расширения, защиту от фишинга и обеспечение подлинной безопасности каждому пользователю. А главное, этот браузер показал нетехническим пользователям вне мира Linux, что такое открытое ПО и на что оно способно. *Firefox* бесплатен для всех пользователей и работает на массе самых разных платформ.

Google

У *Firefox* были прекрасные отношения с Google, благодаря тому факту, что 90% доходов Mozilla создавалось Google, который в *Firefox* был поисковиком по умолчанию. Напряженность между ними возникла, когда компания Google объявила о намерении разработать *Chrome*, собственный открытый браузер, и выпустила его первую версию для Windows в сентябре 2008 года.

Chrome интересен тем, что существует в двух ипостасях. Официальный продукт Google доступен только как бинарник. Однако, поскольку он базируется на множестве свободных библиотек (включая *WebKit*, *Portable Runtime* и *SQLite*), Google предоставляет также и исходный код, в составе проекта под названием *Chromium*. После компиляции эта версия почти идентична *Chrome*, с некоторыми исключениями: отличаются логотипы,

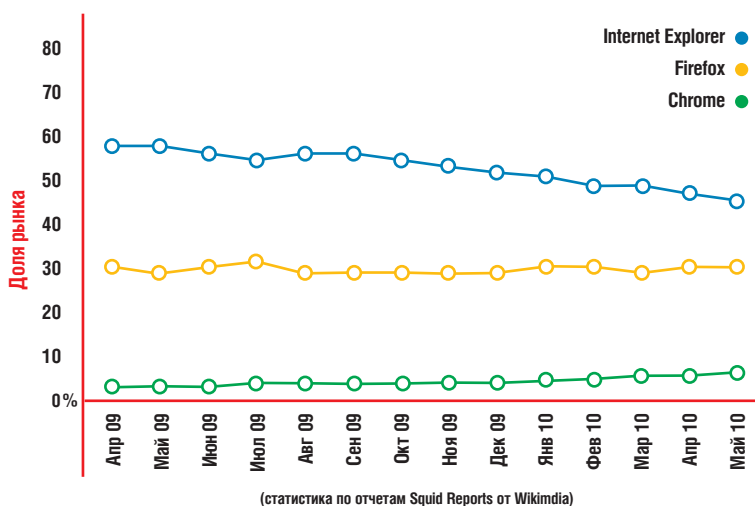


» *Phoenix 1.0* все еще работает в Windows, но не в Linux.

отсутствует инструмент автоматического обновления, и в Google не отсылается статистика использования.

Взлет *Chrome* был бурным – по сути, он повторил *Firefox*, переделав минималистский интерфейс с учетом эволюции функций. Это быстрый, легкий браузер, совместимый с большинством сайтов. Добавьте сюда инсайдерские знания разработчиков о повышении производительности приложений Google и ускорении поиска, и вам станет ясно, чем он так хорош. К маю 2010 года *Chrome* набрал около 7% пользовательской аудитории. Пусть эти 7% и не обязательно переключались на *Chrome* с *Firefox* – все равно это потенциальные пользователи *Firefox*, выбравшие лучшую альтернативу. *Chrome* отвоевывает и удерживает поклонников, бросающих вызов *Firefox*.

Расклад сил по Wikimedia





File Edit View History Bookmarks Tools Help



http://www.linuxformat.ru

Firefox против Chrome

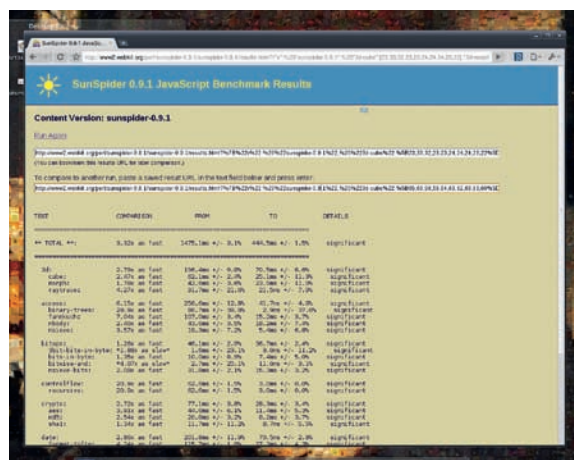
Чем эти браузеры отличаются, и что у них общего?

Чтобы понять, почему *Firefox* и *Chrome* борются за одно и то же место под солнцем, важно уяснить их отличия. И это познается в Сравнении: Сравнение прошлого месяца дало исчерпывающий анализ web-браузеров. Мы выставили *Firefox* отличную оценку – 9/10, отразив его долгосрочную надежность, относительно высокую скорость и гибкость настройки. С таким счетом обычно выходят в победители – но не в данном случае. В забеге участвовал и *Chrome*, и его оценка стала выдающейся: 10/10. Таким образом, именно *Chrome* выиграл наш приз, и именно его мы назвали лучшим web-браузером для Linux.

JavaScript

Скорость – вот самая частая причина, которую называют пользователи, отдавшие предпочтение *Chrome* перед *Firefox*. Не будет преувеличением сказать, что *Chrome* преобразил представления пользователей о требуемой скорости браузера. Отчасти это потому, что сложность сайтов неуклонно растет – чуть ли не каждая щель кишит продвинутыми функциями; а большинство остальных web-браузеров появились до революции Web 2.0, которая перевела множество привычных настольных приложений в разряд облачных вычислений.

Осознав, что большая часть онлайн-взаимодействий осуществляется через JavaScript, группа разработки *Chrome* бросила все силы на обеспечение революционного уровня производительности движка этой технологии. Тем самым они повысили комфорт работы пользователей с сайтами, основанными на JavaScript. Интерес Google к JavaScript был вполне законен – многие облачные



➤ Проверьте способности своего браузера на тесте SunSpider и сравните его с другими.

приложения этой компании обеспечивают свою богатую функциональность тоже за счет данного языка. Но ведь так поступает и вся остальная часть Web. Сейчас вряд ли найдется сайт, не украсивший свою работу парой строк JavaScript-кода.

При своем появлении первая версия *Chrome* вызвала настоящую бурю, показав в JavaScript-тесте SunSpider в два раза большую скорость по сравнению с *Firefox 3.0.1* и в 35 раз – по сравнению с *Internet Explorer 7*. Это достигнуто за счет V8, нового движка

WebKit против Gecko

Когда вы превысите некий порог, работа с Web перестанет определяться только скоростью. Способ визуализации страниц тоже имеет значение, а *Firefox* и *Chrome* применяют для этого разные движки. В *Firefox* это *Gecko*, старинный движок вывода web-страниц [layout engine], применявшийся еще в браузере Netscape. Из-за солидного возраста *Gecko* часто обвиняют в сложности и низкой эффективности.

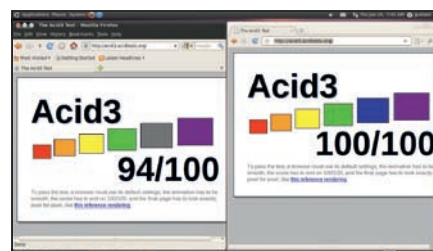
WebKit – более молодой претендент, берущий начало от свободного движка для вывода HTML-страниц в KDE, код которого позаимствовала Apple на год или два. Затем в Apple решили выложить модификации на сервер, и родился движок *WebKit*, ныне используемый многими скоростными web-браузерами, включая *Safari* от Apple и последнюю версию *Konqueror*, а также ряд продуктов от Nokia, Palm, Bitstream и Adobe. *WebKit* повсеместно считается более легким, четким и эффективным движком, чем *Gecko*, но, что еще важнее, *WebKit* гораздо проще. Хотя Mozilla Foundation отклоняет запросы на смену

движка вывода страниц, на том основании, что *Gecko* предоставляет больший контроль.

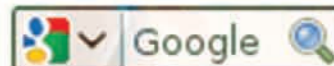
Разработчики Mozilla, несомненно, глубоко понимают эту технологию и заботятся о ее будущем. Недавние улучшения добавили в *Gecko 1.9.3* декодер Ogg, движок OpenGL и, самое главное, видеокодек Google WebM. Но если уж пользователи преимуществ не ощущают, особых причин держаться за старые технологии нет.

По части визуализации web-страниц, оценка качества обоих движков может быть субъективной, однако можно видеть, как оба движка работают с развивающейся спецификацией HTML 5. После выпуска версии 4, *Chrome* стал одним из первых браузеров, который прошел тесты Acid1, Acid2 и Acid3, проверяющие сложность браузера. Так, в финальном тесте *Chrome* показал результат 100/100 – сравните с 94/100 у *Firefox 3.6.3*. Это расхождение обусловлено разными возможностями движков отображения web-страниц. Например, *WebKit* поддерживает 12 из 24 базовых элементов HTML 5

и 20 из 28 элементов форм, а *Gecko* умеет справляться только с двумя элементами из каждой группы. Впрочем, *Gecko* быстро меняется и включает такие функции (например, drag-and-drop), которые *WebKit* еще только предстоит реализовать.



➤ И *Chrome*, и *Firefox* обошли *IE* в тесте Acid3, но результаты *Chrome* лучше.



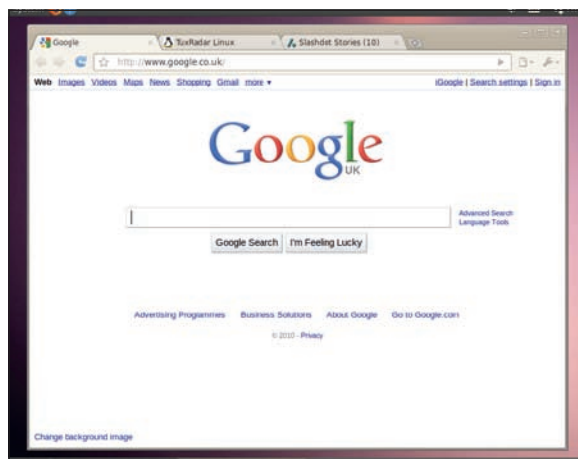
JavaScript, и хотя тесты всегда можно оспорить, поскольку они возлагают на браузеры нереалистичные нагрузки, большинство людей ощущают разницу в скорости, работая с *Chrome*. Тест SunSpider – интересная попытка имитации смеси реальных сценариев, включая такие действия, как декомпрессия, шифрование, математические вычисления и обработка строк.

Разработчики *Firefox* ответили на это своей оптимизацией JavaScript, включив ее в новый движок, *TraceMonkey*. Дополнение Trace Trees к исходному движку *SpiderMonkey* повысило производительность за счет просмотра кода на предмет нахождения общих участков, которые затем прекомпилируются. А *V8* пытается прекомпилировать все, что можно, при первом исполнении переводя JavaScript в машинный код, а не в байт-код, и обеспечивает динамическую природу JavaScript правкой кода на месте через свою систему времени выполнения. Это непросто, но результаты говорят сами за себя.

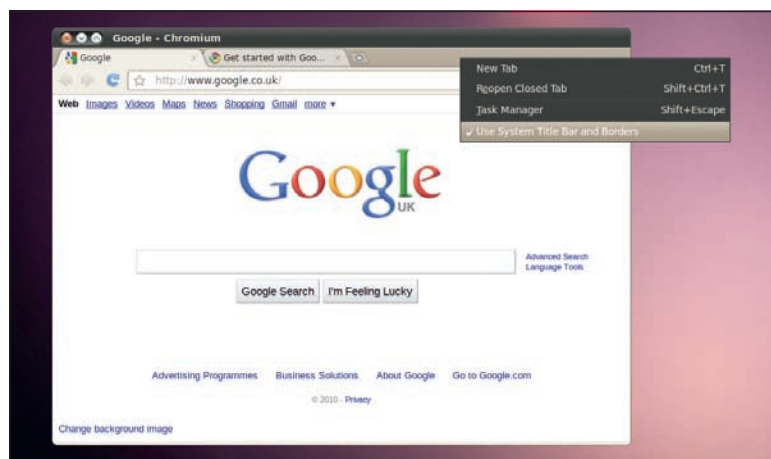
Мы протестировали *Firefox 3.6.3* и *Chrome 5.0.370* с помощью SunSpider 0.9 и увидели, что разрыв между ними увеличился. *Firefox* завершил тесты за 1475,1 мс, а *Chrome* – всего за 444,5 мс, т.е., почти в три раза быстрее, чем при своем дебюте. Но и *Firefox* превзошел по скорости первый *Chrome*. Таким образом, от соревнования разработчиков в создании скоростных движков JavaScript выигрывают пользователи обоих браузеров: оба в будущем обещают многое.

Интеграция с рабочим столом

Firefox глубоко укоренился в Linux, и мы часто считаем его своим родным проектом; это флагман, который мы с гордостью приводим как пример, чего можно добиться на основе свободного кода. В результате, большинство дистрибутивов, не колеблясь, включают его как браузер по умолчанию, кроме, быть может, наиболее упорных фанатов KDE. Причина здесь в том, что *Firefox* построен на основе *GTK*, программного инструментария, примененного для создания Gnome. Поэтому *Firefox* воспринимается как интегрированный в среду Gnome. Панели инструментов, меню, окна с запросами, темы – все выглядит гармонично без всяких



» *Chrome* завоевал немало новых фанатов и принят как браузер по умолчанию во многих дистрибутивах для нетбуков.



модификаций. В KDE, изменения, благодаря которым *Firefox* выглядит как приложение KDE, чисто косметические и иногда выглядят грубоватыми; но все же не такими нарочитыми, как у *Chrome*.

Как пояснил разработчик *Chrome* Бен Гуджер, в Google избегают кросс-платформенных инструментариев – поскольку приложения, использующие таковые, в итоге начинают «говорить с иностранным акцентом»; и для разработки интерфейса пользователя взят *GTK*. В этом есть доля иронии, особенно если учесть, что Гуд-

» С системными заголовком и границами окна *Chrome* выглядит привычнее.

«Firefox глубоко укоренился в Linux; мы часто считаем его своим родным проектом.»

жер, прежде чем встать у руля команды дизайнеров интерфейса *Chrome*, был ведущим разработчиком *Firefox*; но слова его не лишены смысла. При отсутствии многолетней традиции в Linux, использование *GTK* в *Chrome* воздержалось от интеграции с Gnome и устремилось по своему пути. В результате приложение «выламывается» и из Gnome, и из KDE, но зато и не отторгается обоими. Движок тем тяготеет к внешнему виду Gnome как к выбору по умолчанию, но вариантов оформления в *Chrome* немало.

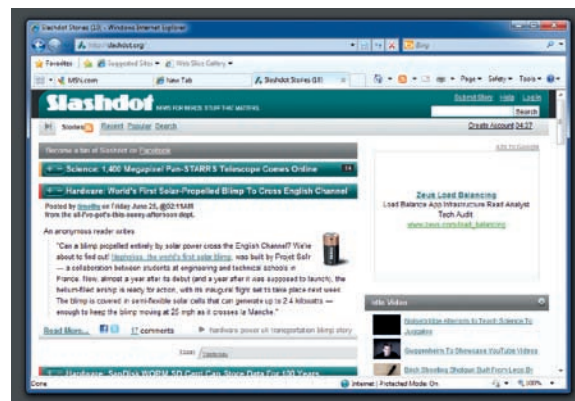
Заметно, однако, отсутствие верхней строки заголовка – по причине того, что в *Chrome* вкладки страниц помещены вверху окна, чтобы эффективнее распорядиться доступным пространством. Верхняя строка заголовка обычно предназначена для взаимодействия с оконным менеджером, который занимается управлением окном, его перемещением и отрисовкой его границ. *Chrome* узурпирует функции оконного менеджера, скрывая его и отображая на границе лишнего рамки окна собственный интерфейс пользователя. Поэтому окно браузера ведет себя несколько иначе, чем ваш «природный» рабочий стол, и *Chrome* менее стабилен, чем *Firefox*, особенно при включенном композитинге. Linux-версии *Chrome* предоставляют опцию восстановления рамки окна, выбором Use System Bar and Borders в контекстном меню, появляющемся при щелчке правой кнопкой мыши по свободному пространству за вкладками.

»

Безопасность и стабильность

Важнейшее свойство браузера – то, что не видно пользователям, а именно, безопасность. Как правило, типичный сеанс работы в обозревателе может осуществлять обмен мгновенными сообщениями, почтой, банковской и другой финансовой информацией, музыкой, фотографиями. При возникновении утечки все это оказывается под угрозой. И здесь-то подход Open Source действительно воссиял – и Firefox, и Chrome открыли свои исходные коды для изучения, и в них кроется на редкость мало сюрпризов. Однако при таких объемах кода совершенство недостижимо, поэтому оба браузера требуют обновлений, по мере обнаружения проблем.

Firefox принимает благородно-открытый подход к безопасности, публикуя список уязвимостей, выделяемых цветом в зависимости от уровня серьезности. За первые четыре месяца текущего года обнаружено 25 уязвимостей; из них 13 признаны критически-ми. Есть даже блог, посвященный безопасности, с ответами на во-



Internet Explorer 8 снял ряд недочетов по безопасности и функциям, на фоне которых Firefox ранее сильно выигрывал.

«Firefox принимает благородно-открытый подход к безопасности, публикуя уязвимости.»

просы пользователей и СМИ. На момент написания данной статьи там был опубликован обзор нового расширения для проверки встраиваемых модулей и способа его обновления, чтоб оповещать пользователей об уязвимостях во Flash от Adobe – этот продукт недавно имел заметные проблемы.

Портал безопасности Chromium не столь открыт, и чтобы узнать последние новости, придется углубиться в специальные форумы по безопасности. Это отражает нежелание «выносить сор из избы», но отнюдь не утаивание серьезных угроз. Последнее стабильное обновление, выпущенное в начале июня, исправило девять серьезных уязвимостей и две проблемы, оцененные как

создающие умеренный риск, хотя большинство этих вопросов касалось WebKit, а не самого приложения Chrome.

Кроме того, по ряду признаков, команды Mozilla и Google охотно помогают друг другу в этой области. Так, именно команда Mozilla обнаружила критическую ошибку в движке V8 Chrome – несомненно, работая над собственным движком JavaScript. Оба проекта также стимулируют поиск ошибок пользователями: Google и Mozilla платят по \$500 за каждую обнаруженную уязвимость. А Mozilla еще и дарит фирменную футболку.

Многопоточность

Если ошибка все-таки просочилась, Chrome способен защитить открытые web-страницы, чем и завоевал последовательность. Открыв в Firefox десять вкладок, а затем еще одну, с «битой» Flash-анимацией, вы теряете все; а в Chrome перестает отвечать только текущая вкладка. Благодаря многопроцессности Chrome, все вкладки запускаются как изолированный процесс в отдельной «песочнице» [sandbox], и сбой внутри «песочницы» не влияет на приложение в целом. Некорректная вкладка заменяется сообщением 'Aw snap', говорящем о неполадках при отображении страницы, и можно перезагрузить страницу либо закрыть вкладку. Этот подход положительно влияет на безопасность, потому что процесс в одной «песочнице» практически не способен посягать на процессы в других «песочницах».

«Песочницы» теперь реализованы даже в Internet Explorer 8 от Microsoft, и следовало бы ожидать, что и Firefox давно перенял эту ценнейшую функцию Chrome. Однако этого пока даже не планируется. Firefox в лучшем случае предложит вам после перезапуска заново открыть все вкладки. Собственный проект «песочницы» у Firefox есть, но это – экспертный процесс, работающий уже более трех лет и призванный гарантировать качество модулей расширения, устанавливаемых с официального сайта.

Есть, правда, и хорошие новости. С выпуском Firefox 3.6.4, версии браузера, которая должна была стать версией 3.7, модули расширения запускаются в виде отдельных процессов, как в Chrome. Это устраняет наиболее частую причину отказа всего приложения, но не распространяется на web-страницы и не спасет вас в случае плохо написанного приложения Flash.

Функция HTTPS Everywhere

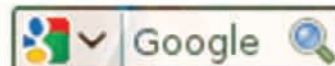
HTTPS Everywhere – одно из лучших расширений Firefox, аналога функций которого у Chrome пока нет. Данный модуль разработан Electronic Frontier Foundation и Tor Project, и позволяет устанавливать HTTPS-соединения везде, где только можно. HTTPS шифрует данные, которыми браузер обменивается с удаленным сайтом, предотвращая их перехват. Кто думает, что перехват трафика по зубам лишь специалистам, тот неправ: во многих случаях для этого достаточно установить лишь одну из утилит вроде Wireshark.

Установка HTTPS Everywhere (см. <https://eff.org/https-everywhere>) очень проста. Перезапустите браузер, выберите опцию Preferences в Extensions Manager и укажите сайты, при соединении с которыми требуется всегда использовать HTTPS. Правда, в небезопасных сетях



Идея применять HTTPS при соединении с сайтами гениально проста.

лучше быть осторожным – трафик HTTPS тоже в принципе поддается перехвату; но это куда сложнее, чем просто запустить Wireshark, так что паниковать не стоит.



Встроенные медиа

Прослушивание музыки и просмотр видео превратились в основные функции web-браузера. Существует множество платных порталов потокового видео, и еще больше – музыкальных порталов. До HTML 5 браузеры обычно запускали для этой цели внешние приложения, такие как *RealPlayer*. Но в HTML 5 данная возможность встроена на уровне движка визуализации. *Firefox* находится на переднем крае этой революции, и реализует свободные форматы, такие как Ogg Vorbis и Theora. Достаточно, например, щелкнуть мышью по файлу Ogg Vorbis, как он автоматически откроется в новом окне и начнет воспроизводиться. Для сравнения: если щелкнуть мышью по файлу MP3, то *Firefox* выведет запрос о том, какое внешнее приложение следует использовать для открытия этого файла.

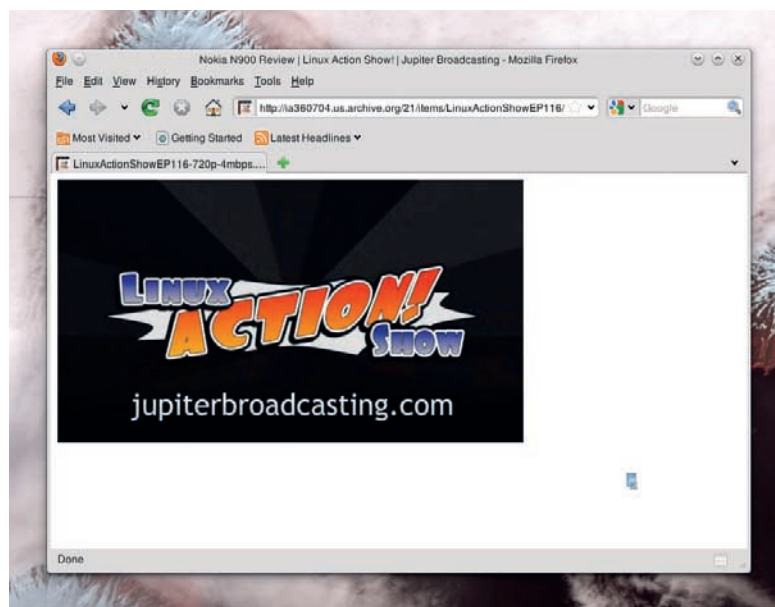
Важность этой функции растет, потому что такие сайты, как Vimeo и YouTube, уже начали переход с Flash на HTML 5. Но эта тенденция готовит также тяжелое испытание для *Firefox*. Как говорит Майк Шейвер [Mike Shaver], вице-президент Mozilla по технологиям, «это важный шаг, дающий видео полноправное гражданство в современном мире Web». Далее он описывает проблему, ожидающую *Firefox* в будущем, когда доминировать будут кодеки, защищенные патентами, наподобие H.264, используемого в настоящее время YouTube и Vimeo:

«Для Mozilla технология H.264 не является наилучшим выбором. Во многих странах она защищена патентами, а значит, без выплаты лицензионных отчислений MPEG-LA использование и распространение ПО, производящее или потребляющее контент в формате H.264, будет незаконным. Даже распространение контента в формате H.264 через Интернет или широкоэвещательные радиосети требует согласия MPEG-LA, и действующее на текущий момент освобождение от лицензионной платы останется в силе только до конца 2010 года». [Освобождение от лицензионных выплат было продлено до 2016 года].

«По одному аспекту Chrome не в силах соперничать с Firefox: это свобода.»

Решение – использовать по-настоящему открытый формат, что означает переход либо на WebM от Google, либо на часто критикуемый формат Ogg Theora. Очевидно, WebM имеет потенциал, при поддержке со стороны *Chrome*, *Firefox* и *IE9*, но дорастет ли он до того, чтобы поддержка H.264 стала несущественной?

Так или иначе, WebM станет важной частью *Firefox 4*, и разработчик Mozilla Роберт О'Каллаган [Robert O'Callahan] объявил об этом после того, как компания Google выпустила новую лицензию для WebM, устранившую неоднозначности и отделившую авторские права от патентов. Если WebM сработает, то *Firefox* окажется на коне. Если нет, пользователям Linux по-прежнему придется полагаться на дополнительные пакеты для доступа к онлайн-контенту. Однако шансы на широкое сотрудничество остаются, так как Майкл Шейвер недавно написал: «Я очень надеюсь, что Google (как команда *Chrome*, так и команда YouTube), Vimeo и многие дру-



гие разделяют наше желание наполнить Web полнофункциональным, высокопроизводительным видео, свободным от патентных ограничений и интегрированным на «родном» уровне, и жду, что мы все – и вместе, и по отдельности – будем работать над этим».

Убойное обстоятельство

Несмотря на все эти разговоры о функциях и о будущем, по одному аспекту *Chrome* не в силах соперничать с *Firefox*: это свобода. Именно свобода и является залогом будущего роста Mozilla. Google, несомненно, разрабатывает *Chrome* в своих целях. Поскольку на подходе их собственная операционная система, а также с учетом завоеванного плацдарма на мобильном рынке, Google несомненно нуждается в личном портале в мир онлайн, в наилучшем свете представляющем приложения и инструментарий Google. *Firefox* от таких ограничений свободен. Это просто браузер, разработанный для предоставления доступа в Web как можно большему количеству людей. Как гласит сайт *Firefox*, его миссия состоит в «продвижении открытости, инноваций и возможностей в Web». Это означает, что интересы пользователей всегда во главе угла, независимо от всяческих искушений. Хотя *Chrome* – тоже открытый продукт, до свободы *Firefox* ему очень далеко.

Митчелл Бэйкер [Mitchell Baker], глава Mozilla Foundation, лучше всего сказала об этом в конце апреля, обнаружив, что ботанический сад, который она решила посетить – не обидарка для туристов, а некоммерческая организация, подобно Mozilla. «Некоммерческие организации могут ошибаться и быть тупыми и неэффективными, как любые другие. Но зато шансов на то, что они созданы только для выкачивания из людей денег, намного, намного меньше». Это справедливо и по отношению к Mozilla.

» Будущее принадлежит встроенному видео, а не внешним кодекам.

»

Будущее Firefox

Firefox 4 выходит в этом году. Посмотрим, что он нам предложит.

Команде Firefox есть о чем подумать, а времени у них не так уж много. Релиз Firefox 4 намечен на ноябрь текущего года, и многие планы его разработчиков были озвучены еще в мае на web-презентации Майка Белцнера [Mike Beltzner], директора Firefox. Сперва он спросил: а кто будет пользоваться браузером? Очевидно, есть существенная разница между твердыми приверженцами продукта уже с его ранних версий, основной массой пользователей и разработчиками; и новая версия браузера должна удовлетворить все эти три несходных группы.

Неудивительно, что первым словом в плане было «скорость». Характерно, что главной заботой стала не гонка за лидерство в тестах производительности [benchmarks], а субъективные ощущения пользователей. Белцнер привел пример того, как дизайн прогресс-индикатора влияет на восприятие. Как следствие, Firefox 4 получит элегантное оформление (тему). Ранние макеты весьма напоминают Chrome. Что касается деталей – отменены модальные диалоги, обновления будут выполняться в фоновом режиме, устранены сбои при запуске, а установка модулей расширения больше не потребует перезапуска приложения.

Важна и безопасность. В новой версии появится модуль менеджера разрешений, отображающий ваши взаимоотношения с сайтом. Сейчас он показывает список сайтов со списками полномочий, в соответствии с возможностями сайта. Можно блокировать данные о географическом местоположении или настроить блокировки всплывающих окон. Но, сохраняя конфиденциальность данных, которые в явной форме не предоставились в доступ, Firefox 4 реализует новый социальный аспект: помощь в обобщении ваших любимых сайтов и даже открытых вкладок.

HTML 5

Проведена большая работа над движком визуализации HTML 5, а поддержка технологии multi-touch звучит очень многообещающе. Достигнут прогресс в области анимации переходов CSS 3, реализованы новый удаленный отладчик JavaScript, менеджер профилей и WebGL – все это компоненты нового web-стандарта Mozilla, призванного обеспечить на web-страницах интерактивную 3D-графику, как в VRML.

Наряду с риторикой о психологическом влиянии интерфейса пользователя на производительность, введен ускоренный движок

Наш прогноз макета Firefox 4

Вкладки приложений

Можно закрепить онлайн-сервисы, скажем, электронную почту, за вкладками, занимающими фиксированное положение, независимо от порядка их вызова.

Перемещаемая панель вкладок

Список вкладок можно переместить с верхней панели окна в нижнюю.

Перемещаемые меню

Панель меню теперь в той же категории, что и строка закладок, и может активироваться и перемещаться по вашему усмотрению.

Поддержка Multi-touch

Проделана большая работа по обеспечению поддержки движком визуализации HTML 5 Multi-touch-жестов.



Модули в «песочницах»

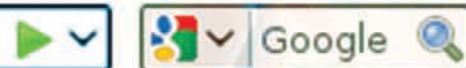
Модули расширения теперь можно устанавливать без перезапуска, а аварийное завершение работы модуля не влияет на работу браузера в целом.

Минималистский дизайн

Новая тема по умолчанию проста, элегантна и выполнена в духе минимализма. Количество кнопок на панели инструментов и в строке состояния резко уменьшено.

Управление конфиденциальностью

Блокировка всплывающих окон, работа с cookie и выдача информации в общий доступ – отдельно для каждого сайта, через модуль Permissions Manager.



JavaScript, *JaegerMonkey* – попытка заменить интеллектуальные процедуры трассировки *SpiderMonkey* более эффективным кодом, компилирующим целые подпрограммы без лишнего анализа. Этот подход идеален для ситуаций, когда процедуры трассировки не работают, поскольку приложение не дает нужного уровня повторяемости. Как объявил в конце мая Дэвид Мэнделин [David Mandelin], программист Mozilla, ответственный за разработку на JavaScript, *JaegerMonkey* «на полпути». Результаты показывают, что эффективность удвоилась, хотя по результатам тестов новый движок пока уступает и *V8* от Google, и *Nitro* от Apple.

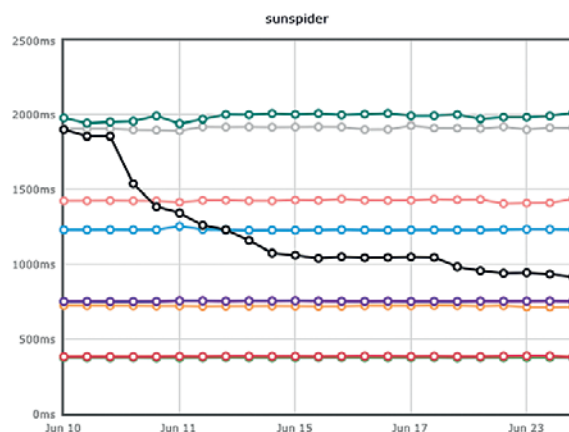
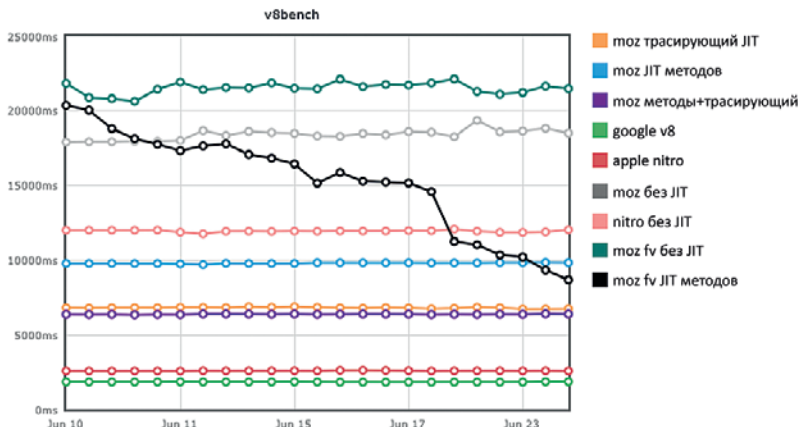
Ускорились не только интерфейс пользователя и движок JavaScript. Разработчики Mozilla обратились в сообщество Reddit за их сверхподробными профилями запуска браузера, с целью увеличить скорость и, возможно, улучшить имидж.

Конкуренция

Подлинным тестом для Firefox будет адаптация на рынке, где доминируют встроенные и заблокированные устройства, что не обязательно упростит жизнь тем, кто захочет им пользоваться. Яркий пример – Chrome OS от Google: это целая операционная система, построенная «вокруг *Chrome*».

Похожая ситуация сложилась на мобильном рынке. Mozilla самоустранилась от iPhone, заявив, что с браузером Apple тяжело конкурировать. Apple не смягчила своей позиции насчет браузеров, сделал исключение только для *Opera*. С Android ситуация иная, и это единственный реальный конкурент iPhone. Поэтому разработчики Mozilla сделали попытку создать мобильную версию своего браузера. Ранняя альфа-версия для Android, *Fennec*, вышла в конце апреля и обладает большинством функций, присутствующих в настольной версии, включая поддержку модулей расширения. Однако очевидно, что работы еще очень много.

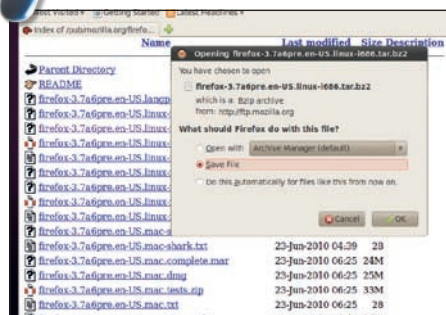
Планы или не планы, большинство пользователей желает знать, что *Firefox* противопоставит *Chrome*. Подход Mozilla прагматичен. Они хотят дать импульс сотрудничеству и избежать конфликтов. Как сказал Белцнер, «мы наблюдаем на наших конкурентах, а они – за нами, и мы здесь, чтобы победить».



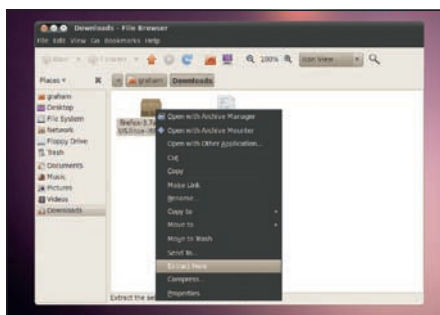
➤ Результаты тестов *Firefox 4* с сайта <http://arewefastyet.com/>.

Для успеха у Mozilla есть нужная квалификация, финансирование и сообщество. Но мир теперь не тот, что раньше, когда стандартом открытого браузера долго был *Firefox*. И все равно конкуренция – это хорошо. Если Mozilla действительно хочет выиграть, выиграем мы все – от изменения мира Linux к лучшему. **LXF**

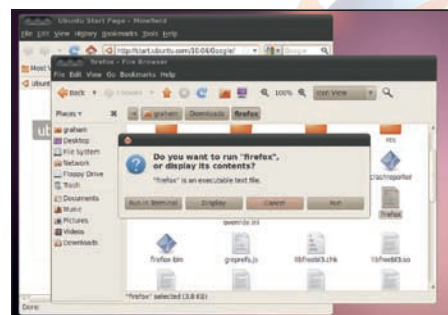
Установка «ночных сборок»



1 Скачайте новейший релиз ➤
Загрузите файл Linux tar.bz2 из каталога ночных сборок с FTP-сайта Mozilla.



2 Распакуйте файл ➤
Откройте папку загрузок, щелкните правой кнопкой мыши на *Firefox* и выберите команду Extract Here.



3 Запустите Firefox
В новой папке найдите скрипт *Firefox* (не *firefox-bin*). Дважды щелкните по нему мышью и выберите команду Run.

Вот пришел Курт!

Джоно Бэкон беседует с Куртом фон Финком из Monty Program о *MySQL*, *MariaDB*, *Oracle* и финской черной водке... М-мм...



История проекта *MySQL* довольно пестрая. Основанный Михаэлом «Монти» Видениусом [Michael «Monty» Widenius], *MySQL* изначально принадлежал *MySQL AB*, которую приобрела *Sun* и которая,

в свою очередь, была приобретена *Oracle*. Не будучи удовлетворен отношением *Sun* к *MySQL*, Монти ушел из *Sun*, чтобы создать *Monty Program AB* и вложиться в более ориентированное на сообщество ответвление *MySQL* под названием *MariaDB*. Сегодня мы беседуем с Куртом фон Финком [Kurt von Finck], главой разработки сообщества *MariaDB*, о положении дел в мире *MariaDB* и *MySQL*.

LXF: Каков ваш фоновый опыт?

Курт фон Финк: #FFFFFF. Основной белый.

LXF: ?!..

КфФ: Ох, извините, вы не то имели в виду. Я уже давно отираюсь в сфере ИТ, занимался поддержкой и управлением проектами. Работал и в компаниях списка Fortune 50s, и в небольших начинающих фирмах. Я давний пользователь и защитник свободного ПО и считаю себя счастливым, поскольку моя карьера отражает мои принципы.

Я – директор *Monty Program* по связям и работе с сообществом; играю эту роль со времени ухода из *Canonical* в феврале 2009, а там я был старшим аналитиком поддержки систем *Ubuntu*. Состою в *Gnome Foundation*, и меня можно встретить на конференциях *FLOSS* – я там раздаю *salmiakikossu*, финскую черную водку.

LXF: Как вы перешли из *Canonical* в *Monty Program*? Переход довольно резкий.

КфФ: Мы с Монти встретились на конференции *eLiberatica* в Румынии в 2007 году. У нас обоих... как бы это сказать...

«свободные и неформальные» манеры, и у меня шведско-финские корни [Михаэль Видениус – швед из Финляндии]. Нам очень понравилось общаться, и оказалось, что мы едины во мнении насчет того, как должны работать программы и компании, занимающиеся программами.

Монти любезно пригласил меня к себе в гости, а я всё откладывал поездку. Монти ушел из *Sun* примерно тогда же, когда я сам ушел из *Canonical* – и я написал ему по электронной почте, чтобы просто пожелать удачи и сообщить, что из-за наших перемещений визит откладывается еще на год; а он сразу же пригласил меня, и я стал третьим по счету работником *Monty Program*.

Через четыре года мы с женой вернулись в США из Канады и навестили Монти в Финляндии. Иногда мне кажется, что моя работа определяется способностью Монти влиять на мои поездки.

LXF: Чем занимается *Monty Program*?

КфФ: Наш главный продукт – *MariaDB*; это замещение *MySQL*, основанное на коде *MySQL*, выпущенном под *GPL*. К этому коду компания *Monty Program* и сообщество *MariaDB* добавляют заплатки, исправления ошибок, новые функции и прочие усовершенствования. Мы пересматриваем код и время от времени возвращаемся к уже имеющимся заплаткам в текущих релизах *MySQL* – для заплаток, не прошедших нашу проверку кода. И это, в свою очередь, укрепляет потенциальную устойчивость *MariaDB*. Вот вам краткий ответ.

За более подробным ответом, обратитесь на наш сайт и почитайте примечания к релизу и стратегические планы. Если вам для работы требуется использовать *MySQL*, вы должны знать, что существует альтернатива с простым обновлением и функцией (функциями), которых вам, возможно, не хватало.

LXF: В чем отличие MariaDB от MySQL?

Кфф: *MariaDB 5.1* (на данный момент) заметно быстрее, богаче функциями и должна быть надежнее соответствующей версии *MySQL 5.1* – в основном благодаря тому, что работает над *MariaDB* большинство первоначальных сотрудников *MySQL*. Мы также применяем *XtraDB* вместо *InnoDB*, что и обеспечивает львиную долю производительности *MariaDB 5.1* – так или иначе добиться только будущим релизам *MySQL 5.5*.

LXF: Много писали о борьбе внутри сообщества MySQL.**В чем отличие MariaDB?**

Кфф: Что касается проекта – возможно, самое большое отличие кроется в нашем характере. Например, если вы хотели сделать из исходника пакет *MySQL* для Windows или MacOS X, было практически необходимо, чтобы вы работали в *MySQL*. Эта задача была какой угодно, но только не простой. *Monty Program* посвятила значительную часть времени и ресурсов созданию открытой системы, способной компилировать и создавать пакеты *MariaDB* без всяких усилий.

Я признаю свою предубежденность, но я честно думаю, что работа с нашим сообществом в целом и обеспечение возможности работать с нами для нашего сообщества – это и есть самое главное различие между *MySQL* и проектом *MariaDB*.

LXF: Monty Program использует так называемую Hacking Business Model; как это работает?

Кфф: *Hacking Business Model* [Бизнес-модель для хакеров] написали *Монти* и *Зак Грент* [Zak Greant]. Это их проект управления небольшой компанией, где работают талантливые и умные программисты.

Основная идея в том, что подобная компания будет работать наилучшим образом и наилучшим образом проявлять свою одаренность, если эти талантливые люди будут иметь голос в управлении компанией и реальную долю в прибыли.

LXF: И это реально работает? Разве это не ведет к хаосу?

Кфф: Это реально работает – в первую очередь потому, что хаос – это отсутствие закона. *Hacking Business Model* – это форма закона, и приверженность изложенным в ней принципам позволяет координировать мнения отдельных личностей, их эго и дела на повестке дня.

Конечно, такая модель сама по себе не будет работать в компании такого размера, как *IBM*. Консенсус и кворум необходимо приводит к энтропии. Поэтому мы всегда советуем компаниям прочитать *Hacking Business Model* и применять только те части, которые имеют смысл для их бизнеса.

LXF: Монти недавно запустил горячую полемику вокруг MySQL, заявив ЕС, что Oracle должны от него отказаться. Не расскажете подробнее?

Кфф: Рассказывать особо нечего. Процесс этот начался как рядовое расследование слияния компаний Европейским Союзом. Комиссия ЕС обычным порядком разослала пользователям и производителям баз данных анкету «по поиску фактов». Мы, как руководители проекта *MariaDB*, формально находились в последней категории, но по сути наша роль заключалась в том, чтобы сообщить Комиссии как можно больше об истории *MySQL* и поделиться внутренней информацией. Это многих озадачило, поскольку нашей компании, как и большинству производителей баз данных, явно было бы выгодно, чтобы *MySQL* попал под контроль *Oracle*. Это потенциально заставило бы многих пользователей искать альтернативы, в нашем случае – *MariaDB*. Так что, к удивлению и замешательству многих, мы вроде как выступили против собственных деловых интересов.

LXF: А зачем выступать против своих же интересов?

Кфф: Поскольку *Монти* – создатель *MySQL*, мы решили, что будет правильнее принять в этом деле сторону пользователей, клиентов, разработчиков и всех, кто построил свое дело на *MySQL*, а не исходить из чисто личной заинтересованности. Как автор, *Монти* ощущает большую ответственность перед кодом *MySQL*, его пользователями и всей экосистемой *MySQL* в целом. В *Oracle* насчет планов по *MySQL* отмалчивались. Учитывая, что *MySQL* – это продукт базы данных и что релизы *MySQL 5.0* и *5.1* были более ориентированы на уровень «готового продукта для предприятия», не возникало впечатления, что *Oracle*, чей доход в первую очередь обусловлен базами данных, будет достойно управлять этим проектом.

LXF: Как вы считаете, что нужно сообществу MySQL для процветания?

Кфф: То же, что и любому сообществу свободного ПО: контроль версий, соединение с Интернетом, хостинг кода, кофеин, алкоголь и отзывчивые, постоянно вовлеченные в работу основные кадры, готовые спать урывками.

LXF: Базы данных наподобие CouchDB затеяли так называемую «революцию NoSQL» – «без SQL».**Что вы с Монти об этом думаете?**

Кфф: Используйте то, что годится вам и вашим пользователям. Если это *MySQL* – отлично. Если *MariaDB* – чудесно. Если это *Drizzle* или *MongoDB*, просто используйте их, и пусть ваше приложение работает.

Мой опыт показывает, что кое-кто склонен создавать антагонизм на пустом месте – просто чтобы убедить самих себя в определенных фактах. Так, администратор базы данных, который полгода убеждал своего начальника, что переход с *MariaDB* на *Cassandra* был бы отличной идеей, станет вряд ли хвалить *MariaDB*. Но вы знаете, *MariaDB* – отличная база данных. Также как и многие проекты *NoSQL*.

Монти сказал: «*NoSQL* – для шибко умных, для тех, кому нужен острый нож. Людям, не способным освоить *SQL*, нечего и пытаться работать с *NoSQL*». Берите то, что работает для вас.

LXF: Что готовит будущее для MariaDB?

Кфф: Проект *MariaDB* будет продолжать выпускать качественные релизы в срок. Будем надеяться, что сообщество, помогающее нам делать это и улучшать нашу функциональность, продолжит свой рост. *Monty Program* и дальше будет самым лучшим управленцем – насколько это в наших силах, будем всячески содействовать и поощрять обратную связь, немало способствующую нашему успеху в данной сфере, и, надеюсь, построим солидный бизнес вокруг нашей поддержки и неповторимой инженерной работы.

LXF: И, наконец, как читатели Linux Format могут влиться в ваше сообщество?

Кфф: Почаще заходите на наш сайт, присоединяйтесь к группе *maria-discuss* на *Launchpad* или к каналу *#maria* на *Freenode IRC*. Поскольку мы – молодая компания, занимающаяся интереснейшим проектом, все очень быстро меняется, и здесь всегда весело. Ну, это я не про «сон урывками». **LXF**

О NOSQL

**«Кое-кто склонен
создать антагонизм
на пустом месте.»**





Знакомьтесь: Launchpad

Джоно Бэкон представляет Launchpad, кузницу открытого ПО.



Наш эксперт

Джоно Бэкон
Менеджер сообщества Ubuntu, автор книги *The Art of Community* и соавтор подкаста *Shot Of Jaz*.

В мире открытого ПО имеются тысячи приложений на все случаи жизни. Чтобы проекты столь разных программ, создаваемых многочисленными добровольцами, успешно развивались, им нужны место для хранения исходного кода, система регистрации ошибок, методика перевода на разные языки, организация поддержки пользователей — и, наконец, процедура планирования и выбора новых функций.

К счастью, вариантов организации коллективного труда хватает. Есть среди них сложные, типа SourceForge, есть и простые — например, Trac. Но есть такое решение, в котором богатство возможностей сочетается с простотой — система Launchpad, родившаяся как попытка упорядочить рабочий процесс «постройки» дистрибутива Ubuntu. Разработка проектов здесь обеспечивается следующими позициями:

«Launchpad — превосходный полигон для исследования ПО.»

» **Отслеживание ошибок** Место для регистрации, «отбора», исправления и обработки ошибок.

» **Хранение кода** Место для хранения кода и систематизации многочисленных «ответвлений».

» **Переводы** Место, где не-программисты переводят текст на иностранные языки.

» **Сборка пакетов** Место для сборки пакетов для Ubuntu и выдачи их пользователям.

» **Учёт спецификаций** Место планирования функций, добавляемых в проект.

» **Поддержка сообщества** Место, где задают вопросы и отвечают на них.

Сейчас мы рассмотрим некоторые из этих пунктов подробно и узнаем, как начать работу в системе Launchpad.

Давайте начнём с получения учётной записи. Запустите web-браузер, перейдите на сайт www.launchpad.net и нажмите

ссылку Log In/Register в правом верхнем углу страницы. Следуя инструкциям, создайте учётную запись и вернитесь на Launchpad, уже под своим именем. Теперь посмотрим, из чего состоит и как работает базовый интерфейс.

Launchpad поделен между множеством проектов; обычно каждый посвящён одному программному продукту. Например, зайдите на www.launchpad.net/lernid: этот проект я недавно начал разрабатывать. *Lernid* – средство организации электронного обучения по каналам IRC. Мы уже давно ведём недели обучения для сообщества Ubuntu, и *Lernid* упрощает доступ к занятиям, а также служит для демонстрации презентаций и web-страниц в реальном времени (LXF133). Это хороший пример конъюнктурного проекта. Он создан, чтобы удовлетворить назревшую потребность, и Launchpad помог многим добровольцам вовремя «подставить плечо».

Страница Launchpad делится на несколько ключевых секторов. В верхней части приведены основные рубрики.

- » **Overview [Обзор]** Это главная страница проекта с общими сведениями.
- » **Code [Код]** Здесь хранится код.
- » **Bugs [Ошибки]** Здесь регистрируются недостатки и дефекты.
- » **Blueprints [Намётки]** Раздел для спецификаций и проектирования новых функций.
- » **Translations [Переводы]** Система перевода текстовых строк.
- » **Answers [Ответы]** Поддержка конечных пользователей.

Все эти разделы предназначены для разных групп добровольных помощников.

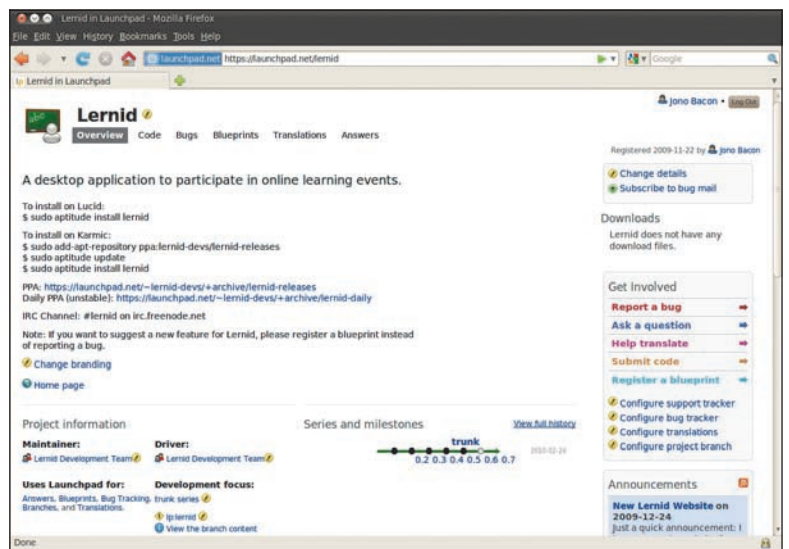
Получение кода

Это одна из важнейших функций Launchpad: здесь можно найти исходные коды множества приложений. Всё это можно загрузить и изучить, а потом придумать собственные модификации и дополнения. Для программиста Launchpad – превосходный полигон для исследования ПО и возможность внести собственный вклад в разработку множества приложений.

Launchpad тесно связан с системой контроля версий *Bazaar*, что обеспечивает беспрепятственную загрузку и выгрузку кода, а также средства контроля и ревизии модификаций. Чтобы использовать эти возможности, необходимо установить систему *Bazaar* (скорее всего, в вашем дистрибутиве она уже есть).

Например, чтобы загрузить исходный код *Lernid*, перейдите на страницу проекта www.launchpad.net/lernid и нажмите кнопку Code. Вы увидите массу ответвлений: в любом проекте их немало. Но люди в основном интересуются «главной ветвью» – стволом (trunk).

Ствол – основа проекта. Это официальное направление, отражающее текущее состояние разработки. В ответвлениях мо-



» **Представление проекта: сводная информация о приложении, со ссылками на основные составляющие.**

гут иметься какие-то свои функции, и многие из них впоследствии снова вливаются в ствол.

Загрузить код ствола просто: достаточно скопировать

```
bzr branch lp:lernid
```

Если вам нужен код другого проекта, просто измените название после `lp:`. Если хотите загрузить другое ответвление, щелкните по нему, и Launchpad подскажет команду `bzr` для загрузки этой ветви.

Количество проектов превышает 17 000 – есть где разгуляться! Мало того, все пакеты Ubuntu тоже хранятся на Launchpad. Например, чтобы загрузить код приложения для микроблогов *Gwibber*, которое войдёт в новый Ubuntu 10.10 Maverick Meerkat [Мигрирующий Мангуст], просто наберите:

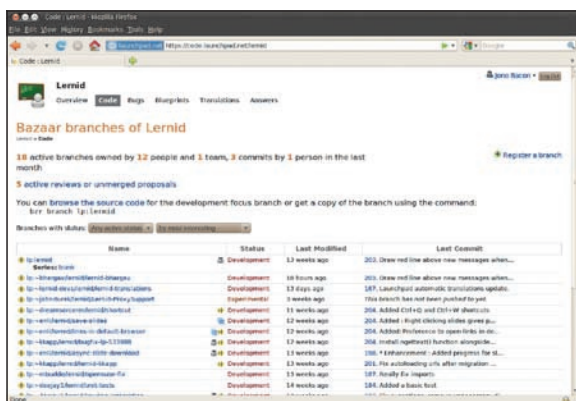
```
bzr branch lp:ubuntu/maverick/gwibber
```

Как видите, принять участие в разработке Ubuntu совсем не сложно: загрузка кода и управление им осуществляются из единого центра.

Внесение изменений

Загружая код проекта, вы, вероятно, хотите улучшить его добавкой какой-либо функции или исправлением ошибки. Вернуть исправленный проект хранителям несложно. Это делается в три этапа:

- 1 **Фиксация** Сохраните изменения на локальном компьютере.
- 2 **Возврат** Выгрузите своё ответвление на Launchpad.
- 3 **Предложение** Предложите автору проекта присоединить исправленное ответвление.



» **Представление кода: ответвления проекта, собранные со всех уголков Launchpad.**

Ещё о Launchpad: Намётки

Большинство новых функций проекта создаётся коллективно. Соответствующие записи ведутся в виде спецификаций, понятных всем членам сообщества: это необходимо для «сверки часов» в процессе разработки. Команда Launchpad реализовала эту функцию в виде особой рубрики, «Намётки» [Blueprints].

«Намётки» – это место для документирования новой функции, разработки плана её реализации, привязки к общей спецификации, определения важности и приоритета; аналогично процессу отслеживания ошибок. Кроме того, рубрика намёток служит информационной панелью, на которой записываются изменения статуса и обновления. Подписавшись на эту рубрику, пользователи получают электронные сообщения о происходящих событиях.



Вернёмся к нашему примеру с *Lernid*: представьте, что вы улучшили проект редактированием кода или составлением документации. На вашем компьютере изменения работают, и вы решили зафиксировать их в системе *Bazaar* – тогда сообщите *Bazaar* о необходимости сохранить изменения, вот так:

```
bzr commit
```

Система предложит составить сопроводительное сообщение с описанием изменений. Теперь можно выгружать изменённые данные на Launchpad. Наберите команду:

```
bzr push lp:~<имя_пользователя>/<проект>/<имя_ветви>
```

Например, для меня, пользователя с именем **jonobacon**, выгрузка ответвления на *Lernid* будет выглядеть так:

```
bzr push lp:~jonobacon/lernid/my-new-branch
```

Перейдя на <http://code.launchpad.net/lernid> для просмотра кода, я обнаружу и свой вклад (раз я указал *lernid* как адрес для *push*). Теперь каждый может загрузить обновлённый код и проверить, всё ли действует как надо. Если я захочу предложить главным разработчикам *Lernid* включить мои изменения в «ствол», то начну с нажатия кнопки *Propose For Merging* [Предложить к слиянию]. Затем, добавив краткое описание своего ответвления, нажму кнопку *Propose Merge* [Предложить слияние]. После этого разработчики смогут проверить мою работу.

Регистрация ошибок

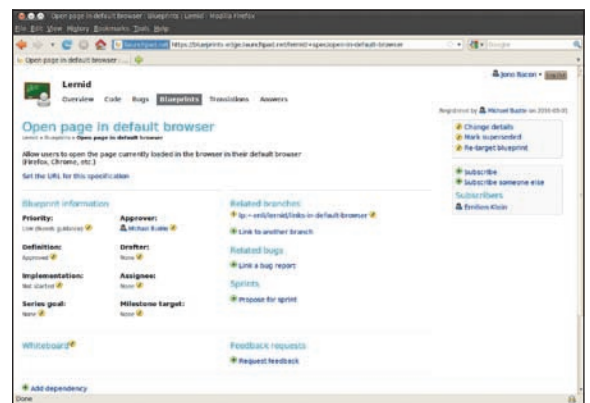
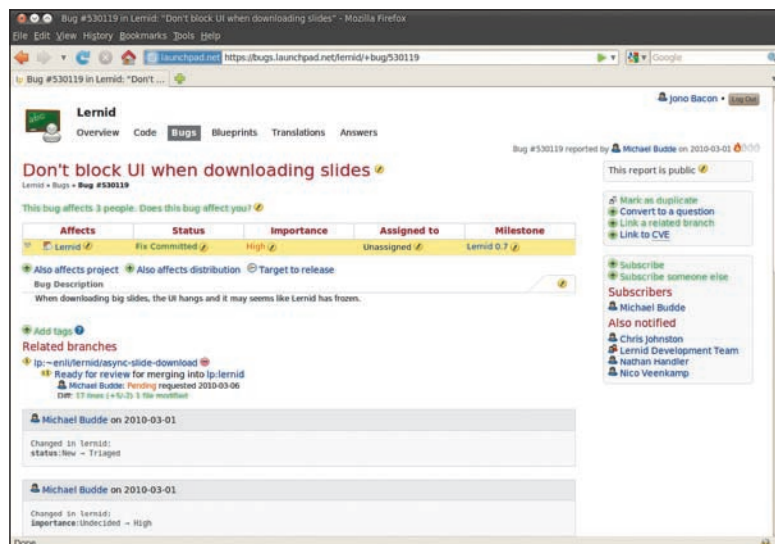
Другая важная функция Launchpad – система отслеживания ошибок [bugs], прозванная разработчиками «Malone» (помните Bugsy Malone? [герой популярного детского мюзикла, – прим. ред.]). Разрабатывать такие инструменты архитектурно, но команда Launchpad оказалась на высоте: система удобна как для пользователей (нетрудно составить сообщение об ошибке), так и для программистов (удобно отслеживать и систематизировать ошибки).

Процесс регистрации ошибок прост. Во-первых, перейдите к странице проекта на Launchpad и нажмите кнопку *Bugs* [Ошибки] вверху. Затем нажмите ссылку *Report A Bug* [Сообщить об ошибке] в правом верхнем углу.

У вас запросят аннотацию – однострочное описание ошибки. Например: “*Gwibber* crashes when connecting [Gwibber падает при подключении]” или “*Evolution* is not displaying messages in my Inbox [Evolution не отображает сообщения в папке Входящие]”. Введите описание и нажмите кнопку *Next* [Далее].

«Ваше сообщение присоединится к череде отчётов о проблемах.»

➤ Ещё одна ценная функция: ошибку, исправленную в ответвлении, можно связать с сообщением об ошибке.



➤ Намётки весьма полезны для коллективной разработки проектов новых функций до начала всеобщего кодирования.

На следующем экране Launchpad попытается найти уже имеющиеся сообщения об аналогичной ошибке и выведет результаты поиска. Нажатие стрелки рядом с описанием ошибки покажет детальные сведения о ней. Если «ваша» ошибка уже зарегистрирована, не отправляйте сообщение – нажмите кнопку “Yes, this is the bug I’m trying to report [Да, об этой ошибке я и говорю]”. Система Launchpad отметит, что такая ошибка обнаружена не однажды. Это поднимет «накал» ошибки, указывающий разработчикам на наиболее частые проблемы.

Не найдя «своей» ошибки среди перечисленных, нажмите кнопку “No, I need to report a new bug [Нет, я хочу сообщить о новой ошибке]”. На следующем экране нужно будет ввести подробное описание ошибки. Понадобятся сведения о компьютере, версиях ПО и шагах, вызвавших ошибку. Указав все необходимое, отправьте сообщение в базу данных нажатием кнопки *Submit Bug*

Report [Отправить сообщение об ошибке].

Отбор ошибок

Ваше сообщение присоединится к череде отчётов, документирующих всяческие пробле-

мы с проектами на Launchpad. Всё это задумано как площадка для обсуждения, где выявляются детальные сведения о проблемах, чтобы в итоге снабдить программиста информацией для поиска решения. Процесс сортировки ошибок на основании полученной информации называется «отбором» [triaging].

Работа над отбором ошибок – весьма ценный вклад в открытое ПО. Просмотр отчётов, расстановка приоритетов, тестирование приложений для выявления недостатков, составление дополнительных комментариев – всё это очень помогает приблизиться к решению проблемы. Рассмотрим основные органы «управления отбором», доступные в Launchpad. Отправляя отчёт об ошибке, вы увидите окно с тремя ключевыми зонами. Вверху, в жёлтой таблице, приведены некоторые сведения о состоянии

Ещё о Launchpad: Ответы

Особо популярна функция Launchpad под названием «Ответы». Здесь конечные пользователи задают вопросы, а члены сообщества кооперируются для ответов на них. Ведётся слежение за тем, чтобы ни один вопрос не остался без ответа; есть возможность добавлять пары «вопрос–ответ» к списку FAQ.

Ответы – верный способ расширить круг пользователей проекта, и каждый удачный ответ повышает ценность этого мощного ресурса поддержки.

и важности ошибки; справа перечислены псевдонимы пользователей, подписавшихся на новости об этой ошибке; основная часть окна отводится комментариям, формирующим дискуссию о поисках решения.

Возьмём жёлтую таблицу вверху. Здесь есть несколько полей, значение которых можно изменить с помощью жёлтых кнопок по соседству. Нам интересны три поля; рассмотрим их подробнее. Первое поле – Status [Состояние], отражающее текущее состояние ошибки и ее исправления. Вариантов много.

» **New [Новое]** Отчёт об ошибке поступил на сайт, но разработчик ещё не дал ему оценку.

» **Incomplete [Неполное]** Отчёт об ошибке, о которой разработчик запросил дополнительные сведения и ожидает их.

» **Invalid [Необоснованное]** Сообщение об ошибке не подтвердилось (иногда пользователи поднимают ложную тревогу).

» **Won't Fix [Не принято]** Ошибка, которую разработчик не считает нужным исправлять (например, пользователь расценил как ошибочное решение изменить цвет кнопки).

» **Confirmed [Подтверждено]** Разработчик признал сообщение достойным внимания.

» **Triaged [Отобрано]** По сообщению собрана вся информация, необходимая для исправления (например, удалось найти причину неполадки).

» **In Progress [В работе]** Разработчик взялся устранить ошибку и занимается этим.

» **Fix Committed [Исправлено]** Исправление ошибки завершено и выгружено в главную кодовую базу.

» **Fix Released [Выдано]** Для ошибок, исправление которых выдано конечным пользователям (чаще всего в очередной версии ПО).

Следующее поле – Importance [Важность]: это весьма важная часть сообщения об ошибке, и работать с ним следует очень внимательно. Значение этого поля указывает на опасность ошибки. Варианты выбора такие:

» **Critical [Критично]** Серьёзная ошибка, вызывающая существенное снижение производительности или связанная с риском потери данных.

» **High [Высокий уровень]** Разрушительная ошибка.

» **Medium [Средний уровень]** Ошибка, вызывающая значительные неудобства в работе с ПО.

» **Low [Низкий уровень]** Ошибка всерьёз не влияет на общую работу приложения.

» **Wishlist [Пожелания]** Многие сообщения об ошибках можно считать запросом на новые функции – и классифицировать соответственно.

» **Undecided [Неизвестно]** Выберите этот вариант, если ни один из предыдущих не подошёл.

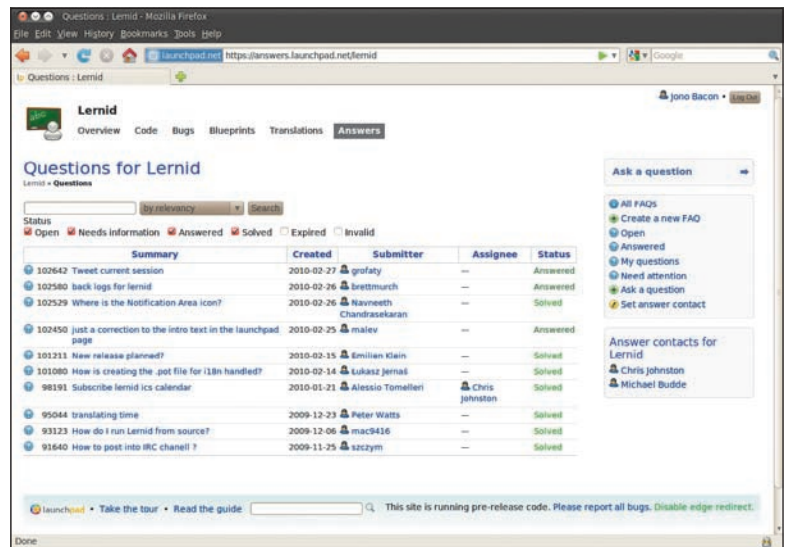
Наконец, последнее важное поле – Assigned To [Поручено]: здесь приводится имя разработчика, который взялся за поиск решения.

Под жёлтой таблицей отведено место для переговоров об ошибке. К участию приглашаются все, обсуждение отображается в хронологическом порядке. Полезный совет: здесь не нужно помещать комментарии типа «эту ошибку я тоже видел». Для этого в верхней части окна есть ссылка Does This Bug Affect You? [Влияет ли эта ошибка на вас?] – нажмите жёлтую кнопку справа от надписи, и все поймут, что да, влияет.

Наконец, если вас особо интересует отчёт о конкретной ошибке и вы хотите следить за изменением его статуса, нажмите кнопку Subscribe [Подписаться] на правой панели – и будете получать электронные сообщения об обновлениях и изменениях.

Переводы

И последний компонент, который мы рассмотрим – секция переводов, известная как Rosetta. Сюда можно импортировать любой



проект, использующий стандартный каркас gettext, и Rosetta обеспечит простой интерфейс для любого желающего заняться переводом. Приятно, что для участия в проекте переводчику незачем изучать программирование: достаточно уметь переводить с английского.

Сообщество переводчиков на Launchpad весьма внушительно. Я выгрузил на Launchpad готовое приложение, подал заявку на перевод, и уже через час были готовы переводы на 10 языков.

Чтобы принять участие, найдите проект, где включены переводы (например, Ubuntu), и нажмите кнопку Translations [Переводы]. Затем нажмите кнопку Choose Preferred Languages [Выбрать языки] и отметьте галочками языки, на которые хотите перевести. Вернувшись на главную страницу переводов, перейдите к отмеченному языку и в первом комбинированном окне выберите Untranslated Items [Непереведённые сегменты]. Теперь введите перевод в поле New Translation [Новый перевод]. Добавляя перевод, не забывайте сохранять добавочные знаки и форматирование.

В завершение

Launchpad – мощная среда разработки, и наш обзор лишь задел верхушку айсберга. За каждым из рассмотренных нами основных компонентов скрывается богатый набор функций и возможностей для коллективной разработки отличного ПО. В продолжение нашего путешествия рекомендую зайти на страницу <https://edge.launchpad.net/+tour>: узнаете много нового. Удачи, и не забывайте извещать нас о своих достижениях! **LXF**

» В рубрике «Ответы» ваши пользователи превращаются в специалистов службы поддержки!



Рабочий стол как IDE

Кроме мощного web-интерфейса, в Launchpad предусмотрена библиотека Python, с помощью которой создано несколько других интерфейсов (для того же Launchpad). Один из наиболее интересных – *Ground Control*, написанный Мартином Оуэнсом [Martins Owens] – доступен по адресу <http://ground-control.org>. *Ground Control* обеспечивает интеграцию Launchpad и *Bazaar* в файловый менеджер *Nautilus* (Gnome).

Ground Control добавляет несколько кнопок к стандартному окну *Nautilus*, что позволяет просматривать ответвления, выгружать изменения, исправлять ошибки и выполнять другие действия – не затрагивая командной строки. Изначально *Ground Control* был создан для содействия группе разработки нетехнической документации Ubuntu, а затем, в сочетании с web-интерфейсом Launchpad, перерос в мощное средство коллективной разработки проектов.

Hugin



Идеальные панорамы

Больше – значит, лучше! Заезжий преобразователь реальности Натан Сэндерс научит вас сшивать дюжину фотографий в одну большую с помощью *Hugin*.

Картинка стоит тысячи слов, а хорошая панорама достойна тысячи картинок. Чудесные пейзажи просто не поместятся в видоискатель вашей камеры. Если нельзя целиком вместить гору в рамки фотографии или за деревьями не видно леса, вас выручит кросс-платформенное приложение *Hugin* для сшивания нескольких снимков в одно изображение. Используйте его для увеличения разрешения вашей камеры и для объединения нескольких фотографий.

Активно разрабатываемый с 2003 года, *Hugin* – это интерфейс *wxWidgets* к подборке открытых инструментов для работы с панорамными фотографиями. Приложение идет под лицензией GPL и доступно для компиляции во многих дистрибутивах Linux, BSD, Windows и Mac OS X.

Создание фотографий

Достаньте из закромов вашу старенькую цифровую фотокамеру, установите *Hugin* с **LXF DVD** и следуйте инструкциям нашего урока. *Hugin* способен сшить любое количество перекрывающихся изображений, но вы добьетесь лучшего результата, придерживаясь следующих правил:

» **Перекрывайте смежные фото** примерно на 30%, чтобы иметь в распоряжении достаточные куски фотографий с одинаковыми элементами для упрощения дальнейшего совмещения фотографий. Поскольку края ваших фотографий могут быть подвержены абберациям, большие перекрытия улучшат равномерность.

» **Двигайте камеру** от снимка к снимку как можно меньше, для уменьшения ошибки на параллакс в финальном панорамном снимке. Не бойтесь снимать с рук: обычный штатив, возможно, и не поможет, поскольку ваша цель – вращать объектив камеры, а не саму камеру.

» **Не скромничайте** с полем обзора. Вероятнее всего, потом вам захочется обрезать свою панораму, поэтому убедитесь, что целевые объекты умецаются внутри кадра.

» **Тщательно выбирайте время для снимка**, поскольку в перерыве между кадрами объекты могут двигаться, а это чревато неприятностями. Качающаяся ветка дерева вызовет небольшой сдвиг, если использовать ее как точку привязки, а проезжающая мимо машина угодит на все кадры панорамы.

» **И даже хотя** процесс сшивания в *Hugin* автоматически корректирует ошибки фотографа-любителя, вы минимизируете «рубцы» между фотографиями с помощью ручного режима вашей камеры, фиксируя баланс белого и выдержку для всех снимков.

Hugin работает с JPEG и TIFF форматами выходов различных камер, а также с PNG- и HDR-файлами. Если вы снимали в RAW, сперва придется произвести конвертацию.

Чтобы познакомить вас с *Hugin*, мы покажем его возможности на примере двух панорам. Первая создается просто из последовательности снимков в один ряд, а вторая – «малая планета», совмещающая три ряда изображений, что потребует немного больше виртуозности.

Панорамные проекции

Помните, как вы в детстве, глядя на карты в проекции Меркатора, заключали, что Гренландия примерно того же размера, что и Южная Америка? Неправильная проекция может стать причиной искажения составляющих панорамы, поэтому прокрутите проекции в окне предпросмотра *Hugin* и выберите с умом! Проекция по умолчанию, сферическая [equirectangular], просто отображает широту и долготу на оси координат $x-y$ – она обычно применима

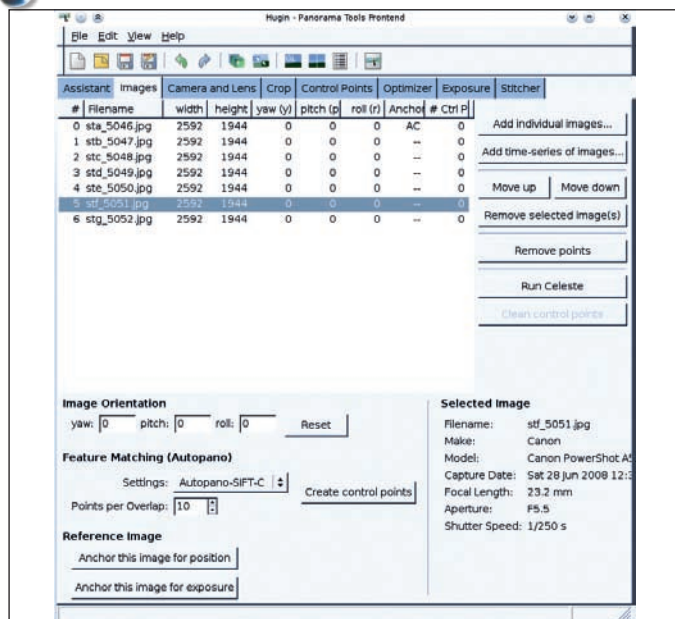
к длинным вытянутым панорамам. Прямолинейная [rectilinear] проекция – это то, что получается в камере-обскура, и подобно восприятию глаза, но здесь края вытянуты. Проекция «рыбий глаз» [fisheye] создает эффект как в одноименных объективах. Панорамы-панини [equirectangular panini] сохраняют вертикальные и радиальные линии, создавая ощущение перспективы в снимках архитектуры.



» Вот часть из 19 панорамных проекций *Hugin*, примененных к 120-градусному изображению. Сверху вниз: сферическая, прямолинейная, «рыбий глаз» и панини.

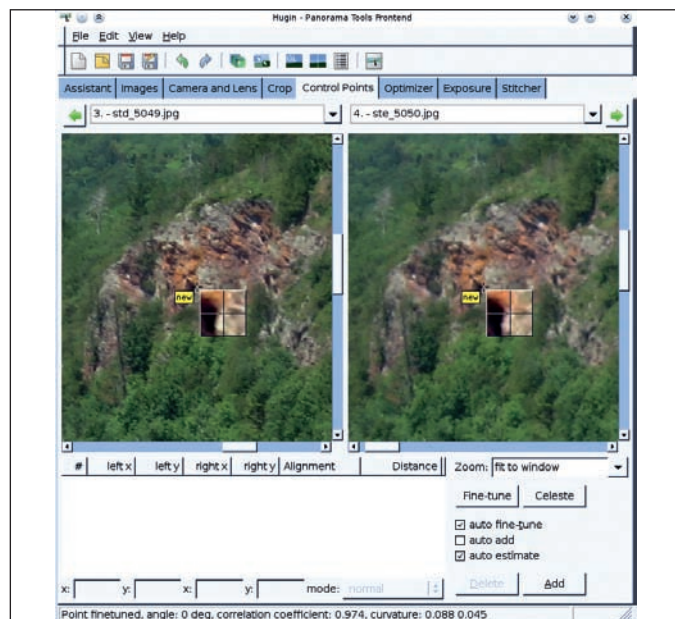


Шаг за шагом: Простая панорама



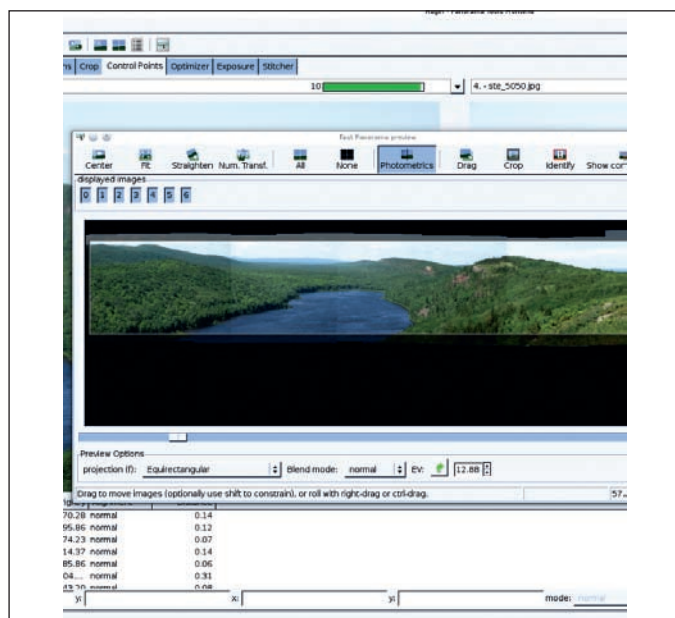
1 Загрузите ваши снимки

Для начала, откройте *Hugin*, перейдите на вкладку Images [Изображения] и перетащите туда ваши снимки. Ваша задача — задать в *Hugin* сортировку ваших фотографий и упорядочить их так, как они были в реальности. Для этого программа должна знать общие места на ваших снимках — контрольные точки. Кто не гнушается патентами, найдёт немало инструментов, легко интегрируемых с *Hugin*, для автоматической генерации контрольных точек. Если вы установили один из них, просто нажмите Create control points [Создать контрольные точки].



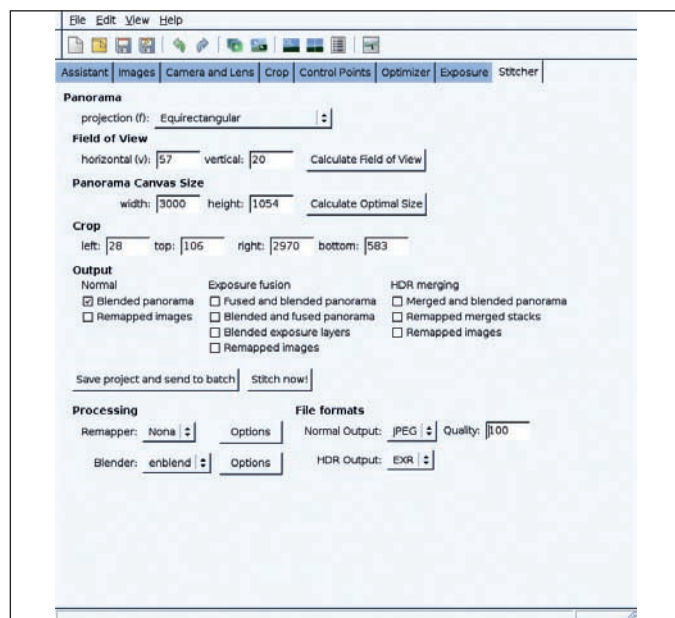
2 Совмещение фотографий

Чтобы определить контрольные точки вручную, перейдите на вкладку Control Points [Контрольные точки]. Через выпадающее меню загрузите два смежных фото, затем щелкните на идентичных элементах обеих фотографий и нажмите Add [Добавить] для создания контрольной точки. Желательно создать хотя бы по пять контрольных точек на пару изображений, перекрыв возможно больший участок. По завершении, перейдите на вкладку Optimizer [Оптимизировать], выберите Positions [Позиции] (указываемые приращениями, начиная от якорной позиции) из выпадающего меню, а затем Optimize Now [Оптимизировать сейчас].



3 Глянем на результат

Нажмите кнопку Fast Preview [Быстрый предпросмотр] на панели инструментов для того, чтобы посмотреть панораму, в которую объединились ваши фото. Если вы недовольны увиденным, нажмите на кнопку Straighten [Выровнять] на панели инструментов Быстрого предпросмотра для автоматического задания центра, увеличьте и выровняйте ряд выбранных снимков. Мы использовали кнопку Crop [Подрезать] для подчистки рваных краев панорамы. Чтобы посмотреть, хорошо ли выровнены ваши изображения, выберите режим Difference [Отличия] из выпадающего меню Blend [Смешение] внизу окна и наведите мышь на область перекрытия. Если изображения хорошо наложились друг на друга, эта зона будет черной, а если она разноцветная — есть неточности.

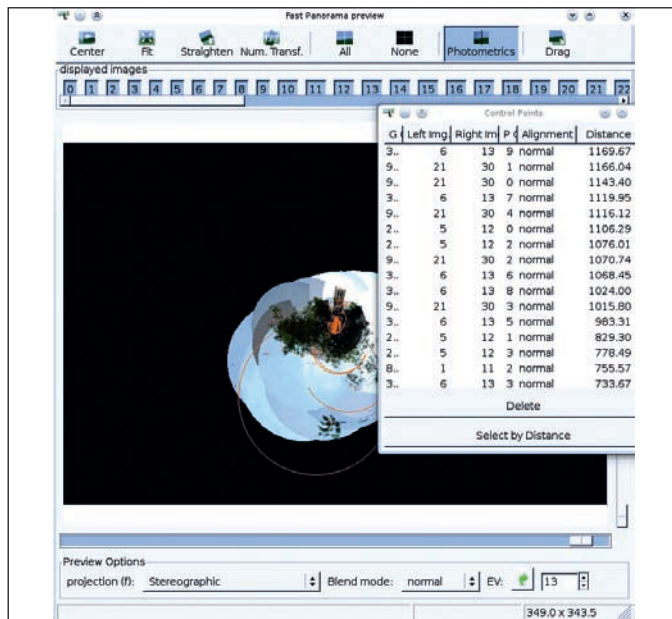
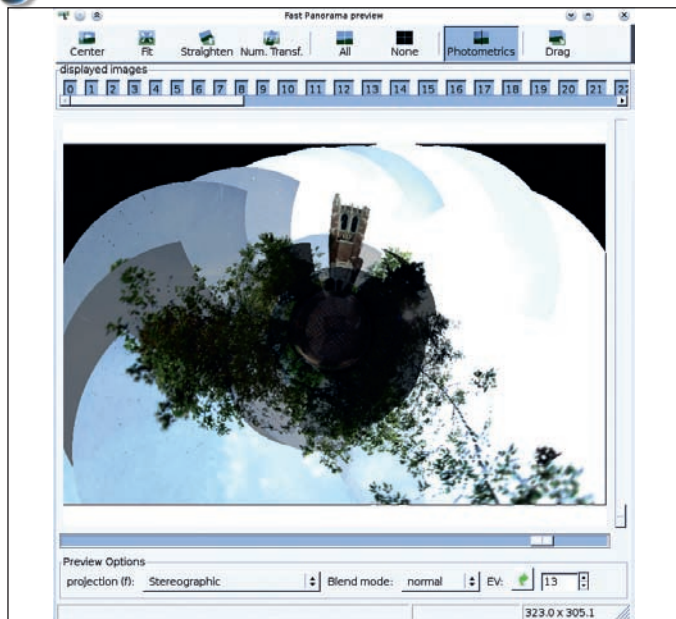


4 Последний штрих

Даже при идеальном наложении, переходы могут быть заметны — из-за разных настроек баланса белого и экспозиции. Для поправки, перейдите на вкладку Exposure [Экспозиция] и выберите предустановку 'Low Dynamic Range, Variable White Balance' («Узкий динамический диапазон, варьируемый баланс белого») в выпадающем меню и нажмите Оптимизировать сейчас. Для предпросмотра результатов в окне, включите Photometrics [Фотометрики] на панели инструментов. Наконец, перейдите на вкладку Stitcher [Сшивание], нажмите Calculate Optimal Size [Рассчитать оптимальный размер], и задайте разрешение. На выбор в File Formats [Форматы файлов] — JPEG, PNG и TIFF с различной степенью сжатия. Удовлетворившись результатом, нажмите Stitch Now [Склеить Сейчас].



Шаг за шагом: Малая планета

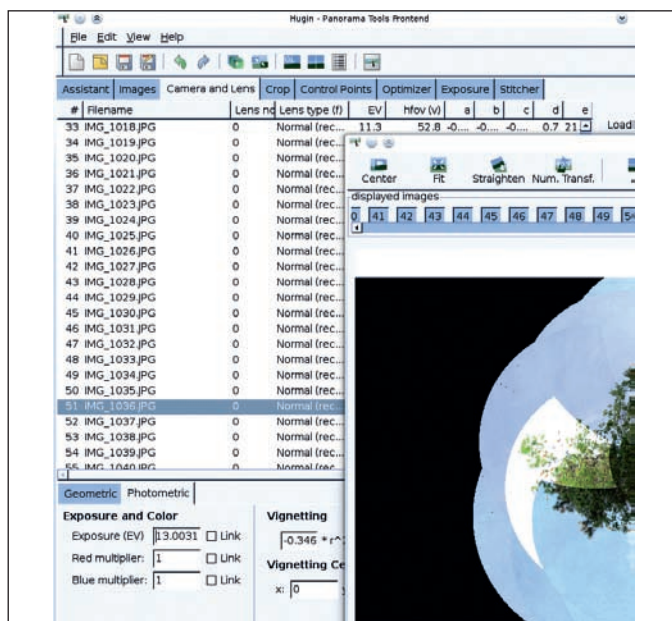


1 Стереографическая проекция

Начните с загрузки ваших изображений и нахождения контрольных точек. Поскольку у нас три ряда накладывающихся изображений, потребуется как минимум 1000 контрольных точек. Не будь на свете автоматического генератора, времени на это ушло бы немало. Затем перейдите к Позициям (указываемым приращениями, начиная от якорной) во вкладке Оптимизировать. Откройте Быстрый предпросмотр и выберите из меню проекций Stereographic [Стереографическую]. Кусок вашей ноги посреди планеты убежден потом, с помощью GIMP или другого постпроцессора.

2 Плохие контрольные точки

Нажмите на кнопку Show Control Points [Показать контрольные точки] в меню Предпросмотр, чтобы посмотреть связи между изображениями. Оранжевые точки — контрольные, а длинные оранжевые линии являются результатом ошибок в контрольных точках. Вернитесь в главное окно Hugin и нажмите на кнопку Показать контрольные точки, затем кнопку Select By Distance [Выбрать по расстоянию] в диалоговом окне. Попробуйте задать расстояние 100 и нажмите ОК, затем Удалить. Избавление от плохих контрольных точек улучшит объединение изображений в панораму.



3 Оптимизируем все!

Если в вашей панораме есть длинные прямые линии (например, углы зданий), можно заставить Hugin сохранить их. Перейдите на вкладку Контрольные точки и найдите окончания линий на изображениях. Щелкните на каждое из них, нажмите Добавить и выберите вертикальные или горизонтальные в меню внизу. Затем перейдите на вкладку Оптимизировать, выберите Everything [Все] из выпадающего меню и нажмите Оптимизировать сейчас. Перейдя к таблице контрольных точек, задайте сортировку по расстоянию. Определите уровень отсечки по расстоянию для отклоняющихся точек (где-то 30 или ниже). Удалите эти точки и нажмите Re-optimize [Пере-оптимизировать] на панели инструментов. Перейдите на вкладку Выдержка и нажмите Оптимизировать сейчас.

4 Фотометрические штрихи

Откройте Быстрый предпросмотр и нажмите Фотометрики на панели инструментов. Если какой-то цвет отсутствует на фотографии, попробуйте редактировать это вручную с помощью фотометрических параметров. Кнопкой Identify [Идентифицировать] на панели инструментов окна предпросмотра определите номер изображения. На вкладке Camera and Lens [Камера и Объектив] выберите нужное изображение и откройте вкладку Фотометрики внизу. Подредактируйте Выдержку (EV) для изменения яркости, а индексы Красного и Синего — для выравнивания цвета. После этого откройте вкладку Сшивание и выберите размер изображения на выходе. Можно задать и полное разрешение исходных фотографий, но приготовьтесь, что на это уйдет часы, даже на мощной машине.

Сшивание фотографий увеличивает охват вашей камеры, но игра не стоит свеч, если заодно не расширить цветовой спектр. Широкий динамический диапазон (high dynamic range, HDR) фотографий позволяет отражать одновременно детали снимка, находящиеся и в тени, и на ярком солнечном свете. Попробуйте заглянуть в руководство пользователя к вашей камере и выяснить, как вручную менять выдержку ваших снимков. Каждую фотографию, предназначенную для создания HDR-панорамы, старайтесь делать с тремя разными выдержками.

Для сшивания HDR-панорамы с помощью *Hugin* вам потребуется поменять всего лишь несколько шагов. Перейдя на вкладку Выдержка, выберите High Dynamic Range, Fixed Exposure [Широкий динамический диапазон, фиксированная выдержка] в выпадающем меню. Для проверки результата, нажмите кнопку Предпросмотр панорамы на панели инструментов. Это происходит медленнее, чем ускоренный в OpenGL Быстрый предпросмотр, но внизу окна из выпадающего меню вы можете увидеть HDR-предпросмотр. Учтите, предпросмотр HDR использует простое логарифмическое изображение для проверки областей с сохраненными деталями – оно хуже окончательной панорамы.

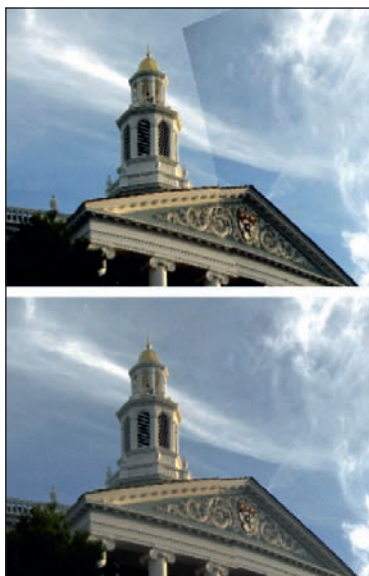
Последний шаг в создании HDR-панорамы заключается в отметке опции Слитая и Сшитая Панорама в HDR-Слиянии на вкладке Сшивание и нажатии Сшить сейчас. В результате получится файл OpenEXR, включающий информацию по цветовым настройкам ваших фотографий на каждом уровне выдержки. У этого изображения, вероятно, гораздо больший динамический диапазон, чем у монитора или принтера, и его нельзя посмотреть без обработки. Откройте HDR-панораму в *Luminance HDR*, кросс-платформенной программе открытого кода для подбора тонов для конвертации ее в JPEG или другой файл с узким динамическим диапазоном (low dynamic range, LDR). Подбор тонов включает назначение цветов пикселей изображения в соответствии с алгоритмом обработки. Поэкспериментируйте с алгоритмами для получения самой реалистичной репродукции снимка, или постепенно увеличивайте контрастность для рендеринга парада цветов.

Веселенькие цвета

Даже если вы не сумели сделать снимки с различными выдержками, *Hugin* восполнит этот пробел, предлагая большой спектр цветов. Просто выберите Fused And Blended Panorama [Комбинированная и сшитая панорама] под Exposure Fusion [Комбинация выдержек] на вкладке Сшивание и нажмите Склеить сейчас. Запустится *Enfuse*, инструмент для работы с выдержкой от *Hugin*. *Enfuse* не соединяет исходные изображения в HDR, а выделяет области с точки зрения лучших параметров по выдержке в процессе объединения изображений.



➤ Сверху вниз: обычная панорама, панорама *Enfuse*, HDR-панорама, HDR-панорама с высококонтрастным подбором тонов. Справа — детали фрагменты.



➤ *Hugin* корректирует изображения, чтобы они сошлись.

Если даже одно из изображений недодержано или передержано, а соседний снимок выглядит лучше, *Enfuse* сохранит максимум деталей.

Представьте себе создание панорамы вручную: вы выкладываете несколько отпечатанных снимков на столе и сдвигаете их для соединения в панораму. Этим базовым приемом достаточно, чтобы создать панораму, но *Hugin* умеет много больше. Во-первых, это коррекция геометрии. Выбрав Все во вкладке Оптимизация, вы велите *Hugin* выровнять изображения, а не просто приткнуть их друг к другу в нужном порядке. Сначала *Hugin* определяет область зрения из ваших изображений, для чего ранее использовались Exif-данные. Затем *Hugin* корректирует искривления объектива с помощью полиномиальной модели третьего порядка. Наконец, *Hugin* определяет вектор, указывающий, на сколько пикселей центр CCD-матрицы отклоняется от центра объектива (оптической оси). Перед объединением в панораму все изображения корректируются согласно этим параметрам.

С исправлениями или без, *Hugin* убирает многие ошибки выравнивания посредством модуля расширения Enblend, сглаживающего переходы между соседними изображениями по сплайновому алгоритму. Сначала он определяет примерную линию перехода, где-то посреди области перекрытия, приложив все усилия, чтобы избежать отличающихся элементов на фотографиях (например, проезжающей машины). Затем он соединяет изображения вдоль этой линии, используя широкую кисть для сглаживания в плавно меняющихся областях (например, небо), а узкую кисть — в областях с резкими изменениями (скажем, трава).

Но и этого мало, чтобы финальная панорама была идеальной: цвета также должны совпадать. *Hugin* корректирует фотометрику пересмотром выдержки всех изображений для оптимизации уровней яркости. Затем *Hugin* изучает градиенты яркости накладываемых зон и определяет три условия виньетной коррекции. Если вы выберете оптимизацию переменного баланса белого на вкладке Экспозиция, *Hugin* также подсчитает индексы красного и синего для каждой фотографии. И, наконец, оптимизация кривой отклика камеры нормализует цветовое пространство.

Итак, склеивайте снимки в панораму как профессионал — камера будет вашей кистью художника! Панорамная фотография и *Hugin* помогут запечатлеть вид любой грандиозности. LXF



➤ Продвинутой геометрической коррекция *Hugin* отлично выровняла колонны.

Патенты на ПО: Надо ли их бояться?

Патенты – неизменно горячая тема в Linux-сообществе, но, как показывает практика, далеко не всякий знает, как обстоят дела с ними на самом деле. Павел Протасов исправит этот недостаток.

Наш эксперт

Павел Протасов

Дипломированный юрист и практикующий журналист, далеко не чуждый вопросов применения свободного ПО в России.

Авторское право защищает форму произведения, поэтому программисту «обойти» его довольно легко, просто самостоятельно написав тот код, который реализует нужные функции. А вот патентная защита распространяется на содержание изобретения: патент защищает те процессы и методы, которые в нем описаны, независимо от формы.

Поэтому тема «софтверных патентов» весьма важна для сообщества разработчиков открытого ПО. Последняя версия GNU GPL посвящает им целый раздел.

Можно ли патентовать?

Зачастую патенты путают с ноу-хау, считая, будто их основная задача – скрывать от окружающих сущность изобретения. Это в корне неверно: предназначение заявки на патент – как раз описание изобретения: автор должен в явной форме отграничить свою «интеллектуальную собственность» от чужой.

Правда, иногда описания патентуемого настолько запутаны, что с трудом можно понять, о чем идет речь. Как вам, к примеру, описание предмета, который «содержит по крайней мере в одном из поперечных сечений линию границы наружной и внутренней сторон и отличается тем, что по крайней мере на границе наружной и/или внутренней стороны часть длины линии границы сечения выполнена в виде фрагмента или комбинации фрагментов косоугольного сечения прямого кругового конуса»? Пока вы читаете эту статью, можете попытаться догадаться, что же это описано, а разгадку я, как и положено, дам в самом конце...

Патентование программ актуально прежде всего для США. Правда, недавно там случился прецедент, с которым многие связывали возможность существенного ограничения такого патентования. Речь идет о так называемом «деле Билски» («*Bilski vs Karpos*»).

Суть дела заключалась в следующем. Некто Бернард Билски попытался запатентовать метод страхования рисков при торговле на бирже. Это не программа, однако при патентовании таких абстракций возникают похожие проблемы.

Сначала Билски получил отказ от патентного ведомства, со ссылкой на то, что его «изобретение» представляет собой непатентуемую «абстрактную идею». Затем отказ на этих же

основаниях повторил суд, а вот когда Билски стал обжаловать и это решение, произошло самое интересное. Апелляционный суд вновь отказал ему, но уже с другой формулировкой: по его мнению, для того, чтобы быть запатентованным, процесс должен быть связанным с определенным материальным устройством, либо преобразовывать в другое состояние какие-нибудь предметы или материалы.

Это решение сторонниками свободного ПО было встречено радостно: оно налагало ограничения на патентование не только бизнес-методов, подобных «страхованию рисков», но и «чистого» программного обеспечения, не связанного с каким-либо устройством. Радость их усилилась после того, как настойчивый Билски, продолжая обжаловать решение, потерпел неудачу в Верховном суде: как мы помним, право в Соединенных Штатах Америки прецедентное, и это решение становилось обязательным для судов нижестоящих.

Но вот оно было опубликовано – и радость сменилась разочарованием: Верховный суд не повторил выводов апелляционного суда об «устройстве или преобразовании», а фактически вернулся к позиции патентного ведомства и суда первой инстанции, назвав страхование Билски «абстрактной идеей». Решающего слова

в деле об ограничении «софтверных патентов» сказано все-таки не было...

Если мы посмотрим на отечественное законодательство, то увидим, что в нем есть прямой запрет на патентование ПО,

«Решающего слова в деле о “софтверных патентах” не сказано.»

в статье 1350 четвертой части Гражданского кодекса. Однако она говорит о патентовании изобретений, а кроме них, патент можно получить еще на полезную модель и промышленный образец.

Промышленный образец – это «художественно-конструкторское решение изделия ... , определяющее его внешний вид». В этом качестве можно запатентовать, к примеру, интерфейс программы. А под «полезной моделью» закон понимает «техническое решение, относящееся к устройству» – так может быть запатентована прошивка какого-нибудь устройства. (Далее «запатентованный объект» для простоты я буду называть «изобретением», понимая под этим словом также и промышленный образец с полезной моделью).

Если мы вернемся к изобретениям, то легко увидим, как запатентовать в таком качестве и программу. Изобретение – это «техническое решение в любой области, относящееся к продукту



» Запасливый «полезномодельщик» застолбил цифровую передачу данных через Интернет — вот и изобретай потом интернет-радио...

(в частности, устройству, веществу, штамму микроорганизма, культуре клеток растений или животных) или способу (процессу осуществления действий над материальным объектом с помощью материальных средств)». В этой формулировке прослеживаются явные параллели с «устройством или преобразованием» из «дела Билски». В США, кстати, из-за этого дела появилась формулировка «machine or transformation», которую используют для обозначения соответствующего критерия патентоспособности.

Так вот, если «голую» программу запатентовать нельзя, то почему бы не связать ее с каким-либо «устройством» или «процессом осуществления действий»? Да без проблем...

Патентование «процесса» вызвало один из самых «громких» отечественных патентных конфликтов, с участием компании «Суперфон», занимавшейся продажей рекламы в сетях мобильной связи. Основал ее бывший работник компании «Гигафон» Олег Тетерин. Рекламные объявления показывались абоненту с помощью специально написанной программы при получении обычных текстовых сообщений. А чтобы тот ее смотрел, сотовым оператором предоставлялись за это разные мелкие бонусы.

Однако эта модель до боли напоминала способ распространения рекламы, который применял бывший работодатель Тетерина, «Гигафон». Начались взаимные обвинения в нарушении патентов, дошло до возбуждения уголовного дела, а поскольку Тетерин не являлся к следователю, ему пришлось даже посидеть под арестом. В апреле 2009 года суд признал его виновным в нарушении патентных прав и приговорил к штрафу. Позже был удовлетворен иск, поданный «Гигафоном» в арбитражном суде.

«Гигафон» получил на «метод и систему для публикации рекламных сообщений» евразийский патент № 007757, который действителен и в России. Тетерин получил два российских патента — № 2322702 и № 2346413, на «способ распространения рекламно-информационных изображений». Патенты Тетерина не были признаны недействительными, но от ответственности его это, как мы видим, не спасло...

Собственно, вот и ответ на вопрос, как запатентовать программу: косвенно, через «метод» или «способ» совершения действий, которые могут быть выполнены только программой. »

А есть ли в России «патентные тролли»?

Историй о том, как кто-то на кого-то подал в суд из-за патента, в отечественные СМИ попадает достаточно, однако большинство из них касается событий, происходящих за границей, чаще всего — в США. Благодаря этому в русский язык проник тамошний термин «патентный тролль»: так на Западе называют компании, скупающие патенты не для того, чтобы что-то производить, а для последующей подачи исков.

В России тоже есть такие «хронические истцы»; однако если к ним присмотреться, то становится заметным одно важное отличие. Подавляющее большинство русских патентов, которые стали предметом разбирательства, получены не на изобретения, а на так называемые «полезные модели». Из-за этого в кругах отечественных юристов такие истцы носят название «полезномодельщиков».

Отличие полезных моделей от изобретений — в процедуре патентования: при выдаче патента на изобретение проверяется, обладает ли оно так называемым «изобретательским уровнем», то есть не является ли оно очевидным при существующем на момент патентования уровне техники. Если вы читаете какую-нибудь заметку об оче-

редном суде из-за патента, можно легко определить, о чем идет речь: у патентов на полезные модели номера пятизначные, для изобретений они семизначные.

При патентовании полезной модели такая проверка на «уровень» не проводится, и в результате патент можно получить на что угодно (нет, не подсматривайте, пожалуйста, в конец статьи). При этом ответственность за нарушение любого патента одинакова, включая уголовную. Суд не может признать патент недействительным, для этого есть специальная процедура. Это, разумеется, на руку «троллям».

Отличие наших «троллей» от западных — в том, что изобретения у последних подвергаются проверке на наличие «изобретательского уровня», и успешно ее проходят. Правда, иногда для получения патента используется умышленно запутанное описание (и все-таки: не подсматривайте).

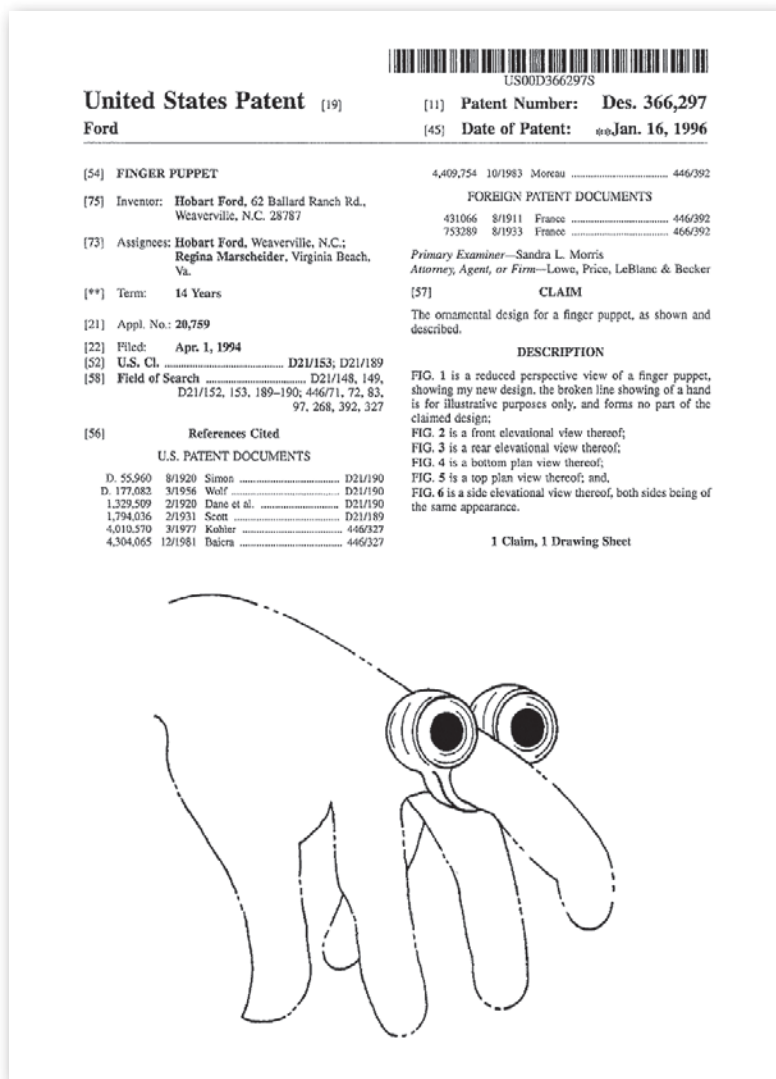
Иногда изобретение остается идеей и не воплощается в жизнь просто из-за того, что существующие технологии не позволяют его реализовать. Время идет, технологии развиваются, идеи носятся в воздухе — в том числе и та, запатентованная. В один прекрасный день

она приходит кому-то в голову, потом еще кому-то, потом ее реализуют, потом — начинают применять повсеместно...

Вот тогда-то и извлекается из-под сукна патент. То, что в нем описано, может быть известным всем, но тривиально запатентованное изобретение только сейчас, когда подается иск.

Один из показательных примеров — иск техасской компании ІРАТ к производителям антивирусов, включая «Касперского» (см. «Ссылки», пункт 3). Подан он был в прошлом году, за нарушение двух патентов, выданных аж в середине девяностых. За это время запатентованная методика ограничения действий, выполняемых программой, стала общепринятой, и в ответчики по иску попали все крупные производители антивирусов.

В Штатах, с давней традицией патентовать любую идею, таких патентов можно найти в количестве, достаточном для того, чтобы засудить кого угодно. В России такой традиции нет, и отечественные «тролли» для подачи исков патентуют какое-нибудь колесо как полезную модель. Поэтому все-таки будет правильнее называть их именно «полезномодельщиками».



➤ **Американский патент № 366 297 выдан на «куклу из пальцев».**

Любой программный продукт, который будет выполнять такие действия, нарушит патент.

Нужно ли патентовать?

Я понимаю, что для ценителей свободного ПО разговор о необходимости «софтверных патентов» – это как красная тряпка для быка, но все же попробую...

Движение по разработке СПО зародилось и наиболее интенсивно развивается в США, где программы патентуются – вот такой парадокс... Может быть, патенты не так уж страшны для разработчиков? Почему бы не допустить охрану ПО именно с их помощью?

Единственное условие – сделать программы не охраняемыми авторским правом. Если по-хорошему, то они и не должны так охраняться. По отечественному законодательству программы приравнены к «литературным произведениям», что, разумеется, есть грубое допущение. Сделано это было для того, чтобы «подогнать» ПО под охрану Бернской конвенции, которая была написана более ста лет назад.

Разумеется, ни о каких программах в ней не могло идти и речи: говорит она только об охране «литературных и художественных произведений». Но, пользуясь тем, что список таких произведений в ее тексте не исчерпывающий, российские (да и не только российские) законодатели «протащили» под ее охрану еще и ПО, чтобы не нужно было принимать для него отдельное соглашение.

Плоды этого допущения мы и пожинаем: вдобавок к охране авторским правом предприимчивые люди де-факто распространили на программы еще и патентную охрану. Парадокс в том, что эта охрана лучше подходит для ПО, чем копирайт, который очень плохо сочетается с теми объективными законами, которым подчиняется разработка программного обеспечения. Ведь программа – это не литературное и не художественное произведение. Программирование – это не творчество, это работа такая...

Авторские права возникают в силу создания произведения; никаких формальностей для этого выполнять не надо. Для получения патента нужно подать заявку, заплатить за нее и подождать некоторое время. Все это и проверка изобретений на наличие «изобретательского уровня» позволяет отсеять совсем уж ненужный шлак, а вот копирайтом он охраняется все равно.

Патент, будучи выданным, действует только на определенной территории, а копирайт – практически всемирно, за исключением отдельных стран-маргиналов, не присоединившихся к соответствующим международным договорам.

Авторским правом охраняются не только произведения целиком, но и их части, которые могут быть признаны самостоятельным результатом творческого труда. Если в случае с обычными произведениями такая охрана чаще всего предоставляется персонажам, то в случае с программой отдельно использована в другом ПО может быть чуть ли не любая функция или библиотека. Это – потенциальный источник проблем при конфликте с автором,

Когда патент нельзя нарушить

Если вы пользуетесь зарубежными дистрибутивами Linux, то вам хорошо знакома стадия «установки кодеков», которой они должны подвергаться сразу после инсталляции. К примеру, сторонники Ubuntu, которые вообще не привыкли к лишним движениям, за что слывят «виндузяниками от линукса», придумали для этих целей специальный пакет *ubuntu-restricted-extras*. Его установка «тащит» за собой кучу других пакетов, после чего наконец-то становится можно проигрывать MP3-файлы, смотреть кино в формате Windows Media, DVD, Flash и много всего другого...

Патент, как уже было сказано в статье, действует только на определенной территории: американский – исключительно в США, российский – в России. Существует процедура получения международного патента, который

действителен в нескольких странах. А поскольку в США наиболее развито не только дистрибутивостроение, но и патентование программ и алгоритмов, то из-за американских заморочек страдать приходится пользователям со всего мира. Раз разработчики не имеют права распространять запатентованные программы, их приходится скачивать отдельно, с серверов, расположенных вне американской юрисдикции. Или пользоваться модификациями дистрибутивов, сделанными в странах, в которых соответствующие алгоритмы не охраняются.

Самое главное я по традиции скажу в конце: дело в том, что наше законодательство (да и не только наше) разрешает использование запатентованного изобретения для удовлетворения личных, семейных, домашних, а так-

же иных потребностей, не связанных с извлечением прибыли. Так что использовать «для дома, для семьи» можно все что угодно и не бояться: патент вы этим нарушить не можете. Патенты часто называют «промышленной собственностью»; так вот, она на то и промышленная, чтобы (не) применяться в промышленных масштабах.

Помимо этого, предусмотрен еще целый ряд подобных исключений. Например, можно использовать изобретение при чрезвычайной ситуации; однако потом необходимо сообщить об этом патентообладателю и выплатить ему соразмерную компенсацию. Аптеки могут использовать изобретение для разового изготовления лекарственных средств по рецептам. Кроме того, оно может быть использовано при научных исследованиях и экспериментах.



› Смайлики запрещены к использованию в статьях LXF, и мы можем быть спокойны: когда Microsoft получит на них патент (заявка 20050156873), мы его не нарушим.

который, передав заказчику права на программу целиком, может предъявить претензии уже по поводу каждой «самостоятельной» ее части, творческий характер которой будет презюмироваться, пока не доказано обратное.

Еще один потенциальный источник претензий – так называемое «право на неприкосновенность произведения». Впервые в отечественном законодательстве оно появилось еще при СССР – правда, куда-то исчезло из закона «Об авторском праве...», но в ГК его вернули. На практике «нарушение неприкосновенности» отличается от переработки тем, что при переработке создается «новое произведение». Чем оно отличается от «не нового» – этот вопрос решается в каждом случае индивидуально, при изучении того, что получилось в результате изменений.

«Право на неприкосновенность» закон относит к «личным неимущественным» правам, которые являются неотчуждаемыми: передать их кому-либо невозможно. В результате программист, передав права на ПО заказчику вместе с правом на переработку, все равно сохраняет возможность препятствовать изменениям кода, объявляя их «нарушением неприкосновенности», а не «переработкой».

Патентное право предусматривает так называемую «принудительную лицензию»: если обладатель патента недостаточно использует свое изобретение, то его можно по суду заставить заключить договор о предоставлении прав на такое использование. С литературным произведением это, разумеется, не пройдет, хотя во времена советского «копилефта» такая возможность была.

Ну и, наконец, патентная охрана длится значительно меньший срок, чем охрана авторским правом: не семьдесят лет со дня смерти автора, а всего лишь двадцать лет со дня подачи заявки на патент. Это для изобретений; а для полезных моделей и промобразцов – и того меньше, десять и пятнадцать лет. Для патентов на ПО даже этот срок слишком велик, можно сделать лет пять...

Более того: распространено мнение о том, что российское законодательство не знает такого понятия, как «публичная лицензия», по которой автор произведения может разрешить его использовать не кому-то конкретно, а сразу всем окружающим. А ведь это не так: в статье 1368 Гражданского кодекса есть нечто подобное.

Статья предусматривает так называемую «открытую лицензию» на патент. Патентообладатель может подать в патентное ведомство заявление о том, что он обязуется предоста-

«Понятие “публичной лицензии” в нашем законодательстве есть.»

вить право использовать свой патент любому желающему. Взамен он получает бонус в виде двукратного

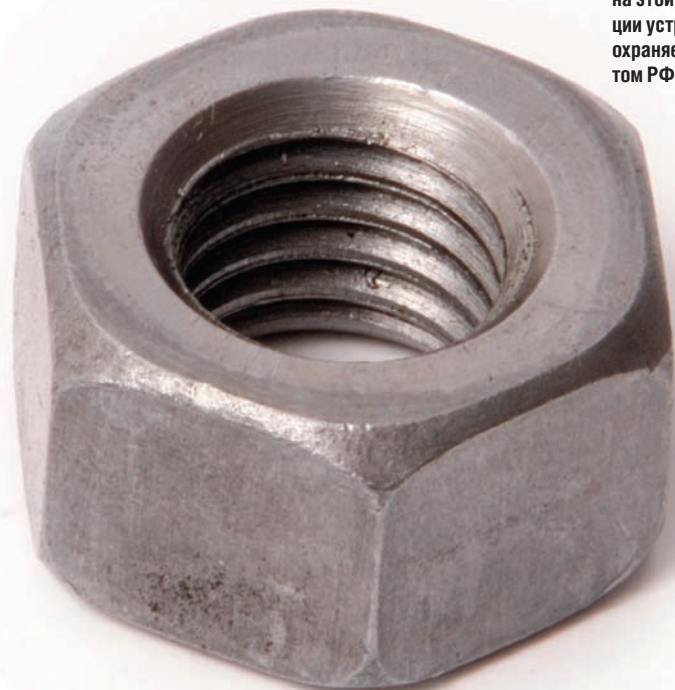
уменьшения пошлины за поддержание патента. Условия такой лицензии публикуются, и патентообладатель обязан заключить на этих условиях договор, если желающий использовать изобретение найдется.

Фактически «открытую лицензию» описывает и одиннадцатый, «патентный» раздел третьей версии GNU GPL: патентообладатель обязан передать права на использование изобретения всем получателям использующего его кода.

Фактически, один из основных принципов разработки свободного ПО, сложившихся стихийно, повторяет принцип из патентного права. При таких условиях борьба СПО-сообщества с «софтверными патентами» выглядит недостаточно радикальной. Бороться надо не против них, а за то, чтобы программы не охранялись авторским правом. Жаль только, что такая борьба никаких шансов на успех не имеет...

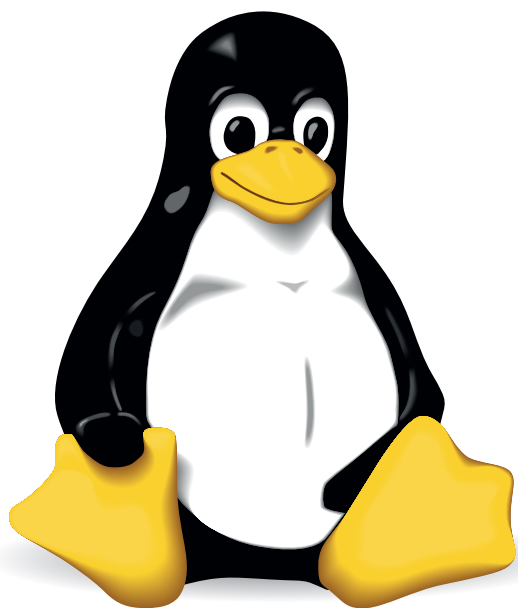
Да, и, наконец, долгожданная разгадка: приведенное описание – цитата из отечественного патента № 2145006. Описана в нем обыкновенная гайка... LXF

› Изображенное на этой иллюстрации устройство охраняется патентом РФ № 2145006.



Ссылки

- 1 Страница «дела Билски» на сайте сайта FSF «End Software Patents»: [http://en.swpat.org/wiki/Bilski_v._Kappos_\(2010,_USA\)](http://en.swpat.org/wiki/Bilski_v._Kappos_(2010,_USA))
- 2 Страница с документами по арбитражному делу «Суперфона»: <http://kad.arbitr.ru/?id=2C24F67A-8BBE-4746-B18A-A98A979D0050>
- 3 Patent-troll sues entire security industry over behavior-based tech // <http://www.thetechherald.com/article.php/200902/2748/Patent-troll-sues-entire-security-industry-over-behavior-based-tech>



Сам себе зеркало

Виталий Сороко расскажет, почему вам может захотеться создать локальный Debian-репозиторий, и покажет, как это сделать.



Наш эксперт

Виталий Сороко

Любит все новое и менять мир к лучшему. Свободное ПО, а также степень магистра технических наук, этому только способствуют.

Репозиторий — это, как известно, вещь, присутствующая у большинства дистрибутивов Linux и представляющая собой коллекцию программ, заботливо собранных разработчиками в виде готовых для установки пакетов. Однако в некоторых случаях имеет смысл создать локальный репозиторий. Несмотря на технический прогресс, который привел к появлению высокоскоростного Интернета, скачивание пакетов из центрального репозитория или его зеркал не всегда является наилучшим решением. Если у вас всего один компьютер, то создание локального репозитория будет иметь следующие преимущества:

- » Отпадает необходимость заново скачивать нужные для работы пакеты из Интернета после переустановки системы.
- » Уменьшается зависимость от провайдера — вам будут не страшны «профилактические работы».
- » Обеспечивается экономия интернет-трафика и пропускной способности канала, что приводит к увеличению скорости скачивания, а значит, и самого главного — вашего личного времени.

Техника и финансы

В принципе, при создании локального репозитория можно обойтись и одним компьютером, используемым в качестве рабочей станции, просто доустановив недостающие компоненты. Преимущества этого подхода очевидны: вам не нужно тратить на покупку дополнительного оборудования (сервера и сетевых компонентов), внедрение и обслуживание, а также на дополнительный

расход электроэнергии. Но недостатки у него также присутствуют — существенно снижается производительность «сервера»; и если при использовании репозитория только для собственных нужд это не слишком заметно, то в локальной сети в моменты пиковой нагрузки вы не сможете нормально работать за таким компьютером. Да и отказоустойчивость подобной системы довольно низкая, поскольку, сделав что-нибудь не так, вы рискуете вместе с системой потерять и репозиторий, что сводит на нет добрую половину преимуществ от его создания (см. выше). Поэтому мы пойдем по другому пути: выделим для хранения репозитория отдельный компьютер.

Чтобы создать локальный репозиторий у себя дома или в офисе, совершенно не обязательно покупать дорогостоящий сервер корпоративного уровня с несколькими процессорами и жесткими дисками, объединенными в RAID-массив. Будет вполне достаточно компьютера класса Pentium, с O3Y 128 МБ и сетевой картой на 100 Мбит. Однако необходимо иметь в виду, что на такой компьютер придется установить жесткий диск большого объема, так как одно только зеркало репозитория Ubuntu 9.04 (Jaunty) со всеми пакетами займёт около 60 ГБ, а вместе с Lucid Lynx может занять около 130–140 ГБ. Не все старые материнские платы могут корректно работать со внутренними жесткими дисками такого объема. Неплохим выходом в этом случае будет покупка совместимого с операционной системой контроллера USB 2.0 или FireWire и внешнего жесткого диска под этот порт. А если у вас уже есть внутренний жесткий диск нужного объема, можно просто докупить для него внешний бокс.



► Рис. 1. Домашний сервер с локальным репозиторием внутри.

Кроме того, использование внешнего жесткого диска дает ряд дополнительных преимуществ: при необходимости можно будет закачивать пакеты для зеркала с другого компьютера; упрощается создание резервных копий и замена сервера или жесткого диска в случае поломки. Лично я использую в качестве сервера компьютер со следующей конфигурацией (рис. 1): процессор Celeron-700, ОЗУ 192 МБ, гигабитная сетевая карта, попате-контроллер USB на чипе VIA VT6212L, встроенная видеокарта на 8 МБ, внутренний жесткий диск на 2,1 Гб, внешний LaCie на 500 Гб (основной) и Samsung STORY STATION 1 ТБ (резервный). Такой сервер вместе с внешним жестким диском и монитором обошелся мне примерно в \$100.

Однако этот путь стоит повторять только при недостатке денежных средств, так как сейчас в продаже появились неттопы на базе Intel Atom. Компьютер, использующий двухъядерный чип Atom 330 или D510, почти бесшумен и, при том, что быстрее в несколько раз, потребляет примерно на 50 % меньше электроэнергии. В общем, неттопы, на мой взгляд, для наших целей подходят как нельзя лучше – особенно если учесть, что цена таких устройств обычно менее \$300. Еще один возможный вариант – «компьютеры-вилки», вроде SheevaPlug/GuruPlug (LXF126–127): с доставкой в Россию они обойдутся примерно в \$150–\$200, жесткий диск придется докупить дополнительно.

Ну, а если вы собираетесь использовать репозиторий для большого количества компьютеров (свыше 100), имеет смысл задуматься о покупке начального однопроцессорного сервера на базе Intel Xeon, который обычно стоит \$800–1200, или, в крайнем случае, обычного компьютера с двухъядерным или четырехъядерным процессором. В последнем случае важно правильно выбрать корпус – лучше всего отдать предпочтение моделям, у которых внутри имеется большое количество отсеков 3,5 дюйма и блок питания мощностью не менее 450 Вт, так как это поможет избежать проблем в дальнейшем, а также позволит с легкостью модернизировать этот компьютер на случай, если вы соберетесь использовать его для других задач, вроде хранения файлов (в роли файл-сервера). Такие модели корпусов выпускают многие производители, например, Modcom (рис. 2).

Программная начинка

В первую очередь вам, конечно, понадобится операционная система. Я использую для этих целей Ubuntu Server 9.04. Подошел бы и любой другой дистрибутив на базе Debian (раз уж мы созда-

ем репозиторий для этого семейства); для Fedora/Red Hat рекомендуем CentOS; для SUSE – openSUSE, и так далее. Понимается, в случае систем, отличных от Debian, инструменты также потребуются другие; но это тема для отдельной статьи. Далее, в зависимости от целей создания репозитория, вам понадобятся утилиты *apt-move* (предназначена для создания частичного репозитория из уже скачанных пакетов) или *apt-mirror* (для создания полного зеркала репозитория). В последнем случае можно также воспользоваться утилитой *debmirror*, которая в целом по функциям аналогична *apt-mirror*, но в отличие от последней не требует редактирования конфигурационного файла – все необходимые параметры задаются прямо из командной строки. В общем, как и всегда, мир свободного ПО полон альтернатив, так что выбирайте то, что вам удобнее.

Первым из инструментов мы рассмотрим утилиту *apt-move*. Она предназначена для извлечения из кэша уже установленных пакетов и последующего создания репозитория на их основе. Это полезно в том случае, если вы хотите сохранить уже скачанные и установленные пакеты перед переустановкой системы. Устанавливается *apt-move* с помощью команды `sudo apt-get install apt-move`. Для настройки *apt-move* необходимо отредактировать конфигурационный файл, который находится по адресу `/etc/apt-`

`move.conf`. Для этого можно воспользоваться консольным текстовым редактором *Nano*, введя команду `sudo nano /etc/apt-move.conf`. После установки этот файл уже содержит параметры, поэтому от вас, ве-

роятно, потребуется лишь задать нужный путь хранения репозитория (переменная `LOCALDIR`) и изменить другие настройки по умолчанию. Для большей наглядности приведем листинг файла `apt-move.conf`:

```
APTSITES="/all/"
LOCALDIR=/media/LaCie/ubuntu
DIST=stable
PKGTYPE=both
FILECACHE=/var/cache/apt/archives
LISTSTATE=/var/lib/apt/lists
DELETE=no
MAXDELETE=20
COPYONLY=yes
PKGCOMP=gzip
CONTENTS=yes
GPGKEY=
```

Как видите, параметров не так уж и много, поэтому вкратце рассмотрим большую часть из них.

»



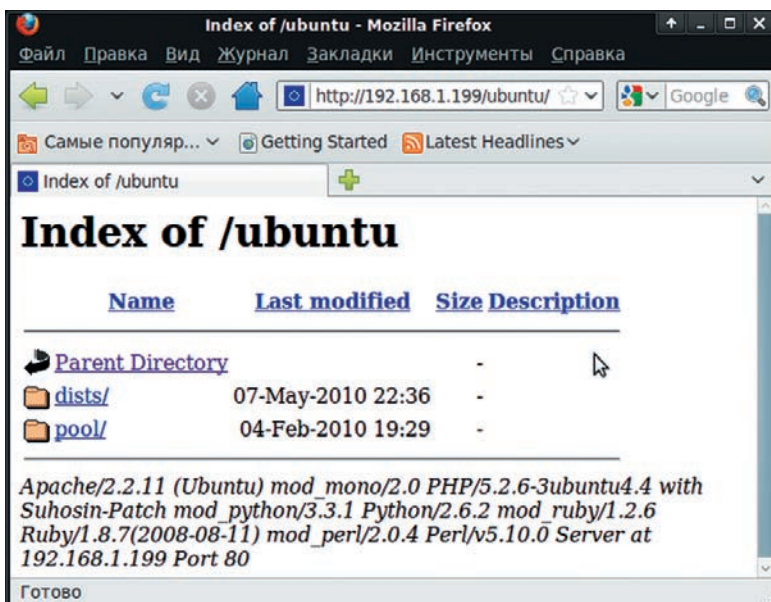
► Рис. 2. В корпус Modcom Platinum можно установить более 9 жестких дисков.



Рис. 3. Хранить репозиторий для дома или малого офиса на таком оборудовании будет явным перебором.

- » **APTSITES** Отбирает сайты из `/etc/apt/sources.list`, пакеты с которых должны попасть в локальный репозиторий (по умолчанию – все сайты).
- » **PKGTYP** Тип пакетов (binary или source; по умолчанию переносятся оба типа).
- » **FILECACHE** Путь к кэшу `apt`.
- » **LISTCACHE** Путь к списку и описанию пакетов, находящихся в кэше `apt`.
- » **DELETE** Указывает на необходимость очистки репозитория от ставших ненужными пакетов.
- » **COPYONLY** Определяет, будет ли производиться очистка локального кэша после переноса пакетов в локальный репозиторий.

Рис. 3. Просмотр локального репозитория в окне web-браузера Firefox (Xubuntu).



» **CONTENTS** Указывает на необходимость создания архивов со списком содержимого подразделов локального репозитория.

Подробнее об остальных параметрах вы можете узнать, прочитав страницу руководства по данной утилите, набрав в консоли команду `man apt-move`.

Зеркалирование

В отличие от вышеописанного способа, создание локального «зеркала» на базе центрального репозитория позволит вам удовлетворить потребности неограниченного числа клиентских компьютеров, что особенно полезно в корпоративной среде, а также если у вас большая семья и у каждого есть компьютер с установленным Ubuntu или Debian Linux. В общем случае, создание локального репозитория на базе центрального осуществляется по следующему алгоритму:

- 1 Выбор утилиты для зеркалирования (`apt-mirror` или `debmirror`).
- 2 Выбор сервера, содержащего базовый репозиторий.
- 3 Выбор архитектуры (i386, amd64 или обе).
- 4 Выбор нужных для зеркалирования разделов репозитория (`main`, `multiverse`, `universe` и т. д.).
- 5 Выбор дополнительных параметров.
- 6 Выбор пути для хранения репозитория.

Чтобы вы не запутались, я сразу оговорю основные параметры создаваемого далее локального репозитория (на примере утилит `apt-mirror` и `debmirror`). Итак, мы будем создавать локальный репозиторий для Ubuntu 10.04 с архитектурой i386, содержащий подразделы `main`, `multiverse`, `universe` и `restricted`, а также архивы со списком содержимого этих подразделов. В качестве базового репозитория у меня будет выступать репозиторий, находящийся в интернете по адресу <http://ftp.byfly.by> (этот неофициальный репозиторий находится на сервере моего интернет-провайдера – вы, конечно, возьмете что-то свое). Создаваемый локальный репозиторий будет храниться на внешнем жестком диске LaCie, и путь для хранения будет `/media/LaCie/ubuntu/`. С целью экономии дискового пространства пакеты, содержащие исходный код, не загружаются, а в конце каждого обновления соответствующие утилиты проведут очистку от ненужного (устаревшего) содержимого.

Для установки `debmirror` наберите `sudo apt-get install debmirror`, а запуск с основными параметрами имеет следующий вид:

```
sudo debmirror --nosource --passive --host=ftp.byfly.by
--root=/ubuntu/ --method=http --progress --dist=ucid --section
=main,restricted,multiverse,universe --arch=i386 /media/LaCie/
ubuntu/ --getcontents --ignore-release-gpg
```

Поясним назначение основных опций (параметров) `debmirror`.

- » **--nosource** Не закачивать пакеты с исходным кодом (экономит место на диске).
- » **--passive** Пассивный режим скачивания.
- » **--host** URL репозитория.
- » **--root** Путь к репозиторию на сайте (подкаталог, содержащий репозиторий).
- » **--method** Метод скачивания (http, ftp, hftp, rsync).
- » **--progress** Отображение состояния скачивания.
- » **--dist** Версия дистрибутива ОС (jaunty, karmic, lucid, ...).
- » **--section** Подраздел репозитория с пакетами (main, multiverse, universe, ...).
- » **--arch** Архитектура вашего компьютера (i386, amd64).
- » **--getcontents** Сохранять архивы с содержимым (в некоторых случаях необходимо для совместимости).
- » **--ignore-release-gpg** Игнорировать отсутствие цифровой подписи (полезно при скачивании с некоторых интернет-зеркал официального репозитория, хотя и менее безопасно).
- » **--cleanup** Принудительная очистка каталогов репозитория от неизвестного содержимого (неизвестных пакетов) – в отличие от очистки по умолчанию, производится в любом случае.

О других параметрах вы можете узнать из справки, введя команду **man debmirror**; ну, а мы перейдем к рассмотрению аналога *debmirror* – утилите *apt-mirror*. В целом, использование данной утилиты аналогично рассмотренной выше *apt-move*; разница только в местонахождении (*/etc/apt/mirror.list*) и содержимом конфигурационного файла. Устанавливается *apt-mirror* стандартной командой **sudo apt-get install apt-mirror**. Для редактирования конфигурационного файла, как и в предыдущем случае, воспользуемся текстовым редактором *Nano* (**sudo nano /etc/apt/mirror.list**). Вам потребуется задать нужный путь хранения репозитория (переменная *set base_path*) и web-адрес зеркала, а также дополнительные параметры. Для достижения результата, аналогичного полученному выше с помощью утилиты *debmirror*, конфигурационный *apt-mirror* файл должен выглядеть следующим образом:

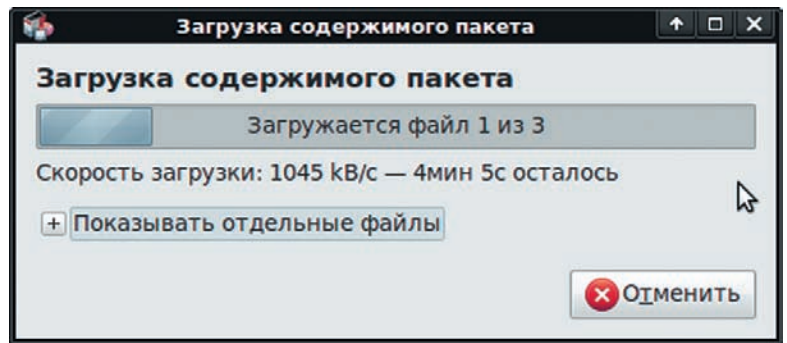
```
set base_path /media/LaCie
set mirror_path $base_path/ubuntu
set skel_path $base_path/skel
set var_path $base_path/var
set cleanscript $var_path/clean.sh
set defaultarch i386
set nthreads 1
set _tilde 0
deb http://ftp.byfly.by/ubuntu lucid main restricted universe
multiverse
```

Обратите внимание на то, что *apt-mirror*, в отличие от утилиты *debmirror*, не осуществляет автоматическую очистку репозитория после окончания процесса скачивания. Вместо этого вам придется вручную запустить скрипт **clean.sh**. О его местонахождении и синтаксисе команды для запуска *apt-mirror* непременно сообщит перед окончанием своей работы. Кроме того, переменная *nthreads* позволяет задать количество потоков для скачивания, что в некоторых случаях очень полезно и позволяет избежать проблем, а также оптимизировать скорость загрузки и нагрузку на интернет-соединение. И еще одно важное отличие пакета *debmirror* от *apt-mirror* – первый обычно сохраняет репозиторий в строго определенном месте, ничего не добавляя при этом к указанному пути; *apt-mirror* же скорее всего добавит имя сайта, с которого был скачан репозиторий. Я неспроста сейчас отметил эту деталь. Она будет очень важна на нашем следующем этапе – предоставлении доступа к созданному вами локальному репозиторию по сети.

Доступ по сети

Через некоторое время работы, процесс скачивания пакетов и формирования структуры локального репозитория будет завершен, и утилита *debmirror* или *apt-mirror*, как я надеюсь, сообщит о его успешном создании. После того как локальный репозиторий будет создан, вам необходимо будет предоставить к нему доступ – как системе с вашего компьютера, так и всем остальным. Наилучшим способом достичь этого, на мой взгляд, является установка и использование web-сервера *Apache*, так как с его помощью проще всего организовать доступ к репозиторию и распределить нагрузку на сервер и сетевой канал. Естественно, первым делом его нужно установить, командой **sudo apt-get install apache2**. Затем необходимо создать Alias для каталога, в котором содержится локальный репозиторий, и разрешить его просмотр. Для этого необходимо открыть (можно опять с помощью текстового редактора *Nano*) конфигурационный файл, находящийся в **/etc/apache2/sites-enabled/@000-default**, и добавить в него следующие строки:

```
Alias /ubuntu/ /media/LaCie/ubuntu/ftp.byfly.by/ubuntu/
<Directory "/media/LaCie/ubuntu/ftp.byfly.by/ubuntu/">
```



► Рис. 4. 1045 Кб/с – и никакой нагрузки на интернет-канал.

```
AllowOverride None
Options +MultiViews +Indexes
Order allow, deny
Allow from all
</Directory>
```

Здесь **/media/LaCie/ubuntu/ftp.byfly.by/ubuntu/** – путь к содержимому локального репозитория, который в вашем случае, скорее всего, будет называться иначе. Затем необходимо перезагрузить web-сервер командой

```
sudo /etc/init.d/apache2 restart
```

На всякий случай напомним, что для разрешения возможности просмотра содержимого в web-сервере Apache, целевой каталог должен иметь соответствующие права (как минимум, на просмотр и поиск). После этого можно проверить результат, введя в web-браузере (рис. 3) адрес репозитория, который состоит из IP или символического адреса сервера и пути **/ubuntu/**. В моем случае это **http://192.168.1.199/ubuntu/**.

После этого, в случае успеха, можно смело настраивать менеджер пакетов на созданный локальный репозиторий. Для этого необходимо добавить его в список, находящийся в файле **/etc/apt/sources.list**. В принципе, это можно сделать с помощью графических инструментов, но наиболее простым и быстрым способом, на мой взгляд, все же является редактирование файла **sources.list** с помощью текстового редактора. В зависимости от используемой системы это может быть *Gedit* (Ubuntu) – **sudo gedit /etc/apt/sources.list**; *Kate* (Kubuntu) – **sudo kate /etc/apt/sources.list**; *Mousepad* (Xubuntu) – **sudo mousepad /etc/apt/sources.list**; *Nano* (Ubuntu Server и др.) – **sudo nano /etc/apt/sources.list**.

Поскольку весь смысл создания локального репозитория сводится к тому, чтобы клиентские машины не скачивали пакеты из центрального репозитория в Интернете, то от вас потребуется заменить URL нужного репозитория на адрес вашего сервера. Так, в нашем случае после модификации соответствующие строки в файле **/etc/apt/sources.list** примут следующий вид:

```
deb http://192.168.1.199/ubuntu/ lucid main restricted
deb http://192.168.1.199/ubuntu/ lucid universe
deb http://192.168.1.199/ubuntu/ lucid multiverse
```

После этого необходимо обновить кэш *apt* с помощью команды **sudo apt-get update** либо используя менеджер пакетов. Ну и, на всякий случай, для надежности, можно проверить работу локального репозитория, установив оттуда что-нибудь через стандартный менеджер пакетов; например, в Ubuntu и Xubuntu таковым является *Synaptic* (рис. 4).

Пользуясь случаем, хочу напомнить, что будет совсем не лишним почаще обновлять локальный репозиторий и делать его резервные копии. На этом, пожалуй, и закончим – и удачи вам в работе со свободным программным обеспечением! **LXF**

ДОК: Новый рабочий стол

Она стала занимать слишком много места и удручать армейским единообразием? Замените панель рабочего стола на док, советует **Семен Есилевский**.



Наш эксперт

Семен Есилевский
 Научный сотрудник, вычислительный программист и энтузиаст Linux и Open Source.

➤ Рис. 1. Диалог конфигурации Cairo-Dock. Каждый пункт имеет десятки опций тонкой настройки — есть где потереяться.

Интерфейс практически всех популярных рабочих окружений, таких как Gnome, KDE, XFCE и LXDE, основан на концепции панели — горизонтальной полосы вверху или внизу экрана, на которой располагается системное меню, панель быстрого запуска приложений, панель задач, область уведомлений, а также часы, календари, индикаторы и прочее. Панели привычны и понятны всем; однако в последнее время они все больше впадают в немилость как у пользователей, так и у разработчиков. Одна из причин — поиск новых, более современных подходов к организации интерфейса, а другая — изменение геометрии экранов. Все современные ноутбуки и многие настольные мониторы являются «широкими», то есть имеют избыток горизонтального и дефицит вертикального пространства. В нетбуках с их крохотными экранами дефицит места по вертикали становится просто катастрофическим. Панели же горизонтальны и еще больше сужают вертикальное пространство, а «поля» по бокам экрана не используются вовсе.

«Панели не рассчитаны на вертикальную ориентацию.»

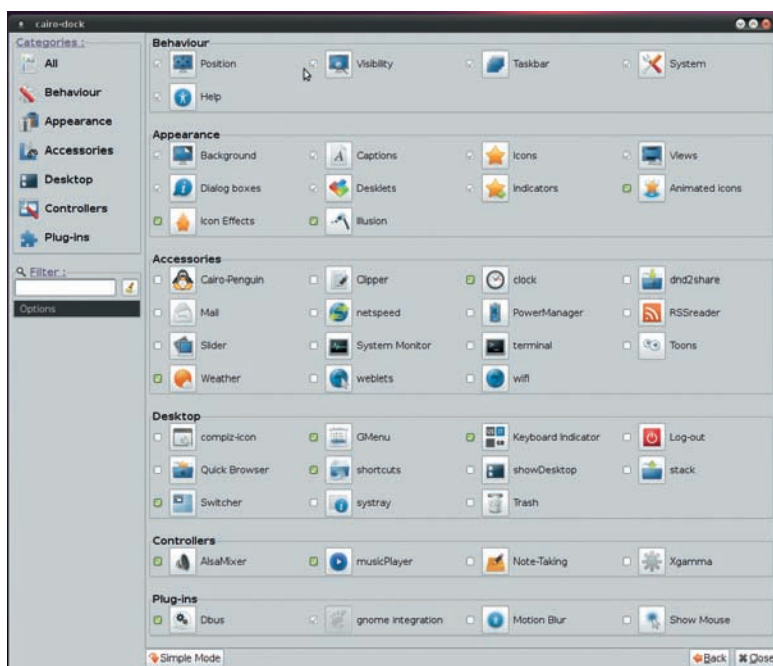
Естественно, многие пользователи приходят к решению разместить панель вертикально вдоль левого или правого края экрана. Вот тут их и ждет сюрприз — ни в одной из рабочих сред панели изначально не рассчитаны на вертикальную ориентацию. В лучшем случае они выглядят не так красиво, а в худ-

шем — становятся нефункциональными. По моему опыту, лучше всего дела с вертикальными панелями обстоят в KDE, а сквернее всего — в Gnome. Не буду описывать многочисленные вопиющие несообраз-

ности, возникающие в Gnome при попытке разместить панель вертикально (достаточно просто попробовать). Для пользователей Gnome существует дополнительная причина искать альтернативу панелям как таковым. В грядущем Gnome 3 весь интерфейс будет кардинально переделан, и появится Gnome Shell (<http://live.gnome.org/GnomeShell>) — революционная, но довольно сомнительная в плане удобства и функциональности оболочка (в которой опять, кстати, проигнорирована проблема экономии вертикального пространства экрана). Кроме чисто функциональных причин, многие пользователи ищут замену панелям с эстетической точки зрения либо стараясь сделать рабочий стол нестандартным и менее «скучным».

Одно из решений этих проблем — использование доков. Идея дока как ключевого элемента рабочего стола появилась еще в ранних графических средах для Unix, таких как CDE, однако всеобщую известность они получили благодаря системе Mac OS X. Док предоставляет ту же функциональность, что и панель, но выглядит и ведет себя иначе. В доке все элементы отображаются иконками одинакового размера со всплывающими подписями и различными индикаторами, показывающими тип и состояние объекта. Док по определению компактнее аналогичной по функциональности панели. Если разместить его вертикально, то он с пользой заполняет «мертвую» боковую зону справа или слева и оставляет свободной всю высоту экрана. И вертикальный, и горизонтальный доки можно использовать как заменитель стандартной панели или как дополнение к ней.

Существует около десятка доков для разных графических сред, однако достаточно зрелыми и активными являются только два проекта — Avant Window Navigator (AWN, <http://awn-project.org>) и Cairo-Dock (также известный как Glx-Dock, www.glx-dock.org). Оба они позиционируются как полноценные заменители панели в среде Gnome (а Cairo-Dock — также в KDE 4 и Xfce), но так ли это на самом деле? Попробуем выяснить.





► Рис. 2. Тема «Rings» с экстравагантными визуальными эффектами («дождь» и вращение иконок при наведении указателя мыши).

Cairo-Dock

Cairo-Dock – активно развивающийся проект, призванный полностью заменить панель в средах Gnome, KDE 4 и Xfce. Кроме основной программы, в него входят многочисленные модули, апплеты и темы оформления. Версии, входящие в репозитории основных дистрибутивов, обычно успевают устареть, поэтому лучше использовать пакеты или репозитории с официального сайта <http://www.glx-dock.org/>. Имеются пакеты для Debian, Ubuntu, Fedora, Mandriva, OpenSuse и других дистрибутивов.

После установки в меню появляются два варианта запуска дока – с использованием OpenGL и без него. Второй вариант стоит использовать только при возникновении проблем с поддержкой OpenGL видеокарты либо при проблемах с композитным режимом в X.Org. По умолчанию док расположен внизу экрана, а настройки призваны скорее продемонстрировать его возможности, чем обеспечить комфортную повседневную работу.

Количество настроек в Cairo-Dock поначалу просто ошеломляет (рис. 1). Здесь даже специально предусмотрен упрощенный вариант диалога конфигурации для новичков, в котором удобно сгруппированы наиболее востребованные опции.

Док может отображаться как на двумерной подложке, так и в 3D на зеркальной плоскости (а-ля MacOS X). Некоторые апплеты открывают дочерние доки, вид которых также настраивается независимо и очень гибко («карусель», параболическая дуга, «веер» и т.д.). В каждом из режимов доступны десятки параметров тонкой настройки. Поддерживается «parabolic zoom» – плавное масштабирование иконок в виде волны, бегущей за курсором мыши (этот красивый эффект является «визитной карточкой» дока в MacOS X). Для иконок независимо настраивается анимация при наведении мыши и щелчке, появлении и исчезновении из дока, получении окном программы фокуса, сигнализации пользователю о каких-то событиях. Среди доступных эффектов – подсветка, пульсация, 3D-вращение, рябь, волны, «подпрыгивание», «натягивание» на вращающийся куб и т.п. «Поверх» анимации можно накладывать эффекты огня, дождя, салюта, снега, мерцающих звезд и т.д. Различные сочетания анимации и дополнительных эффектов открывают широкий простор для фантазии и позволяют сделать ваш док непохожим ни на один другой.

Все иконки дока подразделяются на апплеты, пусковые кнопки [launchers] и программы (значки открытых в данный момент окон – аналог панели задач). Пусковые кнопки могут одновременно служить и значком открытого окна программы (как в Windows 7 или в OS X); при этом отображается индикатор, вид которого также гибко настраивается. Порядок расположения, внешний вид и анимация для иконок каждого типа настраиваются независимо.

С Cairo-Dock поставляются несколько стандартных тем оформления. Еще несколько десятков тем автоматически загружаются и устанавливаются со специального сервера одним щелчком мыши. Многие из имеющихся тем изобилуют «агрессивными» визуальными эффектами (рис. 2) и быстро начинают раздражать.

На мой взгляд, оптимальными являются темы Clear и Humanity-Dock, но это дело личного вкуса. После того как док настроен оптимальным для вас образом, можно сохранить текущее оформление в виде новой темы и даже создать архив для ее распространения (рис. 3).

Все визуальные эффекты и анимация прорисовываются очень качественно и плавно. Изредка встречаются некритичные артефакты (вроде срезанных краев подписей), которые не портят общего впечатления.

Функциональность Cairo-Dock дополняется большим набором апплетов. Все они могут находиться как внутри дока, так и просто на рабочем столе в виде независимого «десклет» [desklet]. Десклеты можно гибко настраивать, меняя их размеры, фон и положение. Кроме этого, любой десклет можно свободно вращать, причем не только в плоскости экрана, но и в трехмерном пространстве! Для всех этих операций поверх десклета отображаются соответствующие значки-захваты. Можно также зафиксировать десклет из контекстного меню, чтобы случайно не сбить его настройки.

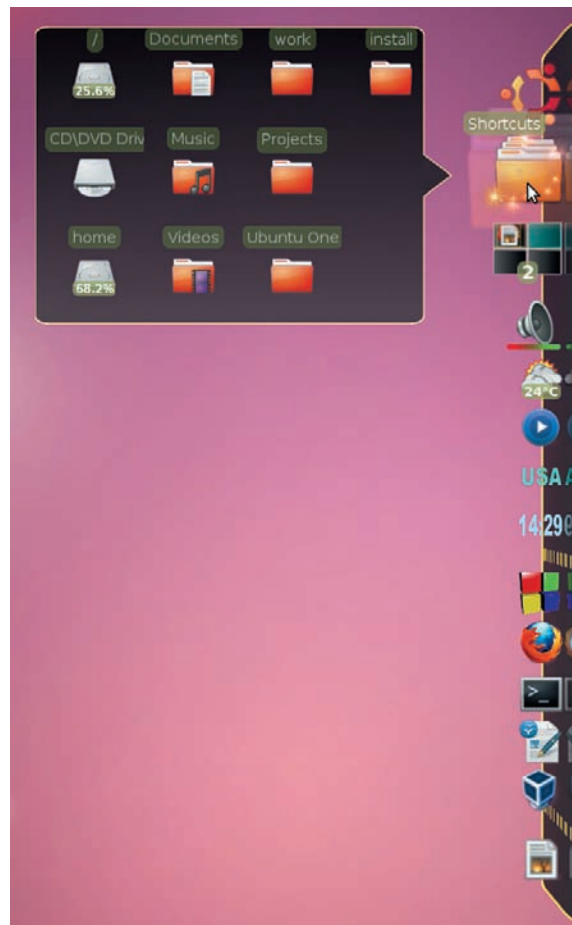
Некоторые из апплетов представляют из себя просто безделушки, но многие действительно незаменимы:

► **AlsaMixer** Управляет громкостью звука. Щелчок средней кнопкой включает/выключает звук, колесо мыши регулирует громкость.

► **Gmenu** Главное меню Gnome, содержащее все пункты, кроме подменю Places [Переходы].

► **Keyboard Indicator** Отображает раскладку клавиатуры и меняет ее по щелчку или повороту колеса мыши. Смена раскладки красиво

»



► Рис. 3. Модифицированная тема «Humanity Dock». Док расположен вертикально. Показаны анимация иконок, эффект звезд и дочерний док апплета «Shortcuts».



➤ Рис. 4. AWN и диалог для его настройки.

анимирована. К сожалению, раскладка отображается исключительно текстом, режима отображения флажка страны нет.

➤ **Shortcuts** Заменяет подменю Places, а также позволяет монтировать и отключать диски и флэшки. Под иконками дисков и сменных носителей отображается свободное пространство на них.

➤ **Switcher** Переключатель рабочих столов. Может работать непосредственно в основном доке или открывать дочерний док со списком всех рабочих столов.

➤ **Weather** Отображает текущую погоду и прогноз на несколько дней.

➤ **Clock** Отображает аналоговые или числовые часы. В аналоговом режиме рассмотреть в доке стрелки довольно сложно, поэтому лучше использовать его как десклет либо установить числовой режим и подобрать подходящий шрифт.

➤ **System Monitor** – системный монитор, который может отображать загрузку процессора, использование памяти и температуру видеокарты. Желательно использовать как десклет, т.к. разглядеть что-либо на индикаторах в доке затруднительно.

Для трех последних апплетов (Weather, Clock и System Monitor) имеются свои собственные темы оформления. Так, внешний вид системного монитора может варьироваться от автомобильного спидометра в стиле «ретро» до футуристического цветowego индикатора.

Имеются также апплеты для индикации заряда батареи ноутбука, силы сигнала Wi-Fi, наличия новых почтовых сообщений, контроля музыкального плеера (поддерживаются все популярные программы). Имеются также апплеты сторонних разработчиков, не входящие в официальный пакет Cairo-Dock. Они доступны на странице http://www.glx-dock.org/mc_album.php?a=4 и добавляются в док перетаскиванием ссылки прямо из браузера, что выглядит очень эффектно.

Все эти апплеты работают практически безупречно, однако для полной замены панели Gnome не хватает области уведомлений (трея). К сожалению, с этим в Cairo-Dock большие проблемы. Апплет области уведомлений реализован как нестабильная и неэлегантная «обертка» для кода из Gnome panel. Его также нужно помещать отдельно от дока где-нибудь в углу экрана.

➤ Рис. 5. Апплет «Preferences» в действии. Док можно перетаскивать и менять его размер.



В текущей версии Cairo-Dock (2.1.3-6) он вообще отказался работать как следует. По информации на форуме, разработчики не намерены совершенствовать этот апплет до появления нового универсального стандарта области уведомлений для основных рабочих сред, который все равно заставит переписывать апплет с нуля. Таким образом, обойтись исключительно одним Cairo-Dock довольно затруднительно – приходится держать где-то в углу маленькую панель Gnome с областью уведомлений либо использовать отдельные программы для отображения уведомлений, например stalonetray (<http://stalonetray.sourceforge.net/>). Не считая проблем с этим апплетом, Cairo-Dock является полноценной и очень красивой заменой панели Gnome.

Нужно сказать что Cairo-Dock достаточно хорошо интегрирован со средой Xfce. Однако некоторые апплеты, разработанные исключительно для Gnome (например, индикатор раскладок), в Xfce работать не будут. Об интеграции Cairo-Dock со средой KDE 4 будет отдельно сказано ниже.

Avant Window Navigator

Avant Window Navigator (AWN) – док, разработанный исключительно для среды Gnome и призванный полностью заменить стандартную панель. В Ubuntu AWN устанавливается из PPA-репозитория (<http://wiki.awn-project.org/DistributionGuides>). Имеются также пакеты для Debian, OpenSUSE, Fedora, Mandriva и Arch Linux.

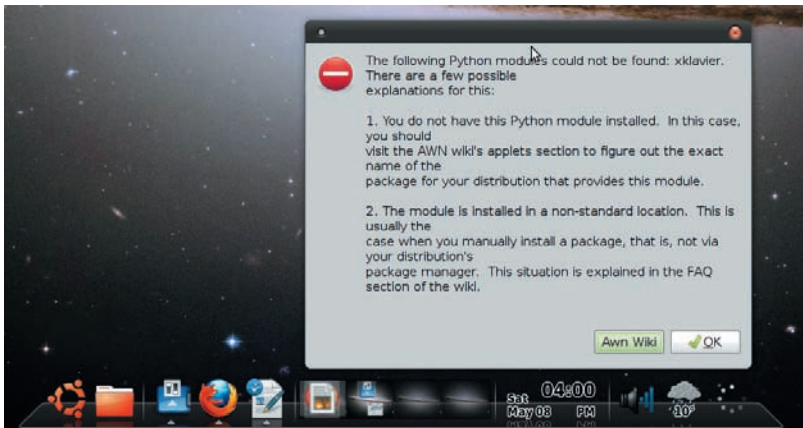
Идеологически AWN очень похож на Cairo-Dock – он также состоит из основной программы, апплетов и тем оформления. В отличие от Cairo-Dock, AWN практически не нужно настраивать – конфигурация по умолчанию достаточно хороша для повседневной работы, а диалог настройки очень прост, интуитивно понятен (рис. 4) и вызывается из контекстного меню дока. Имеется также апплет Preferences, который включен по умолчанию и позволяет менять ориентацию и размер дока «на лету», что несколько напоминает настройку панелей в KDE 4 (рис. 5) и для Gnome выглядит непривычно. Пожалуй, единственным нетривиальным моментом является то, что порядок апплетов в AWN нельзя поменять обычным перетаскиванием иконок в самом доке – для этой цели в диалоге настройки имеется специальное поле.

Функциональность самого дока и визуальные эффекты в AWN беднее, чем в Cairo-Dock. AWN не поддерживает «parabolic zoom»; разные режимы отображения дока отличаются только способом прорисовки фона; нет дочерних доков. Анимация иконок по сравнению с Cairo-Dock весьма примитивна, а гибкая настройка эффектов для разных типов объектов не предусмотрена вовсе. Такая относительная аскетичность оборачивается более «спокойным» внешним видом, который не раздражает и не отвлекает от работы, но в то же время любителям более ярких эффектов и тонкой настройки развернуться негде.

Темы оформления для AWN не вызывают ничего, кроме разочарования. Они просто меняют фон дока и существуют по сути «для галочки». По-видимому, поддержка тем добавлена «на вырост» и со временем будет совершенствоваться.

Подключаемые модули AWN достаточно разнообразны, но, к сожалению, далеко не идеальны как функционально, так и с точки зрения стабильности (рис. 6).

Имеется несколько вариантов главного меню, из которых, пожалуй, самый удобный – Yet Another Menu Applet, который обеспечивает доступ ко всем разделам меню Gnome. Апплет Digital Clock – это удобные числовые часы с хорошо читаемыми циф-



► Dialect Applet работать не желает... К сожалению, это типичная ситуация для многих апплетов в AWN.

рами и удобный календарь. Имеется и апплет Analog Clock – для любителей аналоговых часов; правда, разглядеть их стрелки на небольшом циферблате не так просто, а режима десктола в AWN нет. Volume Control прекрасно справляется с регулировкой громкости, а Weather Applet показывает текущую погоду и прогноз на несколько дней. Апплет Notification Area полностью заменяет область уведомлений панели Gnome (чего так не хватает в Cairo-Dock). Таким образом, чтобы полностью отказать от панели, нужны переключатель рабочих столов и индикатор раскладок клавиатуры. К сожалению, с ними в AWN есть проблемы. Имеются два апплета для переключения рабочих столов – Shiny Switcher и SlickSwitcher. Первый неправильно отображается при вертикальном положении дока, а второй периодически отказывается работать. Если док расположен горизонтально, то Shiny Switcher прекрасно справляется со своим заданием. Любителям вертикального расположения дока придется полагаться на переключение рабочих столов с помощью клавиатуры или мыши. Апплет Dialect Applet для переключения раскладок попросту не работает, жалуясь на отсутствие нужных ему модулей. Однако, даже если бы он работал, его функциональность крайне ограничена – он даже не поддерживает разные языки для разных запущенных программ. Таким образом, как и в случае с Cairo-Dock, приходится держать в углу экрана маленькую панель Gnome, но на этот раз как минимум с индикатором раскладок.

Нестабильность апплетов в AWN уже стала печальной традицией и на сегодняшний день является главной проблемой этого проекта. В остальном AWN является полноценной заменой стандартной панели Gnome.

KDE без панелей

Исторически сложилось так, что замена панелей на доки в основном интересует пользователей Gnome и в несколько меньшей степени – Xfce. В среде KDE доки не снискали широкой популярности. Стандартная панель в KDE 4 не вызывает особых нареканий, очень хорошо интегрирована в систему и даже приемлемо работает в вертикальной ориентации. Это, однако, не лишает ее присущих всем панелям недостатков и отсутствия оригинальности. К сожалению, зрелых и активно развивающихся доков, специфичных для KDE 4, нет. Существует два довольно «сырых» проекта: Xqde (<http://sourceforge.net/projects/xqde/>), который трудно назвать даже альфа-версией, и несколько более зрелый kiba-dock (<http://www.kiba-dock.org/>). Kiba-dock доступен исключительно в виде исходных кодов, а процесс его сборки,

часть апплетов отображаются и работают без нареканий. Исключение составляет апплет Gmenu, который в KDE функционирует неправильно и вместо системного меню отображает только список последних документов. В некоторых темах досадной проблемой является отсутствие иконок для некоторых программ и апплетов. В KDE их либо нет, либо программа их не находит, и вместо них отображаются вопрошительные знаки. Эта проблема снимается сменой темы либо назначением иконки вручную. Пусковые кнопки в большинстве тем настроены «под Gnome» (например, в качестве терминала программа пытается запустить gnome-terminal, а не konsole, как это нужно в KDE). Эти недоразумения также нужно устранять вручную.

В целом Cairo-Dock в KDE 4 требует несколько большей настройки, чем в Gnome, но работает практически идеально и вполне способен заменить стандартную панель. Функциональность, которой нет в Cairo-Dock, можно дополнить виджетами рабочего стола, которые имеются в изобилии.

Выводы

Идея использования дока как заменителя панели в рабочих средах Gnome, KDE и Xfce, что называется, «витает в воздухе». Док придает рабочему столу нестандартный и гораздо более привлекательный вид, чем привычные и успевшие надоесть панели, обеспечивая ту же функциональность. Кроме того, будучи расположенным вертикально, док экономит драгоценное вертикальное пространство на широких мониторах. На сегодняшний день Cairo-Dock и AWN являются полноценными заменителями стандартной панели в Gnome, за исключением одного-двух апплетов, которые еще «не доведены до ума». Cairo-Dock также прекрасно работает в KDE 4 и Xfce. Выбор за вами! **LXF**

мягко говоря, довольно сложен. Официальный сайт проекта попросту пуст, поэтому инструкции по установке нужно искать на форуме и сторонних источниках. Несмотря на многочисленные попытки, мне так и не удалось собрать и опробовать kiba-dock в Kubuntu 10.04.

Единственным доком, который прекрасно функционирует в KDE 4, является все тот же Cairo-Dock (несмотря на то, что он написан без применения библиотеки Qt, на которой основана среда KDE). При запуске в KDE автоматически активируется модуль «KDE-Integration», который позволяет доку гладко встраиваться в рабочее окружение (рис. 7). Сам док и большая

и вместо них отображаются вопрошительные знаки. Эта проблема снимается сменой темы либо назначением иконки вручную. Пусковые кнопки в большинстве тем настроены «под Gnome» (например, в качестве терминала программа пытается запустить gnome-terminal, а не konsole, как это нужно в KDE). Эти недоразумения также нужно устранять вручную.

► Cairo-Dock в KDE 4, тема «Chrome». После небольшой ручной настройки док выглядит и работает идеально.



Что за штука... WebM?

Боб Мосс объясняет, каким образом новый, свободный от патентных обязательств кодек от Google скоро появится и в вашем браузере.

» А что такое кодек, если точно?

Это программа или физическое устройство для кодирования аудио- или видеоданных. Существует много разных кодеков, в том числе MP3, Ogg Theora и DivX. Они принимают звуковые или визуальные данные и сжимают их в «формат-контейнер». А это стоит делать, поскольку необработанные двоичные данные (для звука это обычно WAV-файлы) занимают слишком много места на диске. При этом данные можно отчасти сжимать, без особой потери качества. Вы наверняка слышали о контейнерных форматах MP3, Ogg и фильмах DivX.

» И WebM — один из таких «кодек»?

Не совсем. Это контейнер для двух кодеков. Один из них — аудиокодек Vorbis: если вы конвертировали MP3 или аудио-CD в формат Ogg, то уже имели с ним дело. Vorbis свободен от патентных обязательств и авторских отчислений и широко используется в среде Linux.

» То есть эта технология была с нами всегда?

В случае с Vorbis это верно. Но интрига истории с WebM не в использовании Vorbis, а в появлении нового, свободного от патентных обязательств кодека VP8, который покрывает «видеочасть» контейнера WebM.

» А я думал, что Ogg Theora обрабатывает видеоконтент!

Да, но по соотношению «сжатие-качество» свободный формат отстает от конкурирующего (и несвободного) формата H.264. Theora далеко не так хорошо поддерживается и не имеет богатых покровителей. Последнее очень важно, ведь если MPEG-LA возбудит дело о патенте против Theora, у последнего нет финансовых ресурсов для защиты. Theora удавалось избежать преследований лишь из-за ограниченности применения. Но у этой «неуловимости» есть оборотная сторона: вряд ли какая-нибудь компания захочет связываться с «юридически опасным» форматом.

» Стоп-стоп, а что это за MPEG-LA и почему он на всех набрасывается?

MPEG-LA — название организации, следящей за соблюдением патентного права по H.264. Этот формат присутствует во всех распространенных браузерах и Adobe Flash, а также широко поддерживается изготовителями оборудования. То есть H.264 обладает преимуществом аппаратного ускорения, значительно повышающим качество воспроизведения.

» Поддержка H.264 в стандарте HTML 5 — это плохо?

Для сообщества свободного ПО — да. Обещанный MPEG-LA льготный период (без патентных исков) однажды завершится, и тогда у держателей патентов будут развязаны руки. Тогда любое свободное ПО с возможностью кодирования или декодирования H.264 окажется вне закона и подвергнется судебному преследованию. Это может привести к сокращению разработки свободного ПО для воспроизведения видео, и свободное ПО не сможет обеспечить легальное воспроизведение высококачественного видео, а то и полностью зачухнет.

» И как же WebM сможет обойти патентные преграды?

Так как после приобретения On2 (фирмы-разработчика кодека) Google сам является держателем патента на VP8, он добавил «патентное обещание» к одобренной OSI свободной лицензии WebM. Обещание гласит, что требовать патентных отчислений за использование контейнера никто не будет. Поэтому любой разработчик может добавлять поддержку формата в свои приложения, не опасаясь вызова в суд.

» А что это за «одобренная OSI лицензия», которую использует WebM?

Это лицензия BSD-типа, одобренная «Инициативой открытого ПО» (OSI) — органом, который следит за тем, чтобы свободные лицензии были подлинно свободными и защищали открытость

исходного кода. Лицензия тоже связана с «патентным обещанием»: если Google подвергнется преследованиям по поводу нарушения патентов, их держатель автоматически лишается прав на патенты WebM. Звучит тревожно, но такое положение можно встретить и в Стандартной общественной лицензии GNU GPL третьей версии, и во второй версии лицензии Apache. Разница в том, что WebM разделяет патентное и авторское право — то есть, даже если Google подвергнется преследованию, WebM останется под защитой свободной лицензии.

» Так почему бы всем сразу не перейти на VP8?

Этот кодек ещё очень «молодой» и не успел обзавестись широкой поддержкой. Но Mozilla и Opera уже пообещали включить поддержку VP8 в свои браузеры, а Adobe добавит её в Flash Player. Есть приятная новость и для тех, кто пользуется *Miro*, *VLC* или *Moovida*: поддержка ожидается совсем скоро.

» Отлично! В чём же основные различия между VP8 и H.264?

Кроме патентного вопроса, разница между двумя кодеками при воспроизведении звука и изображения для конечного пользователя практически неуловима — если фактор аппаратного ускорения не действует. Если действует, то преимущество бесспорно на стороне H.264 — ведь он поддерживается везде и всюду. Как только вопрос поддержки будет снят, разница в патентном обеспечении выйдет на первый план.

» Будет ли WebM представлен не только на настольных ПК?

Да. Планируется партнёрство с изготовителями аппаратного обеспечения, ради единого представления на устройствах любых классов. Теоретически ведь любой прибор — смартфон, планшет, ноутбук, настольный ПК и нетбук — должны воспроизводить видеоматериалы одинаково. Для H.264 это уже обеспечено — значит, чтобы оста-

ваться конкурентоспособным, WebM придётся прийти на эти платформы.

» Скоро ли появится аппаратное ускорение?

Главное преимущество H.264 в том, что он развивается уже давно и поддерживается большинством аппаратных платформ. Но харизматичный Google уже заручился обещаниями поддержки от изготовителей AMD, ARM и Broadcom. Nvidia размышляет о включении поддержки в «систему-на-чипе» Tegra, а Intel обещал подключиться в случае масштабного распространения. Об аппаратной сфере говорить пока рано, но поддержка со стороны ПО ожидается в ближайшие месяцы, если не к концу года.

» А если у меня другая операционная система и я привязан к Internet Explorer или Safari?

Internet Explorer пока не работает с WebM, но если на машине установлены кодеки Vorbis и VP8,

то WebM-контент будет воспроизведён. Чтобы добиться таких же результатов для Safari, пользователям Mac понадобится установить эти кодеки для QuickTime.

» Нужен ли для VP8 специальный кодировщик?

У VP8 нет единого кодировщика или дешифратора. При воспроизведении VP8 использует Ffmpeg, а для пользователей Windows есть фильтры DirectShow. Благодаря открытой природе кодека VP8 и связанного с ним комплекта разработки ПО (SDK) мы скоро увидим множество VP8-плееров, а Firefox, Opera и Chrome будут работать с VP8 без дополнительной настройки.

» Могу я где-нибудь посмотреть контент WebM?

Мне нравится ваш энтузиазм! Если хотите приступить немедленно, можно подключиться к бета-

тестированию HTML 5-версии YouTube (www.youtube.com/html5) и загрузить с той же страницы одну из сборок поддерживаемых браузеров. На время написания статьи VP8-совместимые версии трёх упомянутых браузеров ещё не были опубликованы и существовали в бета-стадии. Но пока журнал дойдёт до вас, ПО должно стать достаточно стабильным для повсеместного использования. VP8-совместимый вариант VLC должен выйти как полноценная версия.

» Приятно слышать! А где можно узнать подробности?

Демонстрационные материалы от Google можно посмотреть по адресу <http://bit.ly/dpK0nS>, а технические данные – на страничке Википедии. За прогрессом можно следить на главном сайте проекта (www.webmproject.org) или On2 (www.on2.com). **LXF**





По рецептам доктора Брауна

Д-р Крис Браун

Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

Добротное администрирование систем из причудливых заворотов кишок серверной.

Проникая вглубь

Какова длина береговой линии Великобритании? По данным картографического управления, она равна 11073 мили. По данным сайта береговой охраны – 7760. Конечно, единственно верного ответа в природе не существует – он зависит от масштаба карты и от изгибчивости вашей мерной ленты.

Я упомянул это для сравнения со столь же нелепым вопросом, нередко задаваемым преподавателям курсов по Linux: «Сколько дней требуется на этот курс?» – что, по сути, значит «Сколько нужно знать о Linux?» Для большинства из нас реальность такова, что мы изучаем его ровно настолько, чтобы делать свою работу, и не более.

Уровни знания

Сколько ни изучай Linux, почти всегда найдется уровень детализации еще глубже. Фракталы вы видели? Вот нечто вроде этого. Например, есть хорошо известная утилита исследования сети *Nmap*. В контексте обучения можно описать ее парой фраз, предъявить минутное демо и двигаться дальше. Или можно показать примеры двух дюжин технологий *Nmap* для обнаружения хостов и сканирования, что отнимет час или два. А можно раздать экземпляры книги Гордона Лайона [Gordon Lyon] по данному предмету (он же и автор сетевого сканера *Nmap*). 468-страничного талмуда хватит, наверное, на добрую неделю летних дождливых деньков.

В своем комментарии к 6-й редакции Unix, опубликованной еще в 1976 году, Джон Лайонс [John Lions] заметил: «Полная документация в принципе помещается в ученическую сумку». (Он говорил о документах, напечатанных на бумаге, а не о компакт-диске или USB-брелке.) Керниган и Пайк [B.W. Kernigan, R. Pike] в своей книге «Программирование в среде Unix» уместно ему возразили: «В последних версиях это было исправлено».



Настройка Ubuntu



Ubuntu Tweak Исправьте те настройки, которые вы всегда хотели изменить, да не знали как.

Возможности настройки рабочего стола Gnome безграничны. Я, в общем, не из тех, кто носится с фоновыми картинками или бьется за точный оттенок розового в контуре окна (хотя, признаюсь, цветовую схему в своем Ubuntu 10.04 я-таки изменил), но вот что я не люблю в настройках по умолчанию – это когда в окне входа в систему отображается список учетных записей пользователей.

Хотите – считайте меня параноиком, и все равно не нравится мне идея демонстрации имен пользователей в окне входа в систему! Поэтому я ужаснулся, узнав, что (как я понял) Canonical, стремясь к упрощению, не позволяет менять это в настройках. Мне известно, что любую настройку Gnome можно задать в базе данных *GConf* с помощью *GConfTool*, имея ее точное название.

Все это навело меня на *Ubuntu Tweak*, графическую программу, в которой легко настроить все, что не слишком доступно в обычном

Будьте бдительны!

Экран Source Center в *Ubuntu Tweak* предоставляет быстрый и простой способ добавления PPA (Personal Package Archive, личного архива пакетов) к вашему списку репозитория. Но знайте: пользуясь этим способом, на время установки пакета вы выдаете автору архива право запуска на вашем компьютере его собственных скриптов от имени root.

графическом режиме дистрибутива. Надо сказать, что опции этой программы довольно экзотичны и охватывают множество областей, а не только Gnome. Например, можно сделать такое:

- » Задать до 12 клавиатурных комбинаций (горячих клавиш для запуска программ).
- » Отключить показ списка пользователей в окне входа в систему (вот это мне было особенно интересно).
- » Отключить появление диалога подтверждения при выходе из системы и выключении компьютера.
- » Изменить способ показа и установки прав доступа к файлам в окне свойств файла *Nautilus*.
- » Установить степень снижения яркости монитора при работе от батареи.
- » Изменить имя хоста компьютера.

Список далеко не исчерпывающий и не систематичный – просто ряд примеров, дающих почувствовать потенциал программы. Более полный перечень возможностей *Ubuntu Tweak* см. на сайте <http://ubuntu-tweak.com> или в LXF132.



» Меню *Ubuntu Tweak* дают понятие о диапазоне варьируемых настроек.

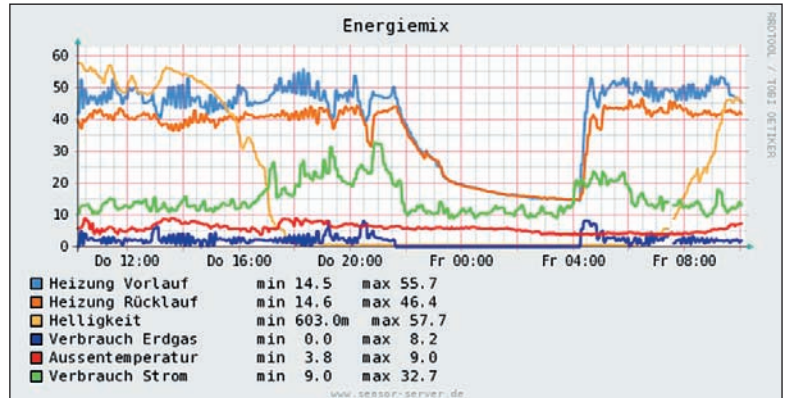
RRDTool

Утилита работы с циклической базой данных Полноценное решение для хранения, обработки и отображения данных временных рядов.

Если вы когда-либо пользовались программой построения графика изменения некой величины во времени, возможно, это была *RRDTool*. Она содержит два компонента: специализированную базу данных для хранения временных рядов и механизм генерации симпатичных масштабируемых графиков по этим данным. *RRDTool* – разработка Тоби Этикера [Tobi Oetiker] (<http://oss.oetiker.ch/rrdtool/>); это база данных, поддерживаемая огромным количеством программ, которые строят графики производительности, загрузки системы, потребления ресурсов, сетевого трафика, производительности сети, температуры, скорости ветра... список можно продолжить.

RRDTool – не база данных в обычном смысле слова, с таблицами и связями между ними. Она приспособлена для хранения и обработки измерений, генерируемых через равные интервалы времени. Например, можно создать базу данных, которая раз в минуту считывает показание температуры в моем парнике. Для хранения данных определим несколько циклических архивов [Round-Robin Archive, RRA]. Каждое значение, поступающее в архив, рассчитывается по заданному числу измерений путем применения к ним какой-либо функции уплотнения – например, среднего, минимума или максимума. Такие значения называются консолидированными точками (КТ). Количество КТ, хранимых в одном архиве, постоянно и определено при его создании. По заполнении архива, запись просто начинается снова, причем старые данные стираются. Такое поведение и дает базе название циклической. Итак, каждый архив хранит данные за определенный период времени (возможно, день, неделю или год), заданный заранее.

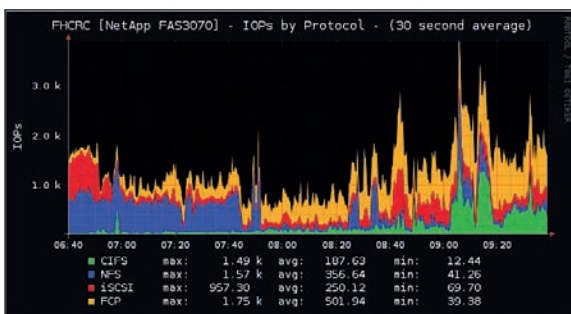
Например, для базы данных о температуре воздуха в моем парнике можно задать архив с вычислением среднего по каждому



10 измерениям (т.е. КТ будет возникать раз в 10 минут) и хранить 1008 значений; поэтому в любой момент времени у меня есть довольно подробный график температуры за последнюю неделю (в неделю 1008 периодов по 10 минут). Я мог бы сопроводить его архивом минимальных значений из 60 точек данных (одна КТ в час) и хранить 8760 значений, что даст мне менее подробный график, но зато за целый год. Также можно создать и третий архив, для максимальной температуры. Как видите, объем базы данных фиксирован и определяется в момент ее создания. В моем примере это $1008 + 8760 + 8760 = 18528$ консолидированных точек данных.

Как я уже говорил, заполнение базы данных ожидается через равные интервалы времени. Если результат измерения не пришел (например, компьютер, отвечающий за измерения, выключен), значение в базе данных помечается как Неизвестное (Unkown).

Классический график *RRDtool*; кривые независимых измерений отличаются цветом.



Снова несколько графиков – на сей раз один поверх другого, и можно видеть их сумму.

Черед прийти и уйти

Мы узнали, как данные попадают в базу данных *RRDTool*; но как их оттуда забрать? Конечно, можно экспортировать ее содержимое в текстовый файл (например, для переноса на другой компьютер), но красивый график гораздо интереснее. Графики строятся в форматах PNG, SVG, EPS или PDF, и их можно поместить на сайт, чтобы посетители просматривали их в браузере, или вставить в отчет об исследовании, который вы пишете в *OOo*. Возможности построения графиков в *RRDTool* обладают огромной гибкостью – взгляните на экранные снимки в галерее <http://oss.oetiker.ch/rrdtool/gallery>, чтобы оценить все их разнообразие.

Имеются команды для создания базы данных *RRDTool* (*rrdcreate*), добавления в нее измерений (*rrdupdate*), построения графика (*rrdgraph*) и решения других административных задач. Так что можно управлять базой данных вручную из командной строки (для тестирования или ознакомления) или написать для этого сценарий оболочки. Поддерживаются и другие языки – Perl, Python, Ruby, Lua и PHP.

Архив с исходными кодами можно загрузить с сайта <http://oss.oetiker.ch/rrdtool/pub>, там же можно найти скомпилированные пакеты для AIX, HP-UX, Solaris, Debian, RHEL, Fedora, Windows и даже NetWare. Найти эти пакеты можно в репозитории вашего дистрибутива, а установить через менеджер пакетов.

Счастливого графопостроения!

Что такое round robin?

Краснопузые птички [robin – англ. малиновка; round – круглый; round robin – карусель], поедующие червячков, которых я извлекаю, вскапывая весной огород, очень округлились из-за своего обжорства, но здесь речь не о них. Так называли петицию с подписями по кругу, как бы по спицам колеса, чтобы нельзя было определить главаря. С тех пор это стало прилагаться к любой идее возвращения на круги своя.

Утилиты копирования

Дублирование Следуя древней мантре Perl «есть более чем один способ сделать это», мы открыли множество вариантов копирования файла.

Мой урок этого месяца – подлинное возвращение к истокам и одна из самых фундаментальных тем, которых я касался. Он посвящен копированию. Утилита для этой цели больше, чем вы можете счесть нужным, и мы рассмотрим полдюжины из них.

Начнем с начала, с команды **cp**. В простейшем случае она выглядит так:

```
$ cp файл-источник файл-назначение
```

Если последний аргумент **cp** – имя существующего каталога, в него можно скопировать несколько файлов одной командой. Список файлов генерируется по шаблону. Например, я хочу скопировать картинки Джо в формате JPEG в свой каталог **images**:

```
$ cp ~/joe/images/*.jpg images
```

Для столь несложной команды у **cp** поразительно много опций. Я упомяну три.

Опция **-r** (рекурсивно) копирует каталоги рекурсивно. Она позволяет скопировать целое поддерево файловой системы. Например, в итоге команды

```
$ cp -r /boot .
```

в текущем каталоге окажется полная копия дерева каталогов **/boot**.

Опция **-a** (архивировать) сохраняет атрибуты исходного файла, насколько это возможно. Например, если файл копируется суперпользователем-**root**, будут сохранены исходные права доступа и время изменения файла. При копировании файла обычным пользователем владельцем файла станет этот пользователь. Опция **-a** подразумевает **-r**.

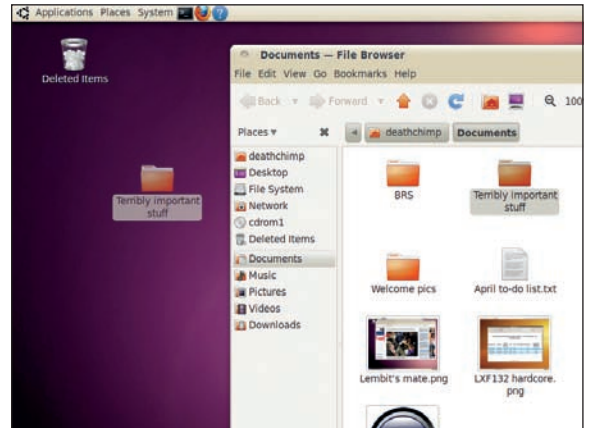
Опция **-b** (резервная копия) велит **cp** создать резервную копию при перезаписи существующего файла. К имени файла резервной копии добавится **~**.

Большой брат cp

scp (защищенное копирование) – старший брат **cp**. Этой командой можно копировать обычные файлы, как и **cp**, но основное ее назначение – копирование файлов с удаленных компьютеров по защищенному SSH-соединению. Выглядит она примерно так:

```
$ scp *.jpg james@servera.example.com:images
james@servera.example.com's password:
storm1.jpg      100% 198KB 197.7KB/s 00:00
sunset1.jpg     100% 135KB 135.2KB/s 00:00
teaching1.jpg  100% 190KB 190.1KB/s 00:00
thegirls.jpg   100%  57KB  56.9KB/s 00:00
UNIX_highlighter.jpg 100%  80KB  80.1KB/s 00:00
```

Здесь мои файлы JPEG копируются в каталог **images** компьютера **servera.example.com**. **james** – имя моей учетной записи на сервере; если его опустить, будет взято имя пользователя, под которым я работаю на клиенте. Обратите внимание на двоеточие –



► Файлы всегда можно скопировать через GUI, но это не спасет при удаленной работе на сервере без монитора.

это ключевой элемент конструкции, он сообщает команде, что я копирую файлы на удаленный компьютер. Имя каталога

images интерпретируется относительно домашнего каталога пользователя **james** на сервере. Точно так же можно загружать файлы с удаленного компьютера. Можно даже делать «копирование от треть-

его лица», когда и исходный, и результирующий файл находятся на удаленных компьютерах.

Чтобы это работало, на удаленных компьютерах должен быть установлен и запущен демон SSH, и на них у вас должны быть учетные записи.

У **scp** есть опция рекурсивного копирования (**-r**) и опция сохранения атрибутов (**-p**). Есть также опция **-C**, включающая сжатие передаваемых данных, хотя, по моему опыту, в довольно быстрой сети это скорее замедлит, чем ускорит.

rsync

Изначально, **rsync** – утилита удаленной синхронизации, хотя ею можно пользоваться и как (цитирую man-страницу) «улучшенной командой копирования для повседневного использования». Так что мы можем просто копировать ею локальные файлы:

```
rsync файл-источник файл-назначение
```

Но славится **rsync** как средство удаленной синхронизации. Ее алгоритм дельта-пересылки значительно снижает объем пересылаемых по сети данных, отправляя только различия между исходными файлами и файлами, существующими на получателе. Это великолепное средство резервного копирования и зеркалирования. По существу, вы говорите ей: «сделай эту файловую систему здесь точно такой же, как та вон там», и она делает это весьма эффективно. Синтаксис этой команды очень похож на синтаксис **scp**. Я могу скопировать весь мой архив статей *Linux Format* на удаленный сервер таким образом:

```
$ rsync -av Linux-Format-Articles servera.example.com:
... тут будет много наименований ...
sent 68556777 bytes received 14403 bytes 5079346.67 bytes/sec
total size is 68497001 speedup is 1.00
```

Вызов копиистам

Сколько еще вы вспомните команд, пригодных для копирования файлов? Сообщите нам, и мы опубликуем самые необычные варианты.

Если я сохраню файл, который сейчас пишу (он называется `copying.txt`) и повторю команду `rsync`, она завершится почти мгновенно, так как передает только различия между файлами:

```
$ rsync -av Linux-Format-Articles servera.example.com:
sending incremental file list
sent 21426 bytes received 161 bytes 4797.11 bytes/sec
total size is 68497303 speedup is 3173.08
```

При копировании на удаленные компьютеры `rsync` может использовать SSH или подключаться к демону `rsyncd`.

`rsync` сама по себе – не средство резервного копирования промышленного уровня, но резервное копирование с помощью `rsync` на USB-брелок – прекрасное решение для индивидуальных пользователей-профессионалов.

wget

`wget` – неинтерактивная утилита командной строки для копирования файлов с URL-адресов по протоколам FTP, HTTP или HTTPS. Файл записывается в текущий каталог. Вот пример:

```
$ wget http://archive.ubuntu.com/ubuntu/ls-IR.gz
--2010-06-14 13:43:57-- http://archive.ubuntu.com/ubuntu/ls-IR.gz
Resolving archive.ubuntu.com... 91.189.88.46, 91.189.88.30, ...
Connecting to archive.ubuntu.com... |91.189.88.46|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 9190681 (8.8M) [application/x-gzip]
Saving to: 'ls-IR.gz'
100% [=====>] 9,190,681
763K/s in 12 s
2010-06-14 13:48:05 (758 KB/s) – 'ls-IR.gz' saved
[9190681/9190681]
```

У `wget` тоже есть опция рекурсивного копирования (да, тоже `-r`), которая велит ей следовать ссылкам на HTML-страницах, по сути воссоздавая локальную копию структуры каталогов сайта. При этом содержимое копируется в каталог верхнего уровня с именем сайта. Например, команда

```
$ wget http://archive.ubuntu.com
```

скопирует содержимое сайта в каталог `archive.ubuntu.com` и будет делать это очень долго!

На практике `wget -r` работает только для статических сайтов, которых сейчас все меньше и меньше. Тем не менее, утилита очень удобна, когда нужно загрузить файлы с сайта из скрипта, или на сервере без браузера.

Из описанных утилит лишь `wget` работает только в одном направлении – с ее помощью можно скачать, но не закачать файлы. Однако для этой цели служит утилита `wput`, если вам интересно.

Причуды dd

Команда `dd` – безусловно, одна из самых странных bestий мира Linux. У нее причудливы и синтаксис, и набор возможностей. С ее помощью можно просто копировать файлы:

```
$ dd if=/boot/vmlinuz-2.6.32-22-generic of=foo
7886+1 records in
7886+1 records out
4037792 bytes (4.0 MB) copied, 0.0290523 s, 139 MB/s
```

но это слегка мимо кассы. Попутно утилита может выполнять всяческие нелепые преобразования данных: например, перевести нижний регистр в верхний (и наоборот) или переделать кодировку ASCII в EBCDIC (и наоборот). (EBCDIC расшифровывается как Extended Binary Coded Decimal Interchange Code – расширенный двоично-десятичный код обмена информацией, эта кодировка введена IBM и больше никем не используется). Она может поменять местами чередующиеся байты и выровнять текстовые строки по длине. Некоторые из ее функций явно нацелены на миграцию данных между Unix/Linux и традиционными компьютерами IBM. Но, пожалуй, самая полезная возможность `dd` – кон-

Что в имени тебе моем?

`dd` берет название из языка управления заданиями (JCL) мейнфреймов IBM – это сокращение от “data definition” [определение данных]. Оттуда же пришел и странный синтаксис, столь неудобный в командной строке Linux. Выглядел он примерно так:

```
//QSAM0080 DD DSN=FOOBAR.DATA.QSAM0080,DISP=OLD,KEEP
```

Я думаю, что смог подавить большую часть своих воспоминаний о борьбе с JCL в мои студенческие годы перфолент и перфокарт. Мне обычно гораздо труднее давались полдюжины строк на JCL, чем следующие за ними 200 строк на Fortran. Обычно я получал распечатку об ошибке (учтите, выдача была раз в полдня), гласившую «отсутствует запятая в строке 3». Много раз я дивился: почему, зная, что там не хватает запятой, компьютер не может просто представить, что она там есть, и двинуться дальше?

троль размера блоков, в которых будет считываться входной файл и записываться выходной, а также возможность задания числа копируемых блоков. Это дает почву для интересных экспериментов. Например, файл размером 1 МБ, полный нулей, можно создать так:

```
$ dd if=/dev/zero of=allzeros bs=1M count=1
1+0 records in
1+0 records out
1048576 bytes (1.0 MB) copied, 0.00476925 s, 220 MB/s
```

Здесь мы записываем один блок размером 1 МБ. Стиль указания файла-аргумента `if=/dev/zero` совершенно чужд Linux и не поддерживает спецсимволы (шаблоны). Обратите внимание на желание `dd` сообщать вам о том, какой объем данных скопирован и насколько быстро это произошло. Утилита также способна клонировать диск. Здесь, например, мы создаем образ раздела `/dev/sda1` и записываем его в файл:

```
$ sudo dd if=/dev/sda1 of=sda1-image bs=64M
1+1 records in
1+1 records out
123346944 bytes (1.0 MB) copied, 0.00476925 s, 220 MB/s
```

Незванный гость

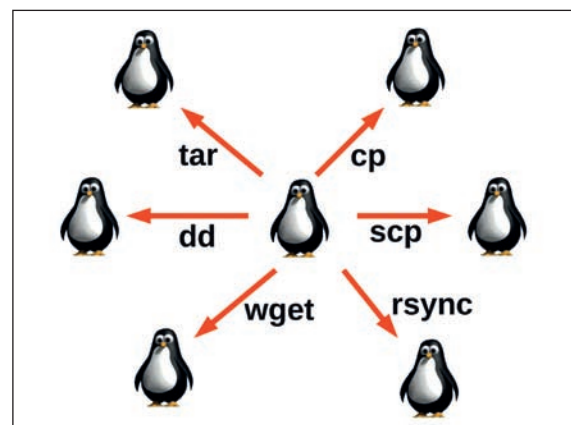
Последнее явление на вечеринке утилит копирования смахивает на незваного гостя: это скорее утилита архивирования, чем копирования. Зовут ее `tar`. Однако, направляя данные в `tar` и получая их оттуда, можно копировать иерархии файлов. Трюк состоит в том, чтобы указать ‘-’ в качестве имени выходного или входного файла, что заставит `tar` писать в стандартный вывод или читать со стандартного ввода. Здесь я копирую свой каталог статей `Linux Format` в каталог `/tmp`:

```
$ tar cf - Linux-Format-Articles/ | (cd /tmp; tar xf -)
```

Точно так же можно создать канал через SSH-соединение и скопировать файлы на удаленные компьютеры:

```
$ tar cf - source-dir | ssh servera.example.com tar xf -
```

Держу пари, что вы до этого не додумались! **LXF**



► Пингинов столько, что всех и не перевешаешь. А надо ли вешать пингинов?

IT-архитектура вашего бизнеса



119991 Москва, ул. Губкина, д. 8
Телефон: +7 (495) 232-00-23
Электронная почта: info@softline.ru
Сайт: www.softline.ru

softline[®]

19 стран, 61 город





Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Упомянул PostgreSQL на защите кандидатской диссертации по физике элементарных частиц.

Перспективные проблемы

Текущие проблемы не должны затмевать долговременные перспективы.
Николя-Поль-Стефан Саркози де Надь-Боча

Создаётся ощущение, что компания Dell сама не знает, чего хочет. Прямо детектив какой-то. Естественно, основные действия упрятаны от глаз наблюдателей, но кое-что становится открытым умеренно пытливого взору.

В середине июня на сайте Dell появилась рекламка Ubuntu. Не программная начинка – просто песня. Вот только DRM нет, а иначе был бы полный «энтерпрайз». Месяцем позже страничку, кричащую о преимуществах дистрибутива GNU/Linux, подредактировали, подкинув в бочку приторного мёда пару ложек дёгтя, а в конце июля вообще полностью свернули продажи предустановленных Ubuntu через сайт компании. Мотивировалось это тем, что квалификация у желающих установить на ноутбук GNU/Linux достаточна, чтобы сделать это по телефону.

Но прошла неделя, и Dell возобновила продажу компьютеров с предустановленным дистрибутивом Ubuntu через свой сайт. Там даже новая модель ноутбука появилась. Что это? Война миров или просто мышиная возня за кусочек сыра? Время покажет, но что-то безусловно происходит.

P.S. Продажи коммуникаторов с Android рвутся в небеса. Конечно, это не настольный GNU/Linux, но тоже неплохо.

P.P.S. Лето в европейской части России прошло под знаком пожаров. Не разжигайте костров или как минимум тушите их. И – да, курить вредно.

E.m.Baldin@inp.nsk.su

В этом месяце вы научитесь...



Тренировать мышей 58
Жесты – лучший способ научить старого грызуна новым трюкам. А еще **Энди Ченнел** разберется с фотографиями в *Shotwell*.



Ускорять OpenOffice.org 62
Мы не можем заставить время течь быстрее, но можем составлять таблицы так, чтобы они требовали меньше времени, считает **Александр Маджугин**.



Мерить скорость 66
Насколько быстр ваш компьютер с Linux? **Боб Мосс** представляет *Bonnie++* и *Phoronix Test Suite*, дающие научный ответ на этот вопрос.



Исправлять фото 68
Для этого есть специализированные приложения, но если вы не боитесь математики, напишите под руководством **Михаила Смирнова** свое собственное.



Применять текстуры 72
Вячеслав Ястребцев завершит беглое введение в *POV-Ray* созданием пейзажа... натюрморта... в общем, воды и шестеренок.



Отслеживать ошибки 76
Без чего немислим ни один программный проект? Правильно – без ошибок. **Никита Шультайс** и *Mantis* помогут следить за этим ценным ресурсом.



Писать сценарии 80
К кинематографу это не имеет никакого отношения: **Ник Вейч** начинает новую серию о языке оболочки *Bash* с файлов и каталогов.



Сортировать почту 84
Неразобранные, захламленные почтовые ящики бояться *Procmail* как огня. **Шашанк Шарма** использует его, и делает жизнь в 457 раз легче.

Совет месяца: На этом диске...



Подписывать диски когда-то было очень распространенной практикой. Как правило, требуемый реквизит включал диск (точнее, дискетку), залапанную наклейку и замусоленный карандаш – кое у кого все это до сих пор ассоциируется с обменом играми Amiga в школе. Но на современных USB-брелках и картах памяти едва ли что-то нацарапашь, так что о содержимом носителя остается только догадываться. Даже будучи подключен к компьютеру, он появляется в системе как безликий `/media/disk`, без всякого намека на содержимое.

А все потому, что ядро не знает, как называется диск. Это исправимо – вы можете назначить осмысленную метку файловой системе. Форматируя

FAT – самый распространенный выбор для флэш-накопителей – используйте `mkfs.vfat -n Music /dev/sdb1`

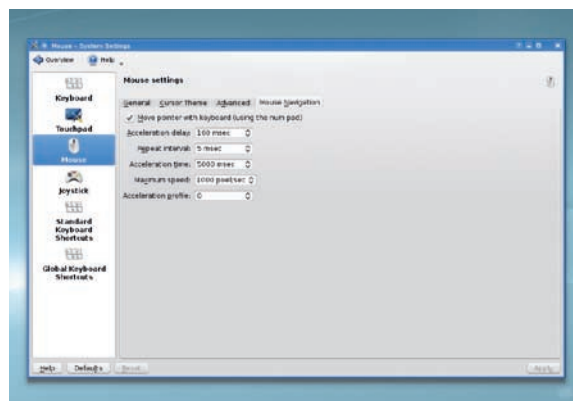
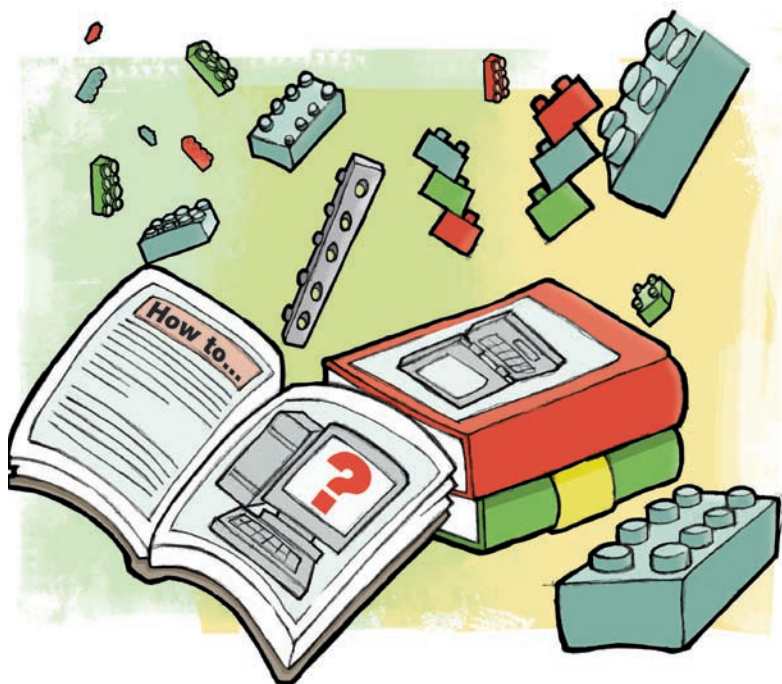
Теперь карта смонтируется в `/media/Music`, что мгновенно скажет о ее содержимом. Камеры также часто снабжают свои карты меткой при форматировании, но метка-то все время одна и та же, и это не слишком информативно. Так как рекомендуется дать устройству самому отформатировать карту «под себя», описанный выше способ не сработает, но можно снабдить меткой и готовую файловую систему:

```
dosfslabel /dev/sdb1 photos1
```

Если выполнить эту команду с именем устройства, но без метки, она выведет существующую метку.

МЫШЬ: РАСШИРИМ

Мышь умеет не только указывать, щелкать и прокручивать. Энди Ченнел снимает завесу тайны с других полезных функций этого друга рабочего стола.



» Если у вас сбоят двойной щелчок при темпе по умолчанию, измените настройки своей мыши.

медленной. Используйте стрелки вверх/вниз, чтобы делать меньше движений мышью при большем эффекте. Нажмите Применить для сохранения изменений.

Значение Пороговой величины [Point Threshold] определяет размер активной области вокруг указателя курсора. Увеличить ее значение полезно для тех, кто не является метким мышестрелком, поскольку это создает иллюзию, что объекты рабочего стола увеличены. Таким же образом, если вы не очень ловко управляетесь с нажатием кнопки, Интервал двойного щелчка [Double-Click Interval] можно использовать для замедления скорости двойного нажатия и упрощения управления мышью. Чем выше значение этого числа, тем больше времени отводится вам на нажатие. Неумеренное увеличение этого значения может привести к тому, что два независимых щелчка будут расцениваться системой как двойное нажатие.

Опция внизу полезна для быстрой прокрутки документов или web-страниц. Большие цифры означают более быструю прокрутку.

Заменитель клавиатуры

Также мышь, до известной степени, может заменять клавиатуру. Отметим, что мышью вряд ли удобно вводить большие тексты, но иногда такой ввод может оказаться полезным. Нажмите на иконку KDE и в поисковом окне в панели сверху введите **Kvkbd** или **virtual keyboard** [виртуальная клавиатура]. Вы получите один результат вашего поиска. Нажмите на него для запуска виртуальной клавиатуры.

В зависимости от конфигурации вашего рабочего стола, клавиатура появится внизу справа или, если она минимизирована, в системном лотке. В последнем случае нажмите на иконку, чтобы вывести клавиатуру на экран. Вы можете перетащить ее мышью на любое удобное место. Потянув за красную секцию в самом низу интерфейса, можно изменить размер клавиатуры.



Наш эксперт

Энди Ченнел

Энди, видимо, будет делать Первые шаги в Linux вечно, а технологиями он заинтересовался, открыв для себя Dragon 32.

Появление мыши было революционным, но повседневное употребление отчасти лишило ее магических свойств. Мы научим вас творить мышью чудеса под KDE. Большинство из того, что мы сделаем на этом уроке, достигается через диалоги Параметров системы KDE; их можно открыть через *KPanel*, выбрав Компьютер > Параметры системы [Computer > System Settings]. Во вкладке Общие [General] войдите в Администрирование [Administration], а затем в Клавиатура и Мышь [Keyboard & Mouse], после чего выберите Мышь в панели слева.

Здесь представлены базовые настройки, такие как переключение клавиш для левшей, открытие папок одним или двумя щелчками, а также реверсивная прокрутка колесом мыши.

В Теме курсоров [Cursor Theme] можно поменять внешний вид курсора. Для выбора одного из имеющихся, выделите его и кликните Применить [Apply]. Изменения вступают в силу незамедлительно. Прекрасную коллекцию новых тем для курсоров можно найти по адресу www.kde-look.org в секции X11 Mouse Themes. Любую понравившуюся можно скачать и установить с помощью кнопки Установить новую тему [Install New Theme]. Новая тема (или темы) появятся в списке, и их можно выбрать привычным способом.

Теперь перейдем на вкладку Дополнительно [Advanced]. Это диалог не визуальный, и придется поработать с цифрами, но это довольно просто. Три вещи, которые нам будут интересны, находятся наверху.

Начните с ускорения движения мыши по рабочему столу. Это удобно, если у вас большой монитор и реакция мыши кажется

» Месяц назад Мы сохраняли файл в облаке и настраивали горячие клавиши KDE.

ее ГОРИЗОНТ

Для использования, просто щелкните мышью в любом месте, где нужно ввести текст, а затем нажимайте на клавиши, чтобы его напечатать. Клавиатура будет оставаться поверх всех других окон, что обеспечивает легкий доступ к ней. Кнопка рядом с Минимизировать (красный крест в верхнем правом углу) в виде гаечного ключа открывает меню настройки, позволяющее закрепить клавиатуру по нижнему краю окна, изменить шрифт на клавишах или закрыть приложение. Иконка внизу вызовет цифровую клавиатуру.

Жесты мышью

С помощью системы жестов мышью также можно «узурпировать» некоторые клавиатурные команды. Например, можно контролировать кнопки Назад и Вперед в web-браузере *Konqueror*, не снимая руку с мыши. Такое поведение мыши, если вы не знали, что система активирована, может изрядно озадачить, поэтому разработчики KDE гуманно оставили ее по умолчанию выключенной.

Для запуска Жестов мыши [Mouse Gestures] перейдите в главное окно Параметров системы и выберите иконку Действия ввода [Input Actions] в Администрировании. Тут вы увидите, что несколько действий уже заданы по умолчанию: например, создание снимка экрана при нажатии **Print Screen**. Некоторые жесты заданы, но не активированы. Начните с нажатия на иконку Жесты [Gestures]. Заметим, что вернуться к этому экрану можно из окна Действия ввода нажатием на кнопку Настройки [Settings] внизу окна.

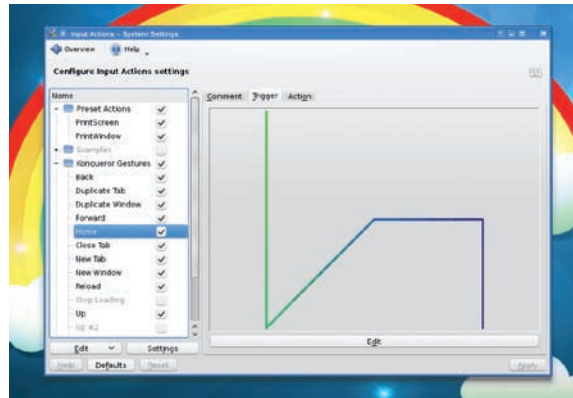
После нажатия Жесты, нажмите Применить, а потом уж переходите к определениям жестов, иначе ваши изменения не сохранятся. Теперь нажмите на иконку *Konqueror Gestures*. Выберите одно из действий в списке, чтобы посмотреть ассоциированный с ним жест. Например, кнопку Назад в *Konqueror* можно активировать, нажав левую кнопку мыши, двинув мышшь влево и отпустив кнопку.

Вы увидите, что этот жест описан во вкладке Комментарии [Comment], а взглянув на вкладку Триггер [Trigger], вы увидите его визуальное представление. В большинстве случаев, нужно начинать от самой светлой зеленой точки и затем двигаться в направлении самой темной синей.

Можно отредактировать и сам жест, перейдя на вкладку Действия и выбрав сочетание клавиш, которому жест соответствует.



➤ Для компьютеров типа «киоск» виртуальная клавиатура подойдет лучше обычной.



➤ Жесты мыши создают новые способы контроля браузера.

Также можно отредактировать жест, нажав на клавишу Редактировать [Edit] во вкладке Триггер, изобразив новый жест в окне, который появляется. Помните, что полезнее простые жесты, поскольку более сложные трудно запомнить и выполнить.

И, наконец, есть полезная функция KDE Hot Corners, позволяющая запускать команды, просто оставляя мышшь на краю экрана. Это полезно для запуска экранной заставки, но может мешать, если вы не самый аккуратный пользователь мыши в мире.

Для доступа к этой функции, вернитесь на экран Параметров системы и выберите Рабочий стол [Desktop]. Затем в полученном списке вариантов выберите иконку Края экрана [Screen Edges]. В центре этого окна вы увидите анимированное изображение экрана с маленькими круглыми кнопками в каждом углу и с каждого края. Нажмите на любую из них, и появится сервисное меню с большим спектром доступных опций, таких как Показать рабочий стол [Show Desktop] или Имеющиеся окна [Present Windows]. Если у вас запущен *Compiz*, для него также найдется несколько настроек. **LXF**

Трекпады

Если вы работаете на ноутбуке, вы видите раздел Тачпад [Touchpad, координатная площадка]. Доступные возможности зависят от конкретного компьютера, но мы рассмотрим основные.

Во вкладке Общие [General] вы найдете опции включения и выключения трекпада, информацию и слайдер для установки чувствительности к нажатию.

В разделе Прокрутка [Scrolling], вероятно, найдутся параметры для подключения прокрутки в режиме multi-touch, горизонтальной прокрутки и прокрутки накатом (кинетической), а также слайдеры для задания чувствительности инструмента.

В опциях Нажатие [Tapping] можно включить нажатие полностью, что довольно полезно, если у вас тяжелая рука. Также мож-



➤ Трекпады тоже можно настраивать.

но определить жесты multi-touch и нажатия в определенных положениях (например, нажатие в верхнем правом углу для имитации правого щелчка), и установить интервал двойного щелчка, как у мыши.

Shotwell. СНИМКИ

Нужна помощь в организации ваших летних фотографий? *Shotwell* готов.

Снова пришло светлое время года, когда пальцы так и тянутся к спусковой кнопке фотоаппарата на всяких спортивных праздниках, пляжах, фестивалях и прочих мероприятиях. Управляться со всей этой массой цифровых данных так же важно, как и изначально делать хорошие снимки, потому что даже от самой красивой фотографии мало проку, если она неизвестно где.

На данном уроке мы установим и разберемся с *Shotwell*, новым приложением в области управления фотографиями, который способен вытеснить *F-Spot* во многих дистрибутивах на базе Gnome, выходящих в следующем году. Мы будем организовывать фотографии на основе событий, избранных и тэгов, а затем посмотрим, как придать вашим снимкам больше блеска и подготовить их к показу всему миру или просто вашей семье.

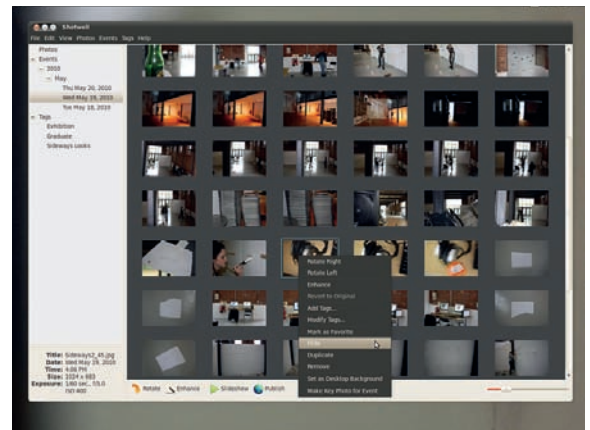
Пользователи Fedora и Ubuntu могут установить приложение из соответствующих менеджеров пакетов, но проще открыть терминал и добавить его командой. В Fedora 12 и 13 можно набрать `pkcon install shotwell`, а в последней версии Ubuntu (10.04/Lucid Lynx) – `sudo apt-get install shotwell`.

Для установки в Karmic, добавьте репозиторий командой `sudo add-apt-repository ppa:yorba/ppa` затем обновите его командой `sudo apt-get update` и только после этого командуйте `sudo apt-get install shotwell`.

Установка в OpenSUSE немного более проблематична, поскольку в дистрибутиве отсутствует пакет *libgee*, от которого зависит *Shotwell*. В любом случае, есть RPM-пакет *Shotwell*, доступный по адресу <http://software.opensuse.org>. После его установки вам потребуется установить *libgee* из <http://download.gnome.org/sources/libgee/0.5/libgee-0.5.tar.bz2>.

Результатом всех вышеописанных действий будет новая строчка в разделе Графика [Graphics] вашего меню приложений.

При первом запуске программы вам предложат добавить содержимое. Это можно сделать, перейдя в Файл > Импорт [File > Import] и указав путь к папке с фотографиями, перетаскивая



➤ Если вам не хочется отображать фотографию, но жаль ее выбросить, нажмите Спрятать [Hide] для удаления ее из библиотеки.

файлы в главное окно *Shotwell*, либо подсоединив вашу камеру и импортируя прямо с нее.

После импорта фотографий вы увидите две новые записи в панели слева – там, где вы управляете вашей библиотекой на макроуровне. Выберите верхнюю строчку для просмотра всех ваших снимков как гигантской галереи с последними фотографиями наверху.

Информационная станция

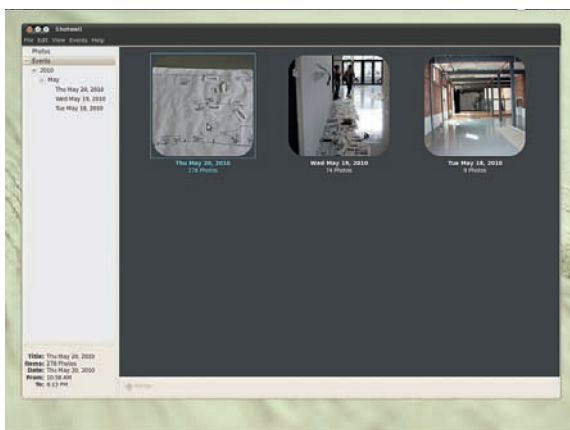
Двойным нажатием на изображение в этом режиме можно вызвать его большую версию, а затем снова нажать два раза для возврата в вид галереи. В основании панели навигации вы увидите информацию, полученную из Exif вашей фотографии, с такими подробностями, как дата съемки, разрешение снимка и иногда сведениями о выдержке и географическом положении снимка.

Вторая запись в панели навигации – События [Events]. Выберите ее, чтобы посмотреть вашу коллекцию фотографий, отсортированных по событиям на основании даты их создания. Щелкните на событии два раза, чтобы посмотреть соответствующие ему фотографии. Можно также создать дерево событий, нажав на иконку (+) рядом с указателем.

События можно отсортировать из главного окна. Например, у вас есть масса отпускных фотографий, но они отнесены к разным Событиям, поскольку вы сделали их в разное время. Чтобы объединить их, выберите первое событие, затем зажмите Shift и нажмите на другие события для объединения. Теперь нажмите правой клавишей мыши на выбранных и выберите Объединить Событие(я) [Merge Event(s)].

При переходе на событие, вы увидите маленькую иконку Слайд-шоу [Slideshow] у основания окна. Выберите ее для полноэкранный слайд-шоу фотографий из этой секции.

➤ Используйте секцию События для просмотра ваших фотографий, отсортированных согласно датам или событиям.



➤ Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

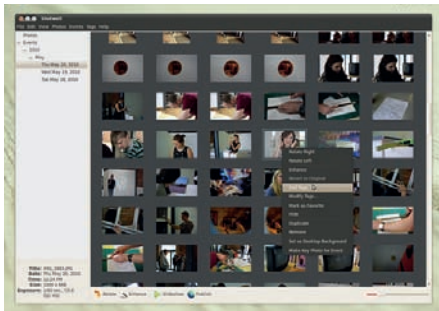


В порядке



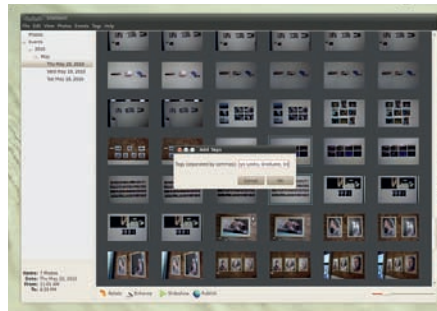
Шаг за шагом: Редактируем и загружаем библиотеку

Умеющий добавлять тэги, создавать категории, редактировать, кадрировать и даже выгружать фотографии на ваш любимый сайт, *Shotwell* – находка для фотографа.



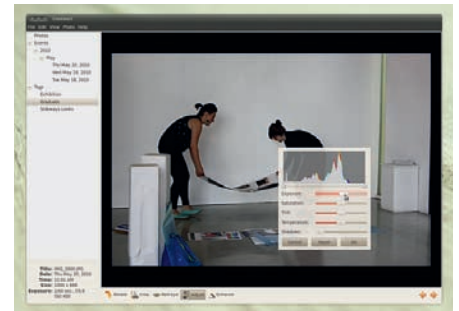
1 Найдите свое фото

Используйте опцию События или Фотографии [Photos] для просмотра вашей коллекции. Выберите отдельное изображение, нажав на него левой клавишей мыши, выберите несколько фотографий подряд, нажав на первую и затем на последнюю, удерживая Shift, или выберите несколько отдельных фотографий, удерживая Ctrl. Вы можете снять выделение всей выборки или отдельных фото, нажав на них еще раз.



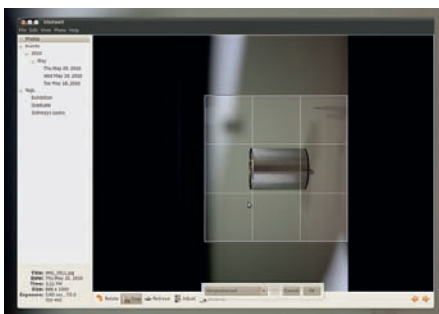
2 Добавьте тэги

События – это прекрасный вариант сортировки, но иногда удобнее добавить Тэги [Tags] для организации групп снимков. Нажмите правой клавишей мыши на выделенных фото и выберите Добавить тэги [Add Tags]. Используйте различные описательные тэги, разделяя их запятыми. После нажатия ОК вы увидите добавленный к фотографиям текст; тэги также появятся в панели навигации.



3 Отредактируйте фото

Нажмите два раза на фотографии для предпросмотра и редактирования ее в полноэкранном режиме. У основания окна есть пять инструментов базового редактирования изображения. Для решения проблем с выдержкой или балансом белого, выберите Корректировать [Adjust] и затем отредактируйте изображение с помощью различных слайдеров. Можно отменять сделанные изменения и даже использовать Фото > Отменить [Photo > Revert] для возврата к исходному снимку.



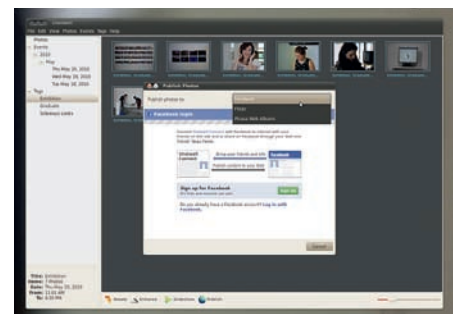
4 Обрежьте кадры

Кадрирование – это искусство, и *Shotwell* предлагает вам несколько инструментов для него. Выберите Вырезать из меню Инструменты и двигайте окно за края или углы, чтобы выбрать нужную область. Сетка, которую вы видите, относится к «правилу третей» и является полезным композиционным инструментом. Также можно выставить соотношение сторон, выбрав один из предустановленных вариантов из выпадающего меню внизу окна.



5 Экспортируйте в файл

Выбрав коллекцию фотографий, перейдите в Файл > Экспорт [File > Export] для открытия диалога экспорта. Здесь можно установить качество JPEG-файла (чем выше качество, тем больше размер файла) и выбрать изменение размера фотографий на выходе. Нажмите ОК и укажите местоположение для экспорта. Нажмите Сохранить для сохранения новых копий изображений.



6 Опубликуйте в Интернет

Ко всему хорошему, вы можете еще и опубликовать фотографии из *Shotwell* на Flickr, Facebook или Picasa. Выберите снимки для публикации, затем перейдите в Файл > Опубликовать и выберите сервис в выпадающем списке. Вам предложат ввести имя и пароль или создать новую учетную запись перед тем, как изменить размер фотографий (если надо) и опубликовать их на вашем сервисе. **LXF**

OOo Calc:

Напоследок **Александр Маджугин** рассмотрит сложные функции, которые используют в своей работе данные из множества ячеек, в том числе – рекурсивно.



На рис. 1 показан конвейер вычисления третьего в ряде значений технического индикатора ADX. Видно, что для получения результата потребовалось предварительно вычислить 33 значения в других ячейках, а ведь это ещё не самый сложный расчёт.

Измеряем скорость

Как и при оптимизации пользовательских функций, написанных на Basic, здесь нам потребуется инструментарий, как минимум для замера скорости вычислений в конвейере. Однако таким же простым решением, какое мы использовали для замера времени работы кода в **LXF134**, нам уже не обойтись.

Формализуем задачу. Итак, нам необходимо замерить время расчёта некоторой связанной последовательности формул, для чего потребуется засечь время инициации расчёта и время изменения ячейки, возвращающей результат вычисления. Это значит, что понадобятся две процедуры – для запуска расчёта и для регистрации его завершения соответственно. *OpenOffice.org* – не Perl, но и в нём для решения задачи есть более одного пути; мой представлен на Листинге 1 на диске. Для большего удобства я написал для себя простенький шаблон, который также можно найти на DVD в файле **CalcSpeedTester.ods**.

Смысл работы кода в следующем: стартовая процедура изменяет содержимое ячейки, которая иницирует вычисление в конвейере, и сохраняет время этого события в глобальной переменной `starttimepoint`. Когда изменяется содержимое последней ячейки конвейера, выполняется функция `CalcTimerStop`, вычисляющая время, прошедшее с последнего изменения глобальной переменной `starttimepoint`.

Для использования шаблона следует ввести два различных значения из области определения конвейера в ячейки A1 и A2 листа «Тестируемая последовательность». Это необходимо для того, чтобы значение ячейки, на которую ссылается конвейер, изменялось при каждом нажатии кнопки Старт, находящейся на листе «Тест». Значения из ячеек A1 и A2 будут чередоваться.

Сам конвейер должен ссылаться на ячейку A3 (start), а в ячейке A4 (finish) должна быть ссылка на последнюю изменяющуюся ячейку последовательности. Конвейер можно разместить тут же – на листе «Тестируемая последовательность».

В предыдущих статьях этого (к сожалению, подходящего к своему логическому финалу) цикла мы рассмотрели вопросы создания, использования и оптимизации пользовательских функций в *OpenOffice.org Calc*. Данный же урок будет посвящён оптимизации конвейеров вычислений непосредственно в самой электронной таблице, а также общей оптимизации электронной таблицы, способной значительно улучшить работу с ней.

Что я называю здесь конвейером вычислений? Конвейер вычислений – это последовательность формул, каждая последующая из которых использует как свой аргумент результат предыдущей. Естественно, что для достижения необходимого результата могут быть использованы конструктивно различные конвейеры. И столь же естественно, что скорость вычисления в них также будет различна, причём, как мы увидим далее, отличаться она может достаточно сильно.

» Рис. 1. Конвейер вычисления ADX насчитывает 33 «этапа» (синие стрелки), и это ещё далеко не предел.

Данные						Направленное движение				Истинный диапазон	Индикаторы направления				Направленное движение	Среднее направленное движение
<DATE>	<TIME>	<OPEN>	<LOW>	<HIGH>	<CLOSE>	+DMpre	-DMpre	+DM	-DM	TR	+SDI	-SDI	+DI	-DI	DX	ADX
04.12.2008	11:00:00	57,1	56,03	57,39	56,5											
04.12.2008	12:00:00	56,65	56	57,51	56,8	0,12	0,03	0,12	0,00	1,51	0,08	0	0,08	0	100	100
04.12.2008	13:00:00	56,7	56,4	56,90	56,5	0	0,00	0,00	0,00	0,50	0	0	0,07	0	100	100
04.12.2008	14:00:00	56,41	56,4	56,94	56,94	0	0,00	0,00	0,00	0,54	0	0	0,07	0	100	100
04.12.2008	15:00:00	56,7	56,55	56,97	56,61	0,03	0,00	0,03	0,00	0,42	0,07	0	0,07	0	100	100
04.12.2008	16:00:00	56,78	56,39	56,89	56,4	0	0,16	0,00	0,16	0,50	0	0,32	0,06	0,02	47	100

» Месяц назад Мы оптимизировали функции, которые сами и писали.



На конвейере

finish		A	B	C	D	E	F	AZ	BA	BB	BC
1	Вход 1		1		1	5050	10099	242353	247402	252451	
2	Вход 2		2		2						
3	Старт		3		3						
4	Финиш		4		4						
5			5		5						
99			99		99						
100			100		100						
101											
102											

Рис. 2. Последовательность конвейеров SUM.

Рассмотрим простой пример – тест функции SUM в сравнении с простым суммированием значения каждой ячейки. Для увеличения времени расчёта данных должно быть много, а сам конвейер лучше повторить несколько раз с последовательными ссылками следующего конвейера на предыдущий.

На рис. 2 показана схема связей тестируемой последовательности. Столбцы с G по AY и строки с 6-й по 98-ю скрыты для экономии места. Ячейки в строках со 2-й по 100-ю содержат простые числовые значения от 2 до 100 соответственно. Ячейка в строке 1 каждого конвейера содержит ссылку на результат предыдущего, чем достигается последовательное вычисление функции SUM. Формулы в строке 101 имеют вид =SUM(#1:#100), где # – имя соответствующего столбца.

Время выполнения этой схемы на моём ноутбуке составило около 7 миллисекунд – этого, как правило, недостаточно для надёжного теста, и последовательность стоило бы увеличить, но в данном случае нас это устраивает, так как для формул вида =#1+#2+...+#99+#100 время расчёта составляет около 170 миллисекунд, что почти в 25 раз дольше.

Таким образом мы приходим к первому правилу оптимизации вычислений в Calc – везде, где это возможно, стоит использовать встроенные функции пакета, а не их реализацию в виде формул, с помощью более простых функций и операторов.

Впрочем, данный приём элементарен и очевиден, а потому малополезен, но зато он хорошо показывает, какими могут быть потери времени при неправильном проектировании конвейеров вычислений.

Другим интуитивным способом ускорения расчётов в Calc, по-видимому, является сворачивание конвейера в минимум формул. Под этими словами понимается такое преобразование формул в последовательно ссылающихся друг на друга ячеек, при котором часть из них встраивается в зависимые формулы для сокращения количества ссылок в конвейере. Например последовательность формул вида C1=A1+B1, D1=C1*32, E1=If(D1>100;1;0), очевидно, приводится к виду E1=If((A1+B1)*32>100;1;0), то есть сворачивается в одну-единственную формулу. Насколько это эффективно?

Для проверки этого приёма в базовом, неоптимизированном представлении, я использовал последовательность из 14 последовательно связанных формул, среди которых были выражения,

содержащие самые обычные арифметические действия, функции условий IF и CHOOSE, а также функции для работы с текстом и функции преобразования типов. В виде, свёрнутом в минимум последовательных формул, они представляли собой лишь три последовательно связанных выражения – D2=COUNTIF(F1;"<0")+1 (F1 – результат вычисления в предыдущей последовательности), E2=IF((D2*3+2)/4=2;56;2) и F2 =VALUE(RIGHT("00"& TEXT(SQRT(E2)*E2);"000000");2))-14.

В результате выигрыш по времени расчёта составил чуть более 6%; откровенно говоря – не так уж и много. Однако если взять ту же последовательность и опустить традиционно долгие операции по преобразованию типов и, соответственно, текстовые операции, то это приводит не только к троекратному росту скорости вычисления данной последовательности, но и примерно к такому же росту выигрыша от оптимизации – до 15%.

Из данного опыта можно сделать вывод, что такой тип повышения производительности если и стоит использовать, то только для математических расчётов и уменьшения размера итогового файла, так как меньшее количество задействованных в расчёте ячеек часто ведёт к сокращению его размеров. В данном случае оптимизированный файл стал (по сравнению с исходным) меньше в два с лишним раза (27 КБ против 65 КБ). А уменьшение размеров файла ведёт к значительному сокращению времени его открытия в OpenOffice.org Calc.

Ещё одним, пусть и менее известным, методом оптимизации как скорости расчёта, так и размера файлов считаются формулы массива. За подробностями о том, что это такое, вы можете обратиться к встроенной справке пакета OpenOffice.org – здесь я не стану описывать их, так как они того не очень-то и заслуживают. Почему я так считаю? Вот об этом-то мы и поговорим подробнее, а заодно постараемся предостеречь вас от излишних трудностей с построением вычислений в надежде на их оптимальность.

Считается, что формулы массива дают при расчётах значительную экономию времени в силу двух причин: во-первых, они потребляют существенно меньше оперативной памяти, во-вторых, при получении данных из влияющих диапазонов извлекают их не по одной ячейке, а сразу целыми диапазонами, на которые ссылаются.

Однако первое критично исключительно для систем с небольшим объёмом ОЗУ (конечно, всё зависит от сложности расчётов,

«Везде, где возможно, используйте встроенные функции пакета.»

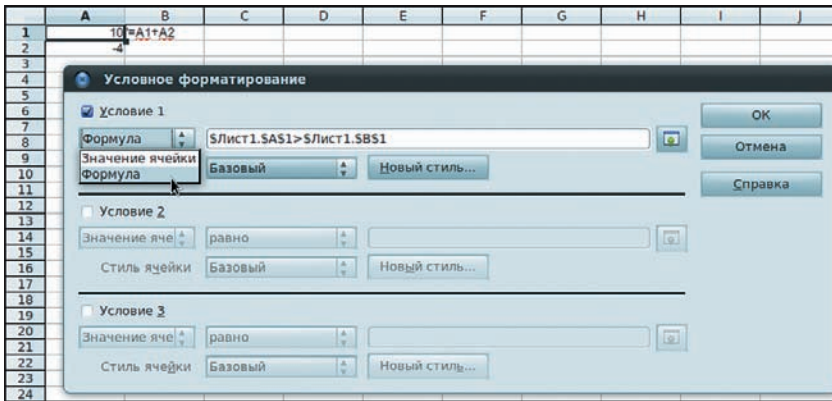


Рис. 3. Условное форматирование ячейки возможно как на основании ее значения, так и по формуле.

но я бы сказал, что даже при очень больших объемах данных это не станет критичным фактором, если у вас больше 512 МБ), а второе характерно только для старых версий пакета – в *OpenOffice.org* от 3.0 и выше извлечение данных из влияющих ячеек значительно оптимизировано, и формулы массивов уже не дают того прироста производительности, какой был раньше.

Единственным преимуществом массивов остаётся значительное, хотя и несколько меньшее, чем при сворачивании конвейеров, уменьшение размера файла, что, как уже говорилось, благотворно сказывается на скорости открытия документа.

Так что же, единственным методом оптимизации, дающим ощутимый прирост в скорости вычислений, является избегание воссоздания функций, встроенных в *OpenOffice.org Calc*? Нет, конечно – можно порекомендовать ещё несколько достаточных стандартных правил:

- » Не делать лишних вычислений и следить за тем, чтобы алгоритм решения был оптимальным.
- » Избегать повторных вычислений (например, если у вас есть формула вида $E1=If(A1*B1+C1>0;A1*B1+C1;A1*B1+C1+D1)$, то её лучше разбить на две – $E1=A1*B1+C1$ и $F1=If(E1>0;E1;E1+D1)$, чтобы не вычислять $A1*B1+C1$ дважды для каждой строки).
- » При преобразовании типов избегайте явного преобразования, особенно когда это касается преобразования текста в число. Это может звучать несколько странно и непривычно для тех, кто знаком с программированием, но это справедливо для *Calc* – используя функцию `VALUE` (например, `=VALUE(A1)*5` вместо `=A1*5`), вы можете потерять до 15% производительности.

Я не буду приводить здесь описание подробного тестирования каждого из приёмов – размер статьи всё же ограничен, а результат применения может очень сильно зависеть от конкретного решения. При этом вы всегда можете сами протестировать любое решение на предмет производительности, используя находящийся на диске файл **CalcSpeedTester.ods**.

Применение описанных выше правил поможет вам сделать расчёты в *Calc* несколько более быстрыми, а работу комфортнее, но это ещё далеко не предел.

Визуализация и форматирование

Расчёты в электронных таблицах обычно не ограничиваются самими вычислениями – им сопутствует ещё и визуализация результата, от простого отображения чисел до сложного условного форматирования. И нет ничего неожиданного в том, что способ отображения может влиять на скорость вычислений, причём иногда очень сильно, а иногда и... странно.

Естественно, что разобраться в большом объёме данных зачастую довольно трудно, и в глазах рябит от цифр – трудно понять, где экстремумы функций, где положительные значения сливаются с отрицательными и так далее. В таком случае на помощь приходят возможности условного форматирования. Однако это, казалось бы, никак не связанное с вычислением результата возможность может, в некоторых случаях, замедлить его получение, а иногда и... ускорить его!

Как вы, наверное, знаете, существуют два способа задания условия форматирования ячейки с результатом – первый основан на сравнении значения ячейки с константой или значением другой ячейки, а второй основан на вычислении истинности формулы; рис. 3. Так вот, в использовании этих методов наличествует значительная разница. Первый из них совсем не страшен: он почти не снижает скорость расчётов, более того, при некоторых условиях способен значительно повышать её, иногда в разы. Причина такого поведения *Calc* остаётся неясной, но наблюдаемый эффект достаточно стабилен – ниже я приведу некоторые цифры. Второй способ (когда условием форматирования является истинность формулы), как и можно ожидать, даёт отрицательный эффект, особенно в том случае, если в формуле участвует ссылка, зависящая от значения форматируемой ячейки.

Теперь собственно о цифрах. Для теста я использовал вычисление стоимости тонны нефти в унциях золота, основываясь

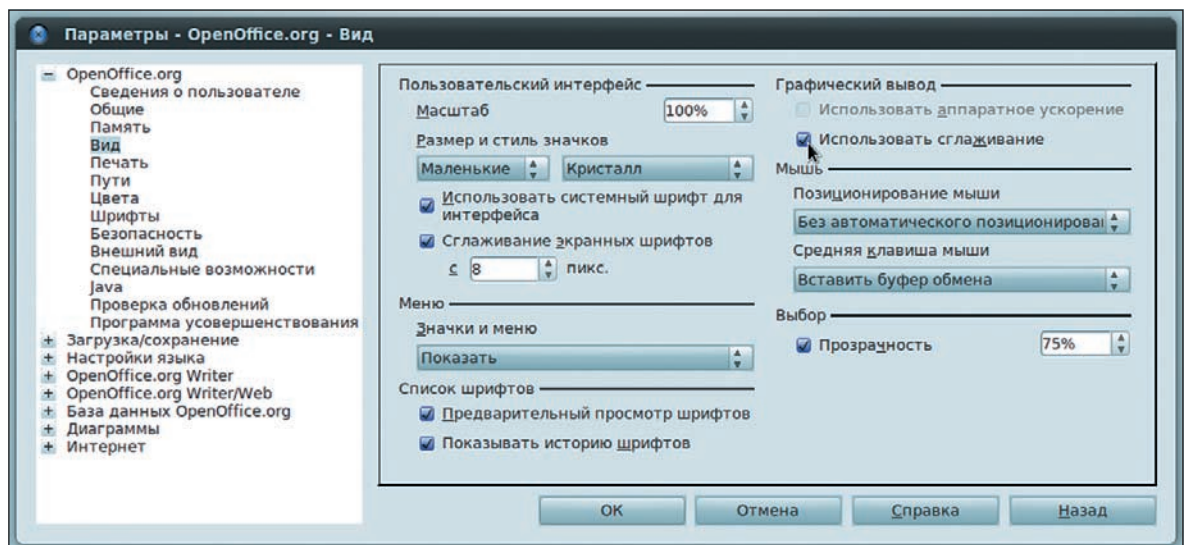


Рис. 4. Настройка отображения графических объектов.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

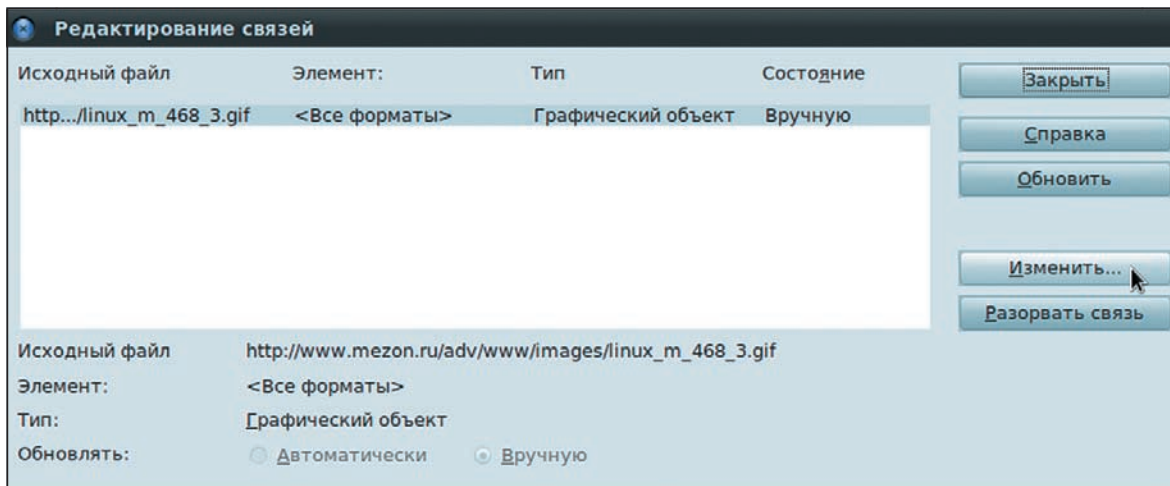


Рис. 5. Интеграция графического объекта в документ.

на цене унции золота и барреля нефти в долларах. Моя таблица исходных данных содержала почасовые котировки приблизительно за два года, а фактически 11 229 значений. Финалом вычисления при этом являлся расчёт средней цены нефти, выраженной в золоте, за указанный период.

Вычисление без форматирования занимало у меня в среднем около 700 миллисекунд. При условном форматировании ячейки, основанном на формуле вида `$'Лист'.#X>$'Лист'.J2`, где `$'Лист'.#X` – ссылка на форматлируемую ячейку (например, `$'Лист'.I2` для ячейки I2), а `$'Лист'.$J$2` – ссылка на результат, время расчёта возросло до 770 миллисекунд (на 10%), если лист с вычислениями и форматированием был неактивен, и до 880 миллисекунд (на 25%), если вычисления выполнялись на активном листе.

Однако куда интереснее результаты измерений для форматирования с условием, опирающимся непосредственно на значение ячейки. Здесь, при вычислении на неактивном листе, скорость падала совершенно незначительно – на 3% (720 миллисекунд). А вот на вычисление на активном листе Calc тратил всего 230 миллисекунд! То есть в данном случае прирост скорости был около 67% – более чем в 3 (!) раза.

Этот эффект пропадает, если столбец с условным форматированием скрыть. Эффект наблюдается практически всегда, если при вычислении на активном (видимом) листе содержится хотя бы одна ячейка с условным форматированием, основанном на сравнении значения ячейки с другой ячейкой на этом же листе, даже если стиль данной ячейки при этом не изменяется.

При всесторонней проверке эффекта, иногда удавалось добиться аж десятикратного ускорения. Это действительно неожиданный результат и одновременно отличная возможность понизить временные затраты практически даром.

Диаграммы и графические объекты

Форматирование, конечно же, не единственный способ сделать данные более наглядными – часто, при обработке больших массивов данных, для графического представления результатов используются диаграммы. И нередко, особенно на не очень производительных компьютерах, при отображении диаграмм возникают ощутимые и досадные задержки, преимущественно во время их прокрутки. Ещё одной причиной раздражения пользователя могут стать графические объекты, расположенные на листах Calc. Кстати, обратите внимание, что примечания к ячейкам листа также являются векторными изображениями, и если документ содержит большое количество примечаний с включенным непрерывным отображением (опция контекстного

меню Показать примечание), это тоже может вызывать серьёзные задержки.

Вначале разберёмся с диаграммами и векторными изображениями, созданными здесь же – в самом Calc. Как правило, длительные задержки при прокрутке этих объектов вызваны включённым сглаживанием векторных объектов. Сглаживание делает линии более плавными, убирая эффект «лесенки», но при этом оно достаточно ресурсоёмко.

Отключить сглаживание можно в настройках пакета: меню **Сервис > Параметры...**, затем в разделе **OpenOffice.org** открывшегося диалогового окна необходимо выбрать пункт **Вид** и снять галочку **Использовать сглаживание**, как показано на рис. 4.

Если ваш компьютер оснащён достаточным количеством оперативной памяти, вы также можете увеличить кэш графических объектов. Для этого необходимо выбрать пункт **Память** в том же разделе того же диалогового окна **Параметры...** Если не жалко, под эти нужды пакета можно отвести до 256 МБ. Кроме того, можно указать предел используемой памяти для кэширования одно-

го изображения и время, через которое память, занятая под кэш, будет освобождена.

А как влияет наличие диаграммы на время вычисления результата? На скорость самого вычисления диаграмма

не влияет практически никак, а вот на скорость отрисовки активного листа может повлиять достаточно ощутимо – при диаграмме для трёх рядов данных и 2000 значений задержка отрисовки листа у меня доходила до 3–4 секунд, и это при выключенном сглаживании; а если оно было включено, то задержка возрастала почти вдвое.

Причиной задержки при отображении вставленных графических объектов – как правило, растровых – может стать вставка объектов по ссылке, когда графический объект не интегрируется в файл документа, а продолжает оставаться на запоминающем устройстве удалённого источника. Здесь задержки возникают при обновлении объектов, при открытии и сохранении файла. Они могут стать особенно длительными, если графические объекты расположены в сети Интернет.

Чтобы решить эту проблему, необходимо интегрировать рисунки непосредственно в файл документа. Для этого в меню **Правка** выберите пункт **Связи...** Откроется диалоговое окно **Редактирование связей**, содержащее список изображений, вставленных с сохранением связи с источником. Выберите все пункты списка и нажмите кнопку **Разорвать связь** – рис. 5. Все изображения будут получены из источников и сохранены в документе. Обратите внимание, что это может занять некоторое время. **LXF**



Железо: Тест на

В самом ли деле ваши обновления и настройки повышают быстродействие системы? **Боб Мосс** расскажет, в чем измерить успех.



Наш эксперт

Боб Мосс

Успевает заниматься программированием и учиться на факультете компьютерных технологий.

Иной раз после замены аппаратного компонента или модернизации ПО мы не замечаем сколько-нибудь существенного улучшения и начинаем сомневаться, стоило ли городить огород.

Гадать совсем не обязательно. В мире свободного ПО есть тысячи инструментов и тестов для анализа быстродействия ЦПУ, воспроизведения фильмов или латентности подключения – с высокой точностью и в мельчайших подробностях.

Подробности как таковые проблему не решат, а вот сравнительный анализ [benchmarking] – может. Для тех, кто раньше не сталкивался с бенчмаркингом: это процесс измерения характеристик до модернизации и после неё, с последующим сравнением результатов. Такие инструменты можно использовать весьма широко, но для целей нашей статьи давайте ограничимся оценкой изменений.

Чтобы получить объективный результат, можно использовать идеи из врезки «Какой анализ лучше?» на соседней странице. Но мы коснёмся только тестирования аппаратуры и наблюдения за работой системы – а вы уже сами оцените полученные результаты и поймёте, где именно нововведения сработали.

Для примера возьмём тестирование жёсткого диска. Основные параметры здесь – эффективность файловой системы и аппаратные характеристики (например, время доступа): они опреде-

ляют скорость чтения и записи. Объём внутренних буферов тоже влияет на быстродействие, ведь при их правильной работе сокращается количество операций чтения/записи.

Сперва применим пресловутую программу *hdparm*. Мы кратко упоминали о ней в статье «Ускоряем Linux» номера **LXF124** (подписчики могут найти статью на сайте *Linux Format*); в большинстве дистрибутивов загрузить её можно через менеджер пакетов. Менее известно, что *hdparm* можно использовать для измерения скорости жёсткого диска, всего лишь введя в терминале строку

```
hdparm -Tt /dev/hda
```

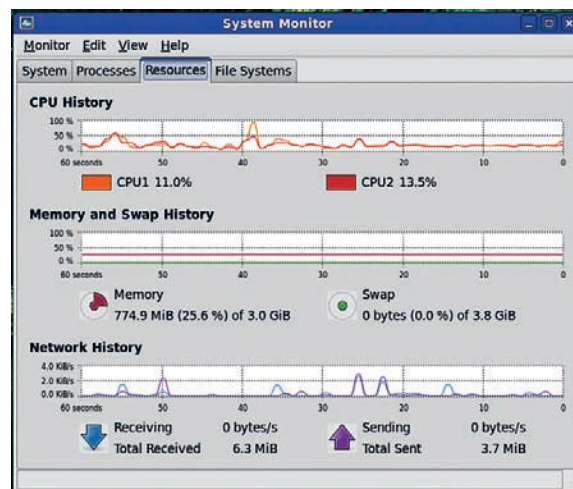
Не забудьте заменить */dev/hda* фактическим обозначением вашего диска. Вывод команды покажет, какой объём данных был прочитан за указанное время, а также переведет результат в мегабайты в секунду. Скорость кэшированного чтения обычно намного выше, чем нижняя цифра – последняя относится к тесту на прямое (буферизированное) чтение.

Давай подробности

Если хотите получить более точные сведения, найдите программу *Bonnie++* (о ней знают менеджеры ПО в большинстве дистрибутивов). Этот очень чуткий инструмент запускается единственной строкой в терминале:

```
bonnie++ /dev/hda
```

И снова: не забудьте заменить */dev/hda* названием своего диска. Этот тест выполняется дольше предыдущего, что вполне объяснимо: *Bonnie++* читает и записывает файл «без церемоний» (с прямым доступом к диску), а затем повторяет процесс с использованием кэша. Сведения, которые вы получаете в итоге, много



► Если консольный вывод недостаточно нагляден, поможет волнистая линия графика.

► **Не хотите пропустить номер?** Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

СКОРОСТЬ



Process Name	Status	% CPU	Nice	ID	Memory	Waiting
abrt-applet	Sleeping	0	0	2615	921.3 KiB	poll_sc
bash	Sleeping	0	0	3570	268.0 KiB	n_tty_r
bluetooth-applet	Sleeping	0	0	2305	1.4 MiB	poll_sc
bonobo-activation-server	Sleeping	0	0	2208	644.0 KiB	poll_sc
chrome	Sleeping	0	0	8832	4.0 MiB	poll_sc
chrome	Sleeping	0	0	7017	72.5 MiB	futex_x
chrome	Sleeping	0	0	5887	142.4 MiB	futex_x
chrome	Sleeping	2	0	4618	82.1 MiB	futex_x
chrome	Sleeping	0	0	4358	33.8 MiB	futex_x
chrome	Sleeping	0	0	4336	35.6 MiB	futex_x
chrome	Sleeping	0	0	4212	9.0 MiB	futex_x

» Как и в *Microsoft Task Manager*, можно наблюдать работающие процессы — и «убивать» лишние.

подробнее и, благодаря развёрнутой проверке, содержат значения латентности, времени записи, создания и удаления файлов в различных режимах.

Однако *Bonnie++* не ограничивается общим надзором за диском. Можно задать число повторения теста, объём используемых при этом файлов и их количество... Просто наберите *bonnie++* в терминале и полюбуйтесь на внушительный список опций, предлагаемых программой для оценки темпа работы вашего диска.

Проницательный Phoronix

Ну, а если проверки винчестера недостаточно, попробуйте *Phoronix Test Suite*: этот пакет есть почти в любом дистрибутиве. Маленькая нестыковка: графический интерфейс запускается из меню Gnome, но библиотека для его работы в современных дистрибутивах пока (на время написания статьи) не обновлена. Это неприятно — однако консольная версия *Phoronix Test Suite* сойдётся не хуже.

```
bob@localhost:~$ bonnie++ -d /home/bob
Writing a byte at a time...done
Writing intelligently...done
Rewriting...done
Reading a byte at a time...done
Reading intelligently...done
start 'em...done...done...done...done...done...
Create files in sequential order...done.
Stat files in sequential order...done.
Delete files in sequential order...done.
Create files in random order...done.
Stat files in random order...done.
Delete files in random order...done.
Version 1.96
-----Sequential Output----- --Sequential Input-- --Rand
Concurrency 1 --Per Chr-- --Block-- --Rewrite-- --Per Chr-- --Block-- --Seek
Machine Size K/sec %CP K/sec %CP K/sec %CP K/sec %CP K/sec %CP /sec
localhost.locald 66 42 39 18123 7 7307 5 693 82 15800 3 89.4
Latency 3506ms 7248ms 2828ms 193ms 1278ms 1970ms
Version 1.96
-----Sequential Create----- -----Random Create-----
localhost.localdoma -Create-- --Read-- --Delete-- --Create-- --Read-- --Delete
files /sec %CP /sec %CP /sec %CP /sec %CP /sec %CP /sec %CP
16 5828 61 +++++ + 6821 16 4806 50 +++++ + 25083 57 3506
248ms, 2828ms, 193ms, 1278ms, 1970ms, 46473us, 3091us, 228ms, 297ms, 926us, 1896us
Latency 46473us 3091us 228ms 297ms 926us 1896us
1.96, 1.96, localhost.localdomain, 1, 1276784667, 66, , 42, 39, 18123, 7, 7307, 5, 693, 82, 89.4, 3, 89.4, 4, 16, , , 5828, 61, +++++, +, 6821, 16, 4806, 50, +++++, +, 25083, 57, 3506, 248ms, 2828ms, 193ms, 1278ms, 1970ms, 46473us, 3091us, 228ms, 297ms, 926us, 1896us
[ bob@localhost ~ ]$
```

» *Bonnie++* протестирует файловую систему и выдаст развёрнутый анализ результатов.

Список всех имеющихся тестов выводится по команде

```
phoronix-test-suite list-tests
```

Если заменить *list-tests* на *list-suites*, выведется перечень опций для проведения тестов по заданным сценариям. Наш пример будет связан с тестом на скорость кодирования аудиофайлов, для чего понадобится такая команда:

```
phoronix-test-suite run audio-encoding
```

Для установки тестов при первом прогоне понадобится некоторое время, зато по результатам проверки вы узнаете скорость кодирования стандартного файла WAV в различных форматах-контейнерах.

И в завершение давайте выясним с помощью комплекта подробные сведения о системе:

```
phoronix-test-suite system-info
```

Мы лишь слегка коснулись темы измерения производительности систем и поиска путей совершенствования. Инструментов для этого очень много, дело за вами: пробуйте, экспериментируйте, доводите свою машину до идеального состояния. **LXF**

Скорая помощь

Анализ не должен длиться сутками! Если тест отнимает больше 30 минут — поищите другой, более соответствующий характеристикам компьютера.

Какой анализ лучше?

Пусть кто-то уверяет вас: «Процессор Intel Core 2 Solo компилирует *Firefox* из исходников на 10 % быстрее». Сразу возникают вопросы. Во-первых, неизвестно, с каким процессором проведено сравнение и какая версия *Firefox* бралась для проверки. Результаты должны обладать повторяемостью, и для их корректного сравнения надо учитывать мельчайшие детали исходного состояния и конечного результата.

Затем, некоторые вещи невозможно сопоставить объективно. Достоверным сравнение будет только в том случае, если модернизировать одну из идентичных систем (например, чтобы выяснить влияние доработки ядра на скорость загрузки). Сравнение времени загрузки Linux и Windows — занятие увлекательное, но целью реального анализа не является.

Наконец, следует пользоваться инструментами для своей платформы. Как отмечалось выше, результаты тестирования на Windows несравнимы с такими же результатами для Linux: ведь системы действуют по-разному. А коли инструменты разные, неправомерно сравнивать и результаты: возможно, дело не в выигрыше скорости, а в отличиях алгоритмов тестирования.

» Пропустили номер? Узнайте на с. 107, как получить его прямо сейчас.

Perl: ПОВЫСИМ

Часть 2: Большинство, наверное, возьмет для этих целей графический редактор, но **Михаил Смирнов** проделает все сам, с помощью Perl и изрядной доли математики.



тронного зума», применение которого может создавать только иллюзию улучшения качества изображения. Вместе с тем, такая числовая характеристика, как количество пикселей, дает лишь общее представление о качестве и реальном оптическом разрешении. Чтобы более точно разобраться в этом вопросе, нужно привлечь к рассмотрению функциональную характеристику – передаточную функцию ПЗС.

Немного теории

Передаточная характеристика устройства на ПЗС – сканера, цифрового фотоаппарата или видеокамеры – полностью описывается функцией передачи модуляции (ФПМ), которая характеризует падение контраста синусоидальных составляющих сигнала изображения в области пространственных частот. При оценке передаточной характеристики этих устройств одновременно будет учитываться и ФПМ оптического объектива, который фокусирует изображение в плоскости чипа ПЗС. В оптике пространственные частоты определяются как число линий на миллиметр, то есть величиной [1/мм]. В качестве примера на рис. 1 представлена типичная кривая ФПМ камеры на ПЗС.

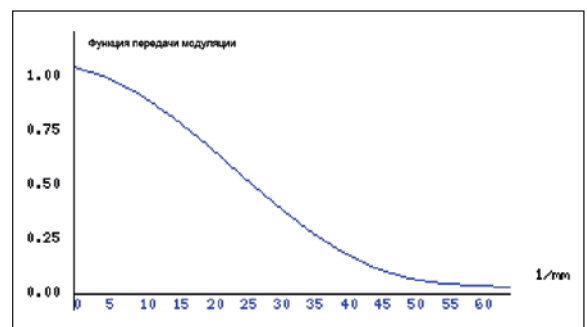
ФПМ характеризует связь между исходным объектом и его изображением. Рассмотрим в качестве объекта синусоидальное распределение контраста, показанное на рис. 2а. Частота синусоиды равна 20 1/мм, то есть на одном миллиметре укладывается 20 периодов синусоиды.

На рис. 2б и 2в представлены сечения синусоидального объекта. Размах синусоиды на рис. 2б характеризует контраст объекта на входе устройства. На выходе оптико-электронного устройства изображение синусоиды будет характеризоваться сечением, показанным на рис. 2в. Таким образом, видно, что ФПМ на частоте 20 1/мм показывает падение сигнала синусоиды с величины 1.0 на входе устройства до величины примерно 0.18 на его выходе.

Для получения значений ФПМ на всех пространственных частотах потребуется множество синусоидальных объектов

Наиболее важной характеристикой цифровой фото- и видеотехники является разрешающая способность преобразователей света в электрический сигнал. Чем она лучше, тем более мелкие детали мы можем наблюдать в изображении. В современных бытовых цифровых фото- и видеоустройствах такими преобразователями являются полупроводниковые приборы с зарядовой связью (ПЗС), которые не так уж давно пришли на смену фотопленке и вакуумным преобразователям света. Основное преимущество ПЗС перед фотопленкой состоит в возможности регистрировать значительно больший диапазон внешней освещенности или, иначе говоря, получать значительно большее количество градаций полутонов в изображении. Преимущество перед вакуумной техникой, прежде всего, характеризуется практически стопроцентным исключением геометрических искажений. Слабым местом первых ПЗС-разработок являлось низкое оптическое разрешение, что было связано с крупным размером фоточувствительных ячеек ПЗС. Чем меньше геометрический размер ячеек и чем больше количество таких ячеек в чипе ПЗС, тем лучшее оптическое разрешение можно получить. Количество ячеек по горизонтали и вертикали определяют размер цифрового изображения в пикселях по ширине и высоте.

В первых разработках, особенно фототехники, применялась процедура искусственного увеличения количества пикселей с помощью интерполяции. Эта процедура получила название «элек-



» Рис. 1. Функция передачи модуляции камеры на ПЗС.



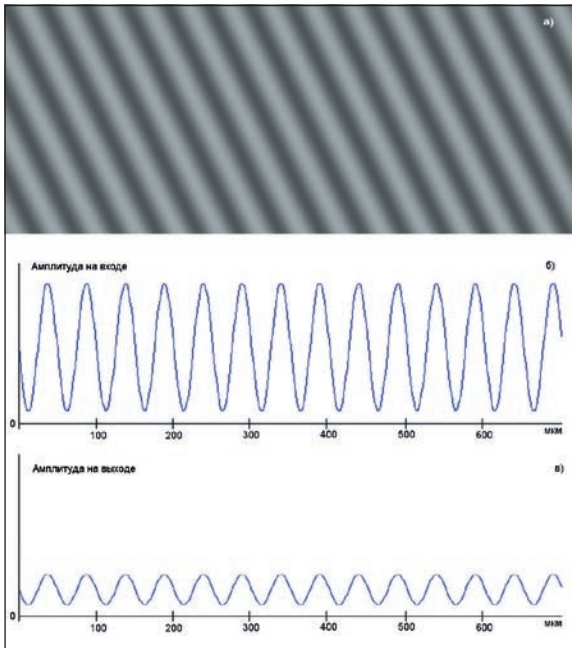
Наш эксперт

Михаил Смирнов

Независимый web- и прикладной программист в области цифровой обработки изображений, голографии и распознавания образов.

» **Месяц назад** Возможности библиотеки *Gd* и создание изображений из Perl.

качество фото

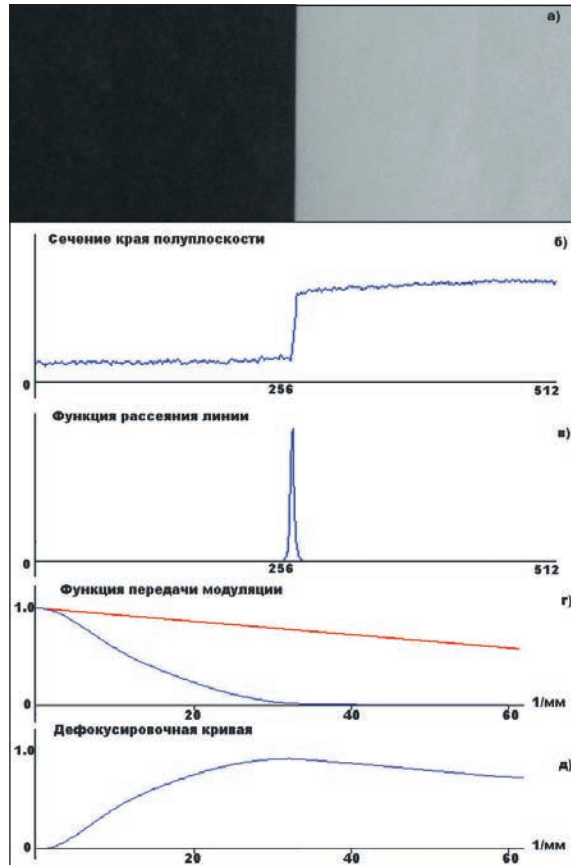


► Рис. 2. Пример синтеза изображения.

с различными частотами. Однако в этом нет необходимости, если воспользоваться объектом в виде протяженного резкого скачка контраста, получившего название «края полуплоскости». Изображение края полуплоскости показано на рис. 3а. Чтобы получить искомую ФПМ, необходимо выполнить дифференцирование края полуплоскости (по нормали к краю), а затем сделать преобразование Фурье.

Последовательность и результаты преобразований иллюстрируются на рис. 3. На рис. 3б показано одно из сечений края полуплоскости. Чтобы снизить влияние шумов ПЗС, перед дифференцированием предварительно выполняется усреднение сечений края полуплоскости. Суммирование m сечений края полуплоскости обеспечивает повышение отношения сигнал-шум в корень из m раз. Результат дифференцирования усредненного сечения края полуплоскости показан на рис. 3в – полученное нами распределение называется функцией рассеяния линии. Для определения ФПМ остается выполнить преобразование Фурье функции рассеяния и затем вычислить модуль от полученных комплексных коэффициентов Фурье (рис. 3г). Чтобы оценить ошибку, которую вносит цифровая опико-электронная система, требуется выполнить вычитание полученной нами ФПМ $H(v)$ из так называемой дифракционной ФПМ $\hat{H}(v)$, которая имеет максимально достижимую частоту v_m :

$$U(v) = \hat{H}(v) - H(v)$$



► Рис. 3. Край полуплоскости: а) тестовое изображение края полуплоскости, б) сечение края полуплоскости, в) функция рассеяния линии, г) ФПМ устройства на ПЗС, д) дефокусировочная кривая.

v – пространственная частота. Предельная частота дифракционной ФПМ определяется простой формулой $v_m = F/\lambda D$, где F – фокусное расстояние объектива; D – диаметр объектива; λ – длина волны света, обычно принимается равной 660 нанометрам (оранжевый цвет).

Дифракционная ФПМ (красная линия на рис. 3г) представляет собой наклонную прямую, равную 1 на нулевой частоте $v = 0$ и равную 0 на дифракционной частоте v_m . Пример разностной ФПМ $U(v)$ показан на рис. 3д. Кривая на рис. 3д характеризует величину ошибки передачи контраста, которая обусловлена реальной опико-электронной системой на ПЗС. Во многих случаях подобная ошибка связана с расфокусировкой объектива (потерей резкости). И действительно, распределение на рис. 3д напоминает дефокусировочную кривую.

Различные варианты настройки фокуса объектива будут, в той или иной степени, приводить к потере резкости. Чем сильнее расфокусирован объектив, тем больше будут значения разностной функции $U(v)$ и тем хуже оптическое разрешение. При этом по пику кривой

» Пропустили номер? Узнайте на с. 107, как получить его прямо сейчас.

можно оценить пространственную частоту ν , на которой ФПМ имеет наибольший градиент падения контраста. Если не принимать в расчет шумы камеры на ПЗС, то наибольшее разрешение оптико-электронной системы определяется соотношением $\nu_L = 1/2L$, где L – линейный размер ячейки ПЗС. Для реальных систем уровень шума на высоких частотах будет превышать сигнал изображения, и реальное предельное разрешение ν_r будет значительно ниже ν_L .

Пример кода для получения искомого ФПМ показан ниже. Исходная матрица изображения $B(x,y)$ края полуплоскости содержится в двумерном массиве @B. Начальная и конечная строки суммирования задаются параметрами \$row1 и \$row2, соответственно.

```
for($j=$row1;$j<=$row2;$j++){
for($i=0;$i<$N;$i++){
    $g[$i] += $B[$i][$j];
}
$Nrr=$row2-$row1-1;
@g=map($_/$Nrr,@g);
for($i=1;$i<$N;$i++){
    $PSF[$i] = $g[$i] - $g[$i-1];
}
use Math::FFT;
for($i=0;$i<$N;$i++){
    $data->[2*$i]=$PSF[$i];
    $data->[2*$i+1]=0;
}
$fft = new Math::FFT($data);
$coeff = $fft->cdft();
for($i=0;$i<$N;$i++){
    $H[$i] = sqrt ( $coeff->[2*$i]**2 + $coeff->[2*$i+1]**2 );
}
@H=map($_/$H[0],@H);
```

Результат дифференцирования заносится в массив @PSF, который представляет собой функцию рассеяния линии. Для выполнения преобразования Фурье воспользуемся математической библиотекой Perl Math::FFT, в которой реализован алгоритм быстрого преобразования. Одномерное преобразование Фурье в этой библиотеке выполняется с помощью подпрограммы cdft(). Результат преобразования Фурье заносится в массив коэффициентов \$coeff, где коэффициенты с четным индексом соответствуют реальной части, а коэффициенты с нечетным индексом – мнимой части комплексного числа, соответственно. Результирующая ФПМ заносится в массив

@H, значения которого нормируются к первому значению массива \$H[0], соответствующему постоянной составляющей изображения. Таким образом, \$H[0]=1, а все остальные значения ФПМ меньше единицы.

На рис. 3г, вместе с вычисленной нами ФПМ, красной линией показана дифракционная оптическая функция. Таким образом, чем ближе ФПМ нашего устройства к дифракционной функции, тем с более высоким качеством мы сможем получать наши фотоизображения.

Улучшение качества

Вывод, полученный нами в предыдущем разделе, позволяет сформулировать подход для улучшения качества фотоизображений, а именно: компенсировать падение ФПМ на тех частотах, где оно выражается наиболее сильно, и приблизить тем самым ФПМ к дифракционному пределу. В цифровой обработке изображений такие восстанавливающие фильтры получили название обратных или инверсных. Математически это выражается достаточно просто, а именно: спектр исходного фотоизображения $S(\nu)$ необходимо умножить на функцию, обратную ФПМ устройства на ПЗС, которую мы ранее восстановили из изображения края полуплоскости. В предположении, что мощность шума пренебрежимо мала, спектр восстановленного изображения равен отношению $S_1(\nu) = S(\nu)/H(\nu)$, то есть функция идеального фильтра $Q(\nu) = 1/H(\nu)$. Принимая во внимание шумы, которые вносятся любой реальной системой, необходимо найти решение для $Q(\nu)$ на тех частотах $\nu > \nu_r$, где шум превышает сигнал. Предельная частота ν_r оценивается как частота, на которой дисперсия (энергия) шума начинает превышать значения ФПМ. Один из вариантов оптимального фильтра имеет вид

$$Q(\nu) = [1 + H(\nu_r)^2 \cdot H(\nu) / (H(\nu_r)^2 + H(\nu)^2)] \quad (1)$$

На рис. 4 показан результат расчета фильтра по формуле (1).

Инверсные оптимальные фильтры, использующие восстановленные ФПМ, обеспечивают повышение контрастности на тех пространственных частотах, на которых это падение обусловлено реальной оптико-электронной системой. Это основное отличие

восстанавливающих фильтров от «слепых» фильтров высоких частот или операторов типа Собеля для подчеркивания границ, применяемых в пространственных программных пакетах.

«Инверсные оптимальные фильтры повышают контрастность.»

Алгоритм улучшения качества фотоизображения с помощью инверсной фильтрации включает следующие этапы. На первом этапе осуществляется преобразование Фурье исходного фотоизображения $B(x,y)$ и получение спектра $S(\nu_x, \nu_y)$. На втором этапе формируется инверсный фильтр $Q(\nu_x, \nu_y)$, значения которого умножаются на спектр $S(\nu_x, \nu_y)$. На третьем этапе выполняется обратное преобразование Фурье произведения $S_1(\nu_x, \nu_y) = S(\nu_x, \nu_y) \cdot Q(\nu_x, \nu_y)$, результатом которого будет являться восстановленное изображение $B_1(x,y)$.

Первый и третий этапы реализуются с помощью стандартных подпрограмм одномерного преобразования Фурье модуля Perl Math::FFT. Двумерное преобразование Фурье получается с помощью последовательного применения одномерного преобразования, например, к функции изображения $B(x,y)$ сначала по строкам, а затем по столбцам матрицы. Отметим, что преобразованию подвергаются RGB-компоненты цветного изображения по отдельности. В наших примерах используется компонента Blue исходного RGB-изображения. Ключевым этапом преобразования является второй этап, обеспечивающий развертывание одномерной функ-

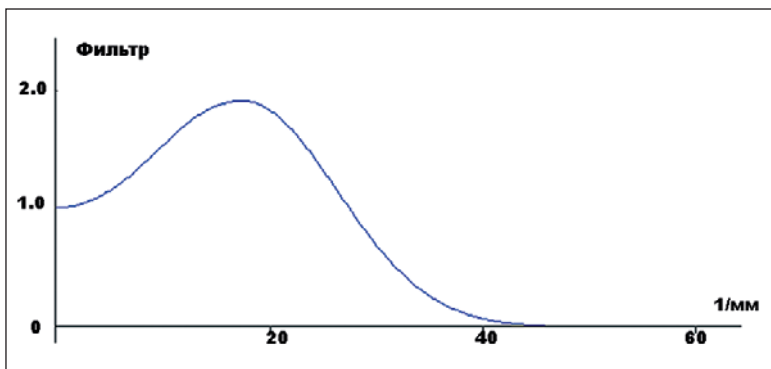


Рис. 4. Оптимальный линейный фильтр для улучшения фотоизображений.

» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe/!

ции фильтра в двумерный массив и умножение значений фильтра на спектр исходного изображения. Фрагмент программной реализации такого алгоритма фильтрации представлен ниже:

```

$N2=$N/2;
for($i=0;$i<$N2;$i++){
    $FILTER[$i] = (1.+ $MTF[$nf]**2)*$MTF[$i]/($MTF[$nf]**2 +
    $MTF[$i]**2);
}
for($j=0;$j<$N;$j++){
for($i=0;$i<$N;$i++){
    $w2[$i][$j]=0;
}}
&d2cdft(\@B,\@w2,\$N,\$PI);
for($k=0;$k<$N;$k++){
    $k1=$k;
    if($k > $N2) { $k1=$k-$N;}
    $x=$k1**2;
for($j=0;$j<$N;$j++){
    $j1=$j;
    if($j > $N2) { $j1=$j-$N;}
    $R = sqrt ( $x + $j1**2 );
    $Q = &parv(\$R,\@Xc,\@FILTER,\$N2);
    $Re[$k][$j] *= $Q;
    $Im[$k][$j] *= $Q;
}}
&d2cdfti(\@Re,\@Im,\$N,\$PI);
    
```

Входное фотоизображение содержится в двумерном массиве @B. Прямое и обратное двумерное преобразование Фурье выполняются с помощью подпрограмм d2cdft() и d2cdfti(), соответственно. Исходная функция ФПМ, полученная нами выше, находится в массиве @MTF. Функция фильтра, рассчитанная по формуле (1), заносится в одномерный массив @FILTER. Для получения двумерной функции фильтра Q(v_x,v_y) используется подпрограмма линейной интерполяции parv(), которая обеспечивает вычисление промежуточных значений фильтра в плоскости пространственных частот (v_x,v_y). Параметр R является текущим радиусом, а вспомогательный массив @Xc представляет собой массив аргумента входной функции фильтра, и, в частности, может быть задан как безразмерная функция с единичным шагом:

```

for($j=0;$j<$N2;$j++){ $Xc[$j]=$j;}
    
```

Выходные значения восстановленного спектра S₁ фотоизображения заносятся в массивы реальной @Re и мнимой @Im части комплексного спектра S₁, соответственно. Результирующее восстановленное изображение, после выполнения обратного преобразования Фурье с помощью подпрограммы d2cdfti(), помещается в двумерный массив @Re. На рис. 5 показано исходное и восстановленное фотоизображения, полученные с помощью цифровой фотокамеры в стандартном режиме съемки. Как правило, в этом режиме съемки цифровые фотографии получаются вполне удовлетворительного качества.

Чаще всего пониженное качество цифрового фото связано с режимом макросъемки. На рис. 6 показано исходное и восстановленное фотоизображения, которые были получены в этом режиме.

С точки зрения нагрузки на процессор, при выполнении процедур улучшения качества наиболее затратной является процедура интерполяции. Для уменьшения времени обработки фотоизображения можно заранее сформировать двумерную матрицу фильтра Q(v_x,v_y) и хранить его на диске. Для перемножения матриц спектра исходного фотоизображения S(v_x,v_y) и матрицы фильтра Q(v_x,v_y) можно воспользоваться пакетом (модулем) Perl PDL. Основное достоинство пакета PDL – высокая скорость сло-



► Рис. 5. Результат компенсации ФПМ фотокамеры для стандартного режима съемки: а) исходное фотоизображение (Blue), б) результат восстановления.

жения и перемножения матриц большой размерности и компактное хранение массивов данных с плавающей запятой. Например, скалярное произведение двух матриц вещественных чисел с плавающей запятой и размерностью 2048×2048 составляет около 1 секунды, а сложение таких матриц займет доли секунды. Двумерный массив чисел с плавающей запятой размерностью 1024×1024, принимая тип данных PDL, будет занимать всего 4 МБ оперативной памяти. Пакет PDL позволяет лаконично записывать математические операторы для работы с матрицами в виде одной строки – например, код для скалярного умножения двух матриц \$a и \$b будет иметь вид

```

use PDL;
$c = $a * $b;
    
```

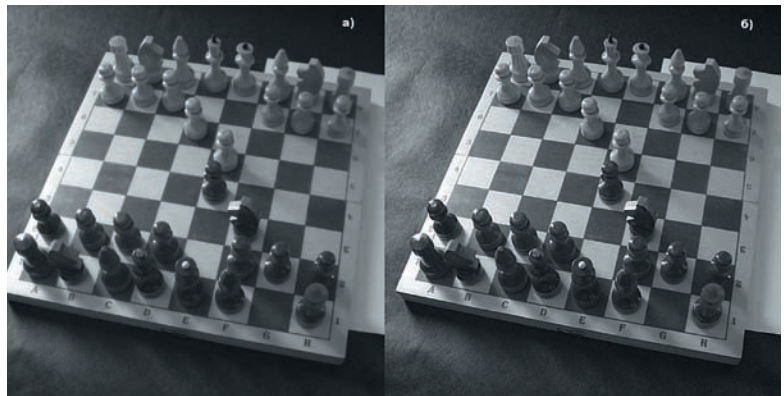
Соответственно, код для перемножения двумерного комплексного спектра изображения на двумерный фильтр запишется в виде

```

$re = pdl[@Re];
$im = pdl[@Im];
$q = pdl[@Q];
$re = $q*$re;
$im = $q*$im;
    
```

Уважаемый читатель, если вас поначалу озадачили «преобразование Фурье» и реализация математических формул, то отнеситесь к ним как к «черному ящику». Главная польза применения Math::FFT для оценки качества состоит в том, что с помощью небольшой программы на Perl у вас появилась возможность объективно сравнивать оптическое разрешение современных фото- и видеосистем во всей полосе видеочастот, а не ориентироваться только на один параметр разрешения – количество мегапикселей. **LXF**

► Рис. 6. Результат компенсации ФПМ фотокамеры для режима съемки макро: а) исходное фотоизображение (Blue), б) результат восстановления.



» Через месяц Стеганография и все-все-все: скрываем данные в изображениях.

POV-Ray: АЛХИМИЯ

Когда мы пишем эти строки, в большей части России стоит жара. Поэтому Вячеслав Ястребцев решил залить сцену водой... приятной, прохладной водой!



Наш эксперт

Вячеслав Ястребцев
Мечтает узнать, что мешает заниматься графикой в Linux, и исправить это.

В прошлый раз мы изготовили две шестерёнки. Эта парочка смотрелась сиротливо, и сегодня мы создадим таинственную лабораторию, используя POV-Ray на все сто!

Голые стены

Как обычно, разделим сцену на несколько файлов. Скопируйте в папку с проектом файлы **tower_shapes.inc** и **tower_textures.inc**. В первом описаны необходимые формы, во втором – базовые текстуры; без них работать будет нельзя. Откроем текстовый редактор и создадим главный POV-файл сцены (**tower.pov** на DVD). Подключим эти файлы директивой **#include**:

```
#include "tower_shapes.inc"
#include "tower_textures.inc"
```

Далее, размещаем камеру и создаём светильник:

```
camera {
    location <0, 5, -40>
    look_at 0
    angle 37
}
light_source {
    <0, 200, 0>,
    color rgb <.9, .8, 1>
    parallel
    point_at <70,0,100>
}
```

Здесь мы применили новые параметры: **parallel** и **point_at**. Один имитирует большой и отдалённый источник света, вроде Солнца или Луны (в сцене – Луна), испускающий пучок параллельных лучей. Второй указывает точку, на которую падает луч из центра источника, задавая таким образом направление света (источник можно вращать, как и любой другой объект, с помощью модификатора **rotate**, меняя направление освещения; иногда так удобнее).

Большая часть лаборатории будет затоплена водой: создадим водную поверхность, используя объект **plane** (плоскость):

```
plane {
    y, -.5
    material {water}
}
```

Плоскость задана вектором нормали и числом, задающим сдвиг плоскости вдоль нормали от начала координат. Чтобы плоскость стала видимой, ей назначен материал **water** из файла **tower_textures.inc**. Теперь определим факел:

```
#declare brand = union {
    #declare len = 3.5;
    cylinder {
        0, len*y, 0.12
        texture { brand_metal }
    }
    cone {
        len*y, 0.12,
        (len+.7)*y, .3
        texture { brand_metal }
    }
}
```

```
light_source {
    (len+1.1)*y,
    color rgb <1, .5, .2>*2
}
```

Ещё один новый объект – **cone** (конус) – задан двумя парами обязательных параметров: вектором, определяющим положение центра торца, и радиусом. Если в одной из пар радиус равен 0 – получаем обычный конус, если оба радиуса ненулевые – усечённый. Конус и цилиндр создают корпус светильника; добавив в объединение источник света, мы заставляем факел светить. Переменная **len** определяет длину ручки факела. Заполним сцену объектами:

```
union {
    object { transmitting
        translate <1.5,4.5,0>
    }
    object { stock
        texture { stock_wood }
        translate -z
    }
    object { stock
        texture { stock_wood }
        translate z
    }
    object { runway
        texture { floor_wood }
        translate -2*x
    }
    object { brand
        translate <0,0, -3>
    }
    rotate 45*y
    translate 10*z
}
```

Пристроим на сцену группу объектов, определённых в файле **tower_shapes.inc**: **transmitting** – шестерни, созданные на прошлом уроке; **stock** – опора, на которой они установлены; **brand** – только что созданный факел; **runway** – помост, на котором все предметы стоят в центре водоёма. Сразу назначим предопределённые текстуры.

Поместим в сцену стены:

```
object { walls
    texture { stonework }
}
```

Объекты размещены; отрисовав сцену, вы увидите, что вышло.

Да будет цвет!

Все объекты на рендере, кроме шестерён (для них текстура была заранее определена), имеют монотонный серый окрас. Это копии простой серой текстуры. Давайте сделаем ее более красочной!

Начнём со стены. Откройте файл **tower_textures.inc** и найдите строчку **#declare stonework = texture { def }**, назначающую текстуру кладки переменной **stonework**. Кладка состоит из двух матери-

Текстур



» Скучные однотонные поверхности? Сейчас добавим текстур.

алов – камня и раствора. Для их имитации понадобятся две дополнительные текстуры: камень темнее и более блестящий, чем раствор. Удалим слово **def** из определения текстуры и создадим текстуру камня, вписав внутри текстуры **stonework** следующее:

```
pigment { color rgb <.2,.21,.2>
  finish {
    specular .1
    roughness .003
  }
  normal {
    crackle
    bump_size -1
  }
}
```

Мы используем новый модификатор – **normal**, для имитации выступающих поверхностей камней. Он ведёт себя как обычная карта нормалей, создавая иллюзию неровной поверхности (можете создать реально выступающие булыжники, используя изоповерхности, изученные на первом уроке). **Crackle** – тип паттерна, используемого для имитации узора; он выглядит аналогично текстуре **vogon** в *Blender*, создавая мозаику, похожую на каменную кладку. **bump_size** задаёт высоту выступов; отрицательные значения этого параметра меняют местами выступы и впадины, заставляя камни выглядеть выступающими внутрь помещения.

Для имитации раствора создадим простую серую текстуру:

```
#declare sw_mortar = texture {
  pigment { color rgb <.53,.53,.53>
}
```

Теперь совместим две текстуры в одном материале, применив специальный модификатор – карту текстур (**texture_map**). Она используется аналогично знакомому модификатору **color_map**, чередующему цвета в зависимости от значений паттерна. Главное отличие карты текстур – возможность чередовать текстуры и прочие свойства поверхности (интенсивность бликов и диффузного отражения, нормали). Создадим карту текстур, используя паттерн **crackle**:

```
crackle
scale 4
texture_map {
  [.06 sw_mortar]
  [.06 stone]
}
```

Между паттерном и картой текстур стоит модификатор **scale**, увеличивающий узор паттерна. Создание текстуры стены закончено.

Наломаем дров

Попробуем изобразить деревянный помост и опоры шестерён. Найдите в файле **tower_textures.inc** строчку **#declare stock_wood = texture(def)**, удалите **def** и начните создавать пигмент:

```
pigment{
  wood
  rotate 3*y
```

В основе пигмента лежит паттерн **wood** (дерево) – он создаёт узор типа годовых колец древесного ствола, для имитации каковых и предназначен. Правда, эти кольца идеально ровные, без всяких сучков и извивов, и для реалистичности придётся внести в узор хаос. Воображаемое бревно бесконечно большое, его сердцевина совпадает с осью Z; поворот на 3° вокруг оси Y чуть нарушит симметрию, добавив натурализма. Для более основательного перемешивания применим знакомый модификатор **warp**:

```
warp {
  turbulence <1,1,0>
  octaves 3
  lambda 1.5
  omega .3
}
```

Единственное новое для нас – вектор **<1,1,0>** в качестве параметра **turbulence**. Как вы, надеюсь, помните с предыдущего урока, **turbulence** задаёт дистанцию начального смещения текстуры, а вектор регулирует величины смещений по разным осям. Текущие значения разрешают пертурбации только в плоскости XY, т.е. «поперёк» воображаемого бревна, что создаст множество узелков. Сожмем «бревно» с боков и растянем вдоль оси Z, уменьшив ширину годовых колец и превратив узелки в веретенообразные волны натурального узора древесины модификатором **scale**:

```
scale <.1, .1, 3>
```

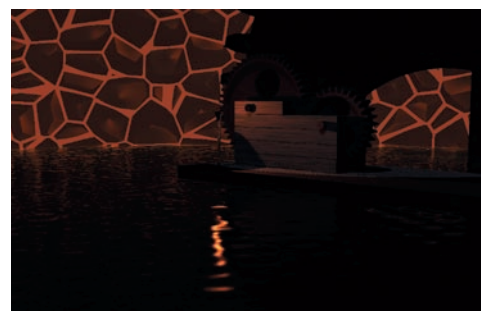
Теперь наша заготовка похожа на настоящее дерево. Одна загвоздка: деталь с такой текстурой будет выглядеть вырезанной из огромного цельного ствола. Такая конструкция в жизни встречается редко – обычно деревянные сооружения набирают из отдельных брусьев. При желании можно набрать цельные фигуры из кирпичиков-брусьев, слегка модифицировав текстуру для каждого бруса – вам помогут макросы из прошлого урока. Но POV-Ray позволяет создать нужный эффект с одной цельной текстурой.

Реальные брусья – это выпиленные из бревна куски древесины, а мы уже соорудили ствол бесконечного виртуального дерева; осталось нарезать его на фрагменты. Снова применим **warp**:

```
warp {
  repeat y
  offset 10*z
  flip <1,1,0>
}
```

Модификатор берет новые параметры и не вносит случайный шум в текстуру, а повторяет её через определённые интервалы с небольшими изменениями. Вектор, принимаемый параметром **repeat**, задаёт направление, в котором будет повторяться текстура, и размер повторяемой области.

» Текстуры полностью изменили изображение!





► Теперь свет как в катакомбах, а не операционной.

Длина его произвольна, но он обязательно должен совпадать с одной из осей! **Offset** – смещение повторяемых фрагментов от предыдущих. Повтор идентичных участков не реалистичнее цельного бревна – сдвигая текстуру при каждом повторе, мы добьёмся большего натурализма.

Направление сдвига может быть любым. **Flip** зеркально отражает каждый фрагмент относительно предыдущего по осям с ненулевым значением передаваемого вектора. Повернём модифицированный паттерн на 90° вокруг оси Y и применим карту цветов:

```
rotate 90*y
color_map {
    [.4 color rgb <.9,1,.8>*.25]
    [.55 color rgb <1,.9,.8>*.35]
}
```

В завершение, сымитируем блики:

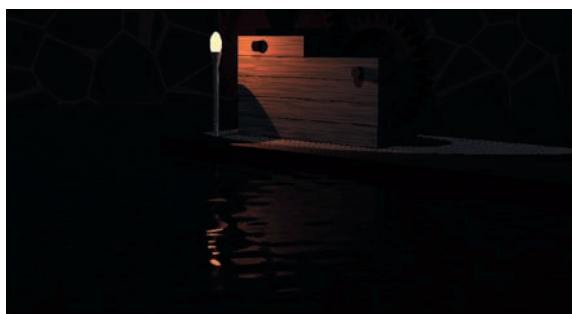
```
finish {
    specular .1
    roughness .1
}
```

На стыках деталей имеются щели; издали они похожи на тёмные полосы. Смоделируем их на нашей текстуре, добавив после фигурной скобки, закрывающей описание текстуры, ещё одну текстуру:

```
texture {
    pigment {
        gradient .5*y
        color_map {
            [0 color rgbt 0]
            [.051 color rgbt <0,0,0,1>]
        }
    }
}
```

Расположение текстур друг за другом создаст многослойную [layered] текстуру – это способ чередования свойств поверхности материала, альтернативный карте текстур. Текстура, определённая первой, будет фоном; все последующие текстуры должны иметь прозрачные участки, сквозь которые видны предыдущие. Многослойную текстуру можно представить как стопку «прозрачек». Пигмент текстуры, определённой поверх древесины, использует паттерн **gradient** – плавный переход значений от 0 до 1 вдоль указанного вектора. На дистанции, равной вектору, будет единица, затем идет плавный спад до нуля на дистанции, равной удвоенному вектору, потом снова начинается спад – и так до бесконечности.

Карта цветов создаёт узкую тёмную полосу вблизи нулевого значения. Цвет задан четырёхмерным вектором; четвёртый компонент – величина пропускания (**transmit**), или прозрачность участка. Итак, большая часть второй текстуры абсолютно прозрачна.



► Вот он, наш волшебный фонарь.

Займёмся текстурой помоста. Просто скопируйте описание текстуры опор шестерён и вставьте вместо **texture(def)** в строке **#declare floor_wood = texture(def)**. Затем нам потребуется внести кое-какие изменения: на помосте, лежащем горизонтально, текстура должна повторяться в другом направлении, и доски, из которых набран пол, будут чуть шире. Для этого во втором модификаторе **warp** первой текстуры заменим значение параметра **repeat** на **1.5*x**, **offset** – на **20*z**, а во второй текстуре – **gradient** на **.75*z**.

Блики

Осталась отделка водной глади и ручки факела. Для факела –

```
#declare brand_metal = texture{
    pigment{ color rgb <.96,.98,1> }
    finish {
        diffuse .01
        ambient 0
        specular .9
        roughness .01
        metallic
        reflection { .9 metallic }
    }
}
```

Параметр **reflection** [отражение] добавлен в секцию **finish** для имитации зеркальной металлической поверхности, отражающей окружающие предметы. Число внутри описания этого параметра определяет долю света, отражаемого поверхностью; ключевое слово **metallic** окрашивает отражение в цвет металла.

Текстура воды должна выглядеть следующим образом:

```
#declare water = material{
    texture{
        pigment { color rgb .1 }
        finish {
            diffuse .01
            specular .85
            roughness .0005
            reflection { .9 fresnel }
        }
    }
    normal {
        ripples
        frequency 30
        bump_size .2
        translate -.5*<1,1,0>
        scale 40
    }
    interior { ior 1.333 }
}
```

Это даже не текстура, а материал [material] – модификатор, объединяющий в блок две различные для POV-Ray сущности объектов – текстуру [texture] и интерьер [interior]. Текстуры задают свойства поверхности – бесконечно тонкой оболочки объектов, интерьеры – оптические свойства внутри объёма. В секции **finish** при описании параметров отражения имеется ключевое слово **fresnel**, означающее реалистичный расчёт отражения в зависимости от угла, под которым видна поверхность. Эта опция работает, только если известен коэффициент преломления материала, задаваемый при помощи параметра **ior** внутри секции **interior**. В секции **normal** использован паттерн **ripples**, имитирующий рябь на воде; параметр **frequency** задаёт число волн на единичной дистанции. По умолчанию, рябь – это результат интерференции волн от десяти источников в кубе со стороны в одну единицу; модификатор смещён и масштабирован, чтобы эффект занял всю доступную площадь.

С текстурами покончено, ура! Запустите рендер, чтобы увидеть, насколько радикальны изменения. Если что-то не сработает, обратитесь к файлу **tower_textures_final.inc** за готовыми текстурами.

Тёплый свет огня

Яркий свет факела режет глаз – освещён огромный объём без малейшего признака затухания, а свет обязан затухать пропорционально квадрату расстояния от источника. В файле **tower.pov** настроим светильник в факеле, добавив в его описание строки

```
fade_power 2
fade_distance 11
```

Первый параметр задаёт степень убывания интенсивности освещения с расстоянием, второй – дистанцию, на которой интенсивность света будет соответствовать определённой при создании источника. После рендеринга вы увидите в помещении таинственный полумрак.

Положение источника света весьма заметно, но сам источник невидим, что выглядит крайне оригинально. Создадим магический светящийся шар! Добавьте в описание источника света следующую конструкцию:

```
looks_like {
  sphere {
    0, .3
    scale 2*y
    texture {
      finish {
        diffuse 0
        ambient <1, .5, .2>*4
      }
      pigment {rgb 1}
    }
  }
}
```

Опция **looks_like** [англ. «выглядит как»] позволяет задать объект, изображающий источник света; у нас это вытянутая сфера. Чтобы сделать сферу видимой, ей присвоена простая текстура. Параметры **diffuse** и **ambient** в секции **finish** задают степени отражения направленного и рассеянного освещения соответственно. Направленное освещение создаётся источниками света (**light_source**) и приводит к появлению на объектах переходов светотени. Рассеянное освещение имитирует свет, отражённый от поверхностей объектов. По умолчанию берется самая простая модель рассеянного освещения, представляющая рассеянный свет однородным по всей сцене. Это не слишком корректно с точки зрения фотореализма, зато позволяет создавать интересные эффекты – в нашем случае, мы симитировали светящуюся поверхность, назначив очень большую величину отражения рассеянного света. Параметр **diffuse** сделан нулевым, чтобы избежать появления бликов.

Наши тени имеют очень резкие, контрастные края. Смягчим их, применив площадные источники света. Вначале займёмся светом Луны – добавим в описание первого источника света строки

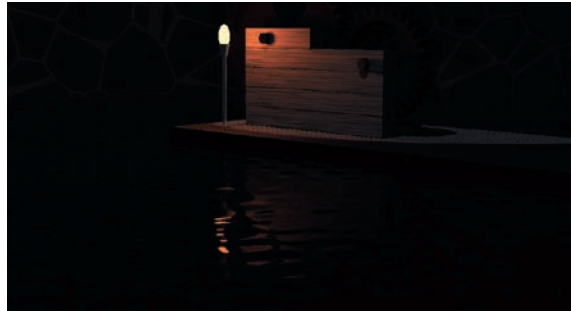
```
area_light
2*x, 2*y, 3, 3
jitter
```

Area_light включает имитацию объёмного источника. На следующей строке первые две величины задают размер излучающей свет плоскости по осям X и Y соответственно. Площадной источник имитируется как массив точечных источников, расположенных на плоскости; следующая пара значений задаёт число строк и колонок в этом массиве. Ключевое слово **jitter** дополнительно размывает контуры тени, смягчая её.

Добавим в описание источника света на факеле следующее:

```
area_light
.7*x, .7*y, 4, 4
jitter
circular
orient
```

Circular деформирует источник света, делая плоскость не прямоугольной, а круглой; **orient** поворачивает источник на точку,



» Пришла пора смягчить края теней.

для которой вычисляется освещённость; а вместе эти опции имитируют шарообразный источник света.

Теперь рендеринг будет долгим, и на финальном изображении края теней станут размытыми.

Антиалиасинг

Работа над сценой закончена – осталось только настроить сам рендер. Наверно, вы уже отметили ступенчатость краёв объектов и то, что текстура помоста выглядит некрасивой мешаниной точек. Для борьбы с этими артефактами применяется антиалиасинг.

Скопируйте с диска в папку с вашим проектом файл **tower.ini**, если вы до сих пор обходились без него. Первым делом убедитесь, что в этом файле в строчке **Input_File_Name="имя файла"** указано имя POV-файла вашей сцены; затем взгляните на следующую строчку **Antialias=off**, включающую/выключающую антиалиасинг. Замените **off** на **on**, активировав функцию. Следующий параметр, **Sampling_Method**, переключает режимы сглаживания. Если он равен 1, то применяется обычный антиалиасинг, при котором пиксель разбивается на множество сэмплов, и их усреднённое значение присваивается пикселю. Число сэмплов одинаково для всех пикселей и равно квадрату параметра **Antialias_Depth**. Если **Sampling_Method** равен 2, используется рекурсивный антиалиасинг. В этом режиме пиксель разбивается на 4 части, если различие между частями больше, чем величина **Antialias_Threshold** (порог антиалиасинга), и каждая часть снова разбивается. Операция повторяется до тех пор, пока разница между сэмплами не станет меньше порога или пока число рекурсивных разбиений не превысит величину **Antialias_Depth**. Данный метод позволяет гибко регулировать число сэмплов, в зависимости от сложности поверхности, проецируемой на пиксель, экономия вычислительных ресурсов без ущерба для качества финального изображения.

Выберем второй метод антиалиасинга, присвоив двойку параметру **Sampling_Method**; **Antialias_Depth** будет равно двум, **Antialias_Threshold** равен **0.01**. Запускаем рендер командой **povray tower.ini**. Теперь придется подождать довольно долго – возможно, час или более, если у вас не самый мощный компьютер – но награда за терпение будет чистой картинка без артефактов.

На этом наше шапочное знакомство с POV-Ray закончено. За пределами рассмотрения осталась масса полезных функций: имитация дифракции и дисперсии с помощью карт фотонов, реалистичный расчёт рассеянного освещения, сплайновые поверхности вращения и много чего ещё. Надеюсь, что нам удастся заняться этим в последующем цикле уроков, полностью посвящённых POV-Ray. **LXF**

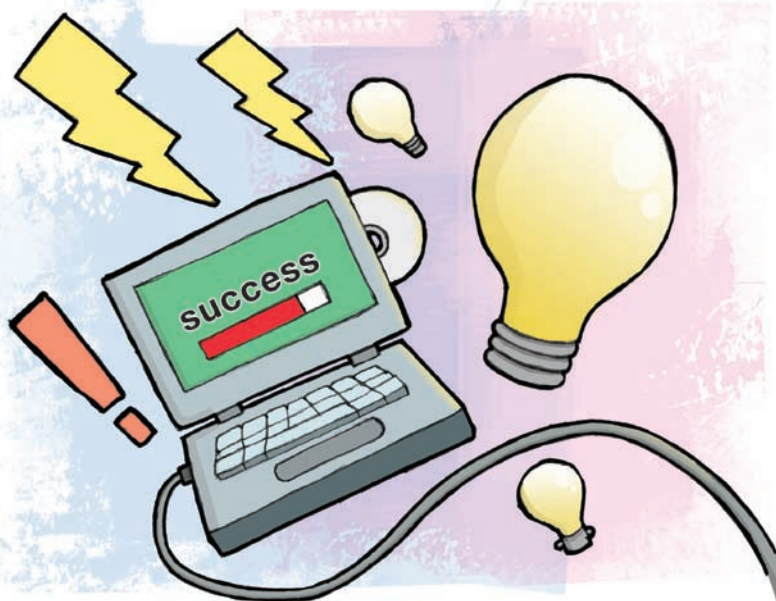


» Антиалиасинг убрал все дефекты.



Инциденты: Най-

Систему отслеживания ошибок предоставляет практически любая служба хостинга проектов, но вы можете захотеть сделать все сами. Если это так, то **Никита Шультайс** готов вам помочь.



Наш эксперт

Никита Шультайс

Профессиональный web-разработчик и преподаватель; не понаслышке знает, как избавиться от ошибок в коде, и умеет об этом рассказать.

Если вы пишете небольшую программу для однокурсницы, то возникающие в ней ошибки нетрудно удержать и в голове; но когда речь заходит о серьезных проектах, без системы отслеживания ошибок уже не обойтись. Сегодня можно найти множество таких программ-«багтрекеров» [англ. bug tracker], причем распространяемых под свободными лицензиями, но мы остановимся на *Mantis*.

Установка

Скачайте *Mantis* с официального сайта (www.mantisbt.org), разместите его на своем сервере под управлением *Apache* и перейдите в браузере по заданному для *Mantis* адресу, чтобы приступить непосредственно к установке.

На первом этапе происходит прединсталляционная диагностика: *Mantis* проверяет, установлен ли PHP и поддерживает ли он безопасный режим. Тут же нам предлагается выбрать тип базы данных (*MySQL*, *MS SQL*, *PostgreSQL*, *Oracle* и др.), причем большая часть из них работает в экспериментальном режиме. Далее необходимо ввести параметры БД: имя и пароль для доступа, название базы и имя и пароль администратора БД (если требуется создать новую базу данных). После нажимайте Install/Upgrade Database и смотрите на диагностические сообщения.

Обычно всё проходит отлично, и вы видите напротив каждой строки слово GOOD на зеленом фоне. Самая частая причина сбоя – запрет на запись в директорию проекта, из-за чего *Mantis* не может создать конфигурационный файл `config_inc.php`, поэтому вам надо либо заранее обеспечить *Apache* соответствующими правами, либо переименовать `config_inc.php.sample` в `config_inc.php` в корне проекта вручную и задать необходимые настройки в файле: их можно посмотреть на той же web-странице в разделе

Write Configuration File. После успешной установки откройте в браузере ваш сайт; вам будет предложено ввести имя и пароль, а также удалить учетную запись administrator или сменить ей пароль (root по умолчанию) и удалить папку admin – проделайте это. Авторизуйтесь и первым делом перейдите по ссылке MyAccount, где в поля Password и Confirm password введите новый пароль, затем снова авторизуйтесь и в меню MyAccount > Preferences смените язык [Language] с английского на русский.

Базовые настройки

Практически вся настройка системы производится вручную с помощью файла `config_inc.php`, и первое, что мы предпримем – русифицируем *Mantis*. До этого мы выбрали язык для единственного пользователя системы – администратора, но если вы зайдете на сайт, например, из другого браузера, перед вами откроется англоязычный интерфейс. Чтобы поправить дело, добавьте в `config-inc.php` две строки:

```
$g_default_language = 'russian';
$g_language_choices_arr = array('english', 'russian');
```

Первая отвечает за язык по умолчанию, а вторая – за доступные языки. В нашем случае по умолчанию выбран русский, а в настройках учетной записи можно указать еще и английский.

Чтобы задать название окна браузера и вкладок, используйте `$g_window_title`, а заголовок, который отображается на всех страницах сверху и по центру, задается в переменной `$g_page_title`. Переменные `$g_favicon_image` и `$g_logo_image` содержат путь до иконки сайта (favicon) и логотипа соответственно. Эти переменные принимают путь до изображений относительно директории, в которой установлен *Mantis*, например, 'images/my_logo.jpg'. Конечно, картинка должна там уже находиться.

Параметр `$g_show_realname = ON`; заставляет систему показывать реальное имя пользователя, а не ник. Заметьте, что вокруг ON нет кавычек, так как это не строка, а заранее определенная переменная. А если добавить в конфигурационный файл строку `$g_show_avatar = ON`; то к комментариям будут подгружаться аватары с сайта www.gravatar.com – конечно, если у ваших пользователей есть на нем учетная запись.

Перед началом практического использования *Mantis* необходимо также произвести несколько нехитрых действий: задать категории ошибок, создать пользователей и проекты.

Категории ошибок

Любая ошибка или нововведение, регистрируемые в системе, должны быть отнесены к определенной категории – например, при разработке сайта (далее по тексту мы будем настраивать *Mantis* под web-проект) можно выделить категории

- » CSS
- » HTML
- » Ввод данных
- » Панель администрирования
- » Фильтрация данных

ТИ И ОБЕЗВРЕДИТЬ

В *Mantis* существуют глобальные и локальные категории. Первые доступны для всех проектов, у которых выбрано поле Наследовать глобальные категории; локальные же создаются для каждого проекта в отдельности. Добавить глобальные категории можно на странице Управление > Управление проектами. Помимо названия, для каждой категории можно назначить ответственного: скажем, при регистрации ошибки в CSS или HTML она сразу передается верстальщику. Если ответственный не определен, то ошибки сперва просматривает менеджер проекта, который назначает исполнителя. Заметим, что ответственный за категорию должен иметь уровень доступа не ниже участника.

Управляем пользователями

Чтобы назначать для категорий ответственных, нужно сначала добавить их в систему. Это делается на странице Управление > Управление пользователями. Нажмите на кнопку Создание новой учетной записи и заполните несколько несложных полей. После создания пользователя вас перенаправят на страницу управления его учетной записью, где можно настроить все поля более детально.

В *Mantis* существует несколько уровней доступа. Каждый следующий уровень получает все возможности предыдущего плюс свои собственные. Например,

- » **Наблюдатель** имеет минимальные возможности, он может лишь просматривать ошибки.
- » **Инициатор** может самостоятельно добавлять записи о найденных ошибках, отслеживать их, а также устанавливать видимость для вопроса. Такой тип пользователя отлично подойдет для тестеров.
- » **Редактор** может изменять основные параметры ошибки: серьезность, воспроизводимость, состояние, приоритет, видимость, решение и т.д.
- » **Участник, руководитель и администратор** имеют почти одинаковый набор прав, ознакомьтесь с которыми и изменить их можно на странице Управление > Управление конфигурацией > Пороги процесса.

Один пользователь может иметь несколько уровней доступа: один глобальный и несколько индивидуальных (для каждого из проектов). Если в проекте явно не указывать уровень доступа пользователя, то используется глобальный.

Теперь, когда учетная запись создана, можно назначить её ответственной за какую-нибудь категорию.

Если вы разрабатываете коммерческий проект или же у вас есть другие причины ограничить доступ к *Mantis*, то в конфигурационный файл следует добавить строку

```
$g_allow_signup = OFF;
```

которая запретит пользователям регистрироваться в системе. Если мы добавим `$g_send_reset_password = OFF;` то забывчивые пользователи не смогут восстановить свой пароль в автоматическом режиме: им придется обращаться к администратору сайта.

```
$g_max_failed_login_count = OFF;
```

снимает ограничение по количеству попыток авторизоваться. Замените OFF на какое-нибудь число, и злоумышленники не смогут бесконечно подбирать пароли учетных записей. С помощью параметра `$g_signup_use_captcha` можно включить или выключить отображение кода подтверждения (Captcha) при регистрации.

НОВОСТИ							
Возможность	наблюдатель	инициатор	редактор	участник	руководитель	администратор	
Просмотр новостей с ограниченной видимостью				✓	✓	✓	
Управление новостями					✓	✓	
ВЛОЖЕНИЯ							
Возможность	наблюдатель	инициатор	редактор	участник	руководитель	администратор	
Просмотр списка вложений	✓	✓	✓	✓	✓	✓	
Загрузка вложений	✓	✓	✓	✓	✓	✓	
Удаление вложений					✓	✓	
Добавление вложения к инциденту		✓	✓	✓	✓	✓	
ФИЛЬТРЫ							
Возможность	наблюдатель	инициатор	редактор	участник	руководитель	администратор	
Сохранение фильтров				✓	✓	✓	
Сохранение фильтров в качестве общих					✓	✓	
Использование сохраненных фильтров		✓	✓	✓	✓	✓	

Создание проекта

Mantis – многопользовательская и многопроектная среда, и прежде чем добавлять ошибки, нужно создать хотя бы один проект. Перейдите в меню Управление > Управление проектами и нажмите на кнопку Создать новый проект. Перед вами появится форма с несколькими полями:

- » **Название проекта.**
- » **Состояние** Разрабатываемый, выпущенный, стабильный, устаревший.
- » **Видимость** Если выбрать Общая, то ошибки, зарегистрированные для проекта, будут доступны любому пользователю; если выбрать Ограниченная, то данный проект смогут увидеть только те пользователи, которые «связаны» с проектом вручную, и администраторы.

» **Наследовать глобальные категории** О них мы писали выше.

- » **Путь для загружаемых файлов** Можно указать как абсолютный путь, так и путь от каталога, в котором находится *Mantis*. Главное, чтобы все директории существовали и у web-сервера были права на запись в них. По умолчанию файлы всё равно будут сохраняться в базе данных; чтобы изменить такое поведение, добавьте строку `$g_file_upload_method = DISK;` в конфигурационный файл. Кроме загрузки на диск и в базу данных, можно передавать файлы по FTP. Необходимые для этого настройки указаны в документации. Так же можно задать максимальный размер загружаемых файлов, допустимые и запрещенные форматы.

» Для более детального описания проекта заполните поле **Подробности**.

После того как проект создан, становятся доступными множество настроек: надо лишь перейти на страницу проекта, которая состоит из нескольких блоков. Сначала идет блок подпроектов – более мелких проектов, входящих в крупный. Например, при разработке городского интернет-портала можно выделить в подпроекты новостной портал, сервис поиска работы, продажу недвижимости и т.д. Затем идет блок категорий: сверху расположены глобальные категории, внизу – локальные, которые задаются тут же на странице. В следующем блоке мы создаем версии проекта. Проставляя галочки напротив соответствующих полей, можно отметить версию как выпущенную или устаревшую, или, если оставить поля пустыми, то версия будет находиться в процессе разработки. Самый нижний блок отвечает за пользова-

» Возможности различных ролей в системе *Mantis*.

»

№	Номер	Категория	Серьезность	Состояние	Изменен	Суть	Целевая версия
0000166	Поиск пользователей	блокада	отработан (Никита Шульгайс)	17.05.10	Сайт падает если человек родился в 2010 году	0.7	
0000163	Добавление/изменение упражнений	авария	отработан (Никита Шульгайс)	17.05.10	TemplateDoesNotExist: 404.html	0.7	
0000156	Главная страница	налая	назначен (Никита Шульгайс)	17.05.10	Не отображается последнее изменение веса	0.7	
0000172	Прочее	налая	назначен (Никита Шульгайс)	18.05.10	Ниже находится список людей, у которых Светлана добавлен в друзья	0.7	
0000155	Прочее	налая	назначен (Никита Шульгайс)	18.05.10	Разработка дневника питания	0.7	
0000171	Добавление/изменение упражнений	неудобство	назначен (Никита Шульгайс)	17.05.10	Добавить вспомогательный текст про переназначение клавишей Tab	0.7	
0000165	Прочее	налая	отработан (Никита Шульгайс)	17.05.10	При авторизации нет перенаправления	0.7	
0000161	Прочее	налая	назначен (Никита Шульгайс)	17.05.10	Социализация проекта	0.7	
0000157	Прочее	налая	новый	17.05.10	Не находится favicon при выборе пользователей	0.7	

Так выглядит список инцидентов.

телей, которые должны получить доступ к проекту. Если видимость проекта задана как Общая и нет ни одного пользователя, связанного с ним, то все участники будут иметь доступ в соответствии с настройками своей учетной записи. Если видимость проекта ограничена, нужно назначить каждого участника и его роль в проекте вручную. Для этого выберите нужного человека из списка пользователей, которые доступны в системе, определите для него уровень доступа и нажмите на кнопку Добавить.

Когда пользователь добавится в проект, в окне Управление учетными записями в поле Действия для этого участника будет доступна кнопка Удалить, которая не удаляет пользователя из системы, а лишь разрывает его связь с проектом.

Зарегистрируем ошибку

Пройдя сквозь дебри настроек, мы добрались до регистрации ошибок, которые в версии 1.2 называются «инцидентами», а раньше были «вопросами». На самом деле под инцидентом может регистрироваться не только ошибка, но и обращение пользователя в службу поддержки или запланированное нововведение.

Кликните на ссылку Создать инцидент в верхнем меню. Перед вами откроется форма со множеством полей, которые нам предстоит заполнить. В первой строчке мы выбираем категорию вопроса из списка, сформированного нами ранее. Следующее поле отвечает за воспроизводимость ошибки. Одни проблемы возникают у всех пользователей, другие – только в определенных версиях браузера, а ошибки проектирования вроде Race Condition проявляются в случайные моменты времени и «затишают» при попытке их локализовать.

Далее определяется серьезность ошибки, причем первым в списке стоит пункт Нововведение, который отвечает за будущие возможности программы. Остальные пункты следуют в порядке увеличения серьезности. Еще одним параметром важности ошибки является Приоритет её обработки и решения.

Следующая группа полей отвечает за платформу, на которой зарегистрирована ошибка, а также за операционную систему и её версию. Нам предлагается выбрать профиль или заполнить

эту Целевая версия, т.е. версия, в которой предполагается решить эту проблему. Если выбрать разрабатываемую версию продукта, то инцидент попадет на страницу План развития, ссылка на которую находится в верхнем меню. Вы всегда можете зайти на эту страницу и посмотреть, как обстоят дела с новыми проектами, сколько процентов нововведений и ошибок уже закрыто и какие остались до запуска.

Между текущей и целевой версией находится поле Назначить, в котором можно выбрать ответственного за ошибку. Если вы уже выбрали категорию, к которой прикреплен исполнитель, то можно оставлять поле пустым; его также можно не заполнять, пока не станет ясно, кто будет отвечать за инцидент. Если для категории определен исполнитель, а вы назначаете другого участника, то значение по умолчанию будет переопределено. Заметим, что задавать целевую версию и исполнителя могут только члены с уровнем доступа не ниже участника. Инициаторы и редакторы могут только указывать версию программы, в которой была обнаружена ошибка.

Теперь самое время описать саму ошибку. Для этого предназначены обязательные поля Суть и Подробности, а также необязательные Шаги по воспроизведению и Дополнительные сведения. В конце вы можете приложить к ошибке файл – например, экранный снимок – а также указать для ошибки видимость. Если есть вероятность того, что ошибку могут увидеть злоумышленники и затем использовать её в своих целях, видимость следует ограничить. Если же большая часть ошибок не должна быть доступна общественности, то добавьте в конфигурационный файл строку \$g_default_bug_view_status = VS_PRIVATE; чтобы по умолчанию выбиралась ограниченная видимость.

Нажмите на кнопку Сохранить инцидент, чтобы зарегистрировать ошибку в базе данных.

Настроим поля

Как я уже упомянул, для web-проектов нет большой разницы, в какой ОС они работают, а вот в каком браузере – есть. Правда, в форме добавления инцидента нет такого поля, но мы можем его

создать. Зайдите в Управление > Управление настраиваемыми полями, введите название поля (латиницей, browser) и нажмите Создать настраиваемое поле, после чего вас перенесет на страницу детальной конфигурации. Здесь вы можете изменить имя поля, задать его тип (список, целое, вещественное и т.д.) или определить допустимые значения с помощью регулярного выражения. Вы также можете выбрать значение по умолчанию и задавать параметры доступа и отображения поля.

Добавим список со множественным выбором из популярных интернет-браузеров – для этого выберите соответствующий тип, а в поле Допустимые

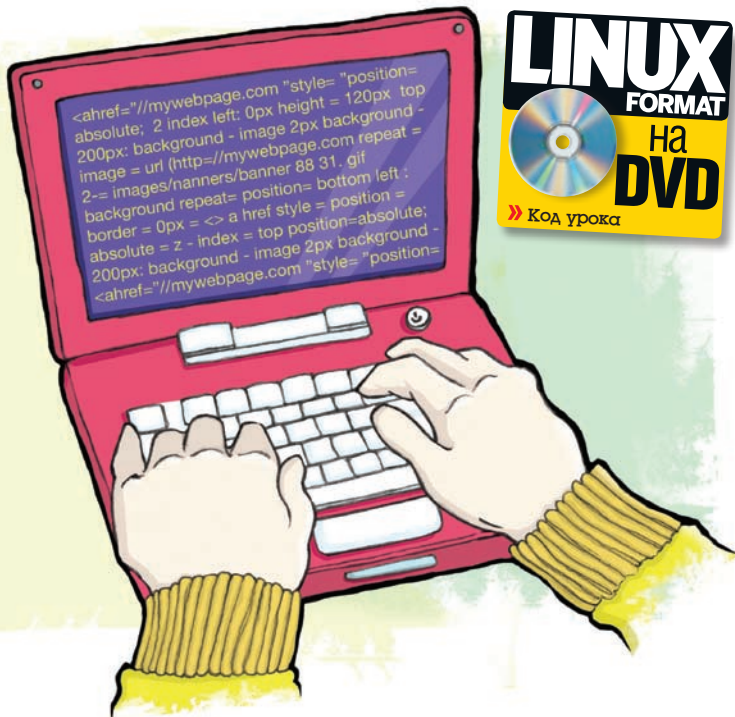
Ошибки в Mantis могут находиться в различных состояниях.

Текущее состояние	Следующее состояние							Значение по умолчанию
	новый	нужен отклик	рассмотрен	подтвержден	назначен	отработан	закрыт	
новый		☑	☑	☑	☑	☑	☑	нужен отклик
нужен отклик	☑		☑	☑	☑	☑	☑	новый
рассмотрен	☑	☑		☑	☑	☑	☑	новый
подтвержден	☑	☑	☑		☑	☑	☑	новый
назначен	☑	☑	☑	☑		☑	☑	новый
отработан	☑	☑ (повторно открыт)	☑	☑	☑		☑	новый
закрыт	☑	☑ (повторно открыт)	☑	☑	☑	☑		новый

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

Bash: Простые и

Написать полезный скрипт просто. В первой части нашей новой серии о программировании **Ник Вейч** приглашает вас порезвиться с *Bash*.



Наш эксперт

По **Ник Вейчу**, лучший способ заставить что-то работать – дать ему по *Bash*’ке.

А ведь вы уже можете писать скрипты (сценарии) *Bash* – разве что они не очень сложны. Рискну предположить, что чаще всего в командной строке вы применяете такой скрипт:

```
> ls
a.png a.txt bash1.sh b.png b.txt c.txt x.jpg y.jpg z.gif
```

Да, это скрипт *Bash*. Нет, правда – но только в нем всего одна строка. В нем не используются специальные возможности *Bash*, просто выполняется простая команда для вывода содержимого текущего каталога. Все еще не верите? Сделаем кое-что еще. Скажем, вы рискнули вывести содержимое каталога в подробном виде:

```
> ls -l
total 4
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 a.png
-rw-rw-r--. 1 evilnick evilnick 2868 Jun 29 12:55 a.txt
-rw-rw-r--. 1 evilnick evilnick 151 Jun 29 12:49 bash1.sh
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 b.png
-rw-rw-r--. 1 evilnick evilnick 223 Jun 29 12:55 b.txt
-rw-rw-r--. 1 evilnick evilnick 120 Jun 29 12:56 c.txt
-rw-rw-r--. 1 evilnick evilnick 3486 Jun 29 12:56 x.jpg
-rw-rw-r--. 1 evilnick evilnick 3486 Jun 29 12:56 y.jpg
-rw-rw-r--. 1 evilnick evilnick 19273 Jun 29 12:56 z.gif
```

Ну не круто? Мы воспользовались «ключом», опцией команды *ls*, который выводит более подробную информацию. Теперь, допустим, нужно вывести список всех файлов изображений. Как это сделать? Попробуем:

```
> ls -l *.png
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 a.png
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 b.png
```

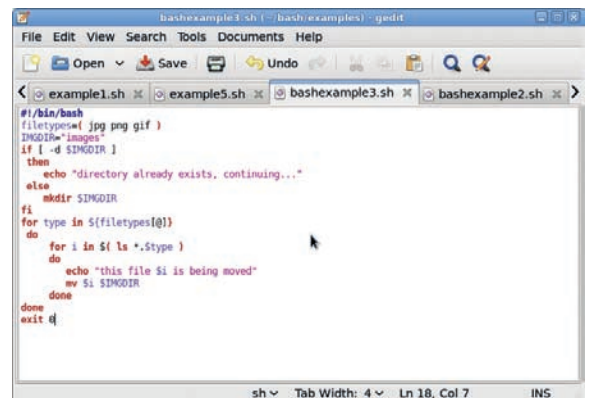
Ага! Спецсимвол *** – не часть команды *ls*, а часть оболочки *Bash*. *Bash* интерпретирует команду, встречает спецсимвол и заменяет его соответствующими значениями. Итак, мы пишем скрипт *Bash*! Одна из выгод добавочных знаний о *Bash* и о работе оболочки в том, что вы можете взглянуть на проблему шире и сделать что-то чуть более умное и полезное:

```
> ls -l *.gif,*.jpg
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 a.png
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 b.png
-rw-rw-r--. 1 evilnick evilnick 3486 Jun 29 12:56 x.jpg
-rw-rw-r--. 1 evilnick evilnick 3486 Jun 29 12:56 y.jpg
-rw-rw-r--. 1 evilnick evilnick 19273 Jun 29 12:56 z.gif
```

Что же тут произошло? Это одна из самых простых уловок, которые можно найти в скриптах *Bash* – раскрытие скобок. В общем случае, *Bash* раскрывает содержимое скобок как все, что идет перед скобками (или после) плюс каждый отдельный элемент внутри скобок. В данном случае, после раскрытия скобок мы получим команду *ls -l *.gif *.jpg *.png*.

Команда стала не намного короче, но немного изящнее, и если список возможных расширений файлов будет расти, вы сэкономите время и силы. Применяйте этот прием в будущих скриптах.

Теперь, убедившись, что изучение скриптов *Bash* может весьма пригодиться, приостановимся и рассмотрим, что же такое *Bash*.



» Если вы беретесь за дело всерьез, найдите себе удобный текстовый редактор. В *Emacs*, *Vi*, *Gedit* и *Kate* есть подсветка синтаксиса для *Bash*, и это плюс.

» **Месяц назад** Мы завершили серию уроков по Python написанием программы.

быстрые скрипты

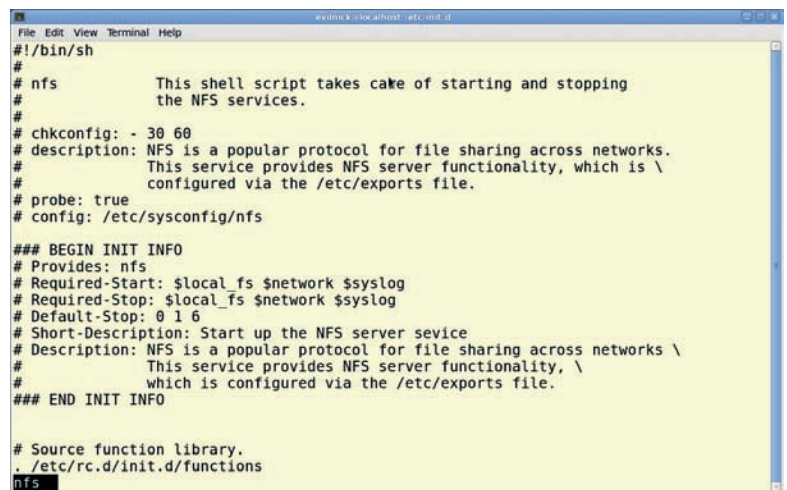
Среда оболочки – просто среда: сцена или окружение, в котором обычные команды могут выполнять свои обязанности. Почти каждая команда, которую вы когда-либо выполняли и которая что-то делает, не зависит от оболочки, но без оболочки у этих команд не будет рабочего контекста. Скрытая за внешним лоском шикарного Gnome/KDE/другого графического интерфейса оболочка жизненно необходима для работы компьютера – он и запуститься не смог бы без среды, где выполняются команды. Строка за строкой, скромные скрипты *Bash* при загрузке системы настраивают сеть, монтируют устройства, запускают сервисы, открывают оконный менеджер; и множество скриптов работает в фоновом режиме, выполняя скрытые задачи поддержки ОС в рабочем состоянии.

Сделаем больше

Многие скажут, что *Bash* хорош для простых задач, но не годится, когда дело доходит до серьезных вещей, вроде взаимодействия по сети. Хотя в таких случаях подход *Bash* и в самом деле немного труднее и нуднее (а также и менее познаваем), это отнюдь не значит, что пользоваться им нельзя, и мы рассмотрим несколько подобных примеров в будущем. А что касается причины пользоваться *Bash*, когда то же самое легко сделать на Python или PHP... ну, иногда *Bash* – это все, что у вас есть. Мы уже отметили, что практически во всех версиях Linux, встраиваемых и прочих, есть *Bash* или *Bash*-подобная среда; поэтому не худо и знать, на что способны даже простейшие утилиты.

Первое, что следует рассмотреть – как мы поступили бы в случае языка программирования – это использование переменных. О разных типах данных забудьте, поскольку тип в основном один: строки. Но строки очень гибкие. Взгляните на короткий пример:

```
> foo="ploppy"
> echo $foo
ploppy
> echo $foo+1
ploppy+1
> foo="ls"
> echo $foo
ls
> foo="$foo -al"
> echo $foo
ls -al
> echo 'foo'
foo
> echo "foo"
foo
ls -al
> $foo *.png
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 a.png
-rw-rw-r--. 1 evilnick evilnick 2450 Jun 29 12:55 b.png
```



» Ваш компьютер с Linux нафарширован скриптами. В /etc/init.d вы найдете примеры скриптов, управляющих системными сервисами.

Прежде всего отметим, что переменная определяется путем ввода ее имени, знака равенства и значения. Не окружайте знак равенства пробелами, иначе *Bash* выдаст ошибку. Имена переменных должны начинаться с буквы или символа подчеркивания и могут содержать цифры, буквы и символы подчеркивания. »

Хитрости с клавишами

В *Bash* предусмотрено множество комбинаций клавиш, и они более или менее соответствуют комбинациям клавиш *Emacs*, так что если вы с ними знакомы, это преимущество. Вот несколько самых ценных – по крайней мере, попробуйте запомнить те, что покажутся вам полезными:

- HOME** Перемещает курсор в начало строки.
- END** Перемещает курсор в конец строки.
- Стрелка вверх** Восстанавливает предыдущую команду.
- Стрелка вниз** Восстанавливает следующую команду.
- Ctrl+R** Ищет последнюю команду в истории, содержащую указанные символы.
- Ctrl+O** Выполняет команду, найденную с помощью Ctrl+R.
- Ctrl+L** Очищает экран.
- Ctrl+U** Очищает содержимое строки перед курсором.
- Ctrl+K** Очищает содержимое строки после курсора.
- Ctrl+C** Отправляет текущей задаче сигнал **SIGINT**, который обычно завершает ее.
- Ctrl+Z** Отправляет текущей задаче сигнал **SIGTSTP**, который приостановит ее. (Затем ее можно возобновить командой **fg**). Чтобы вернуться к ней позже, можно набрать **fg** [имя процесса или идентификатор задания].
- Alt+F** Перемещает курсор на одно слово вперед в строке.
- Alt+B** Перемещает курсор на одно слово назад в строке.
- Alt+Del** Удаляет слово перед курсором.
- Alt+D** Удаляет слово после курсора.

» Пропустили номер? Узнайте на с. 107, как получить его прямо сейчас.

Есть соглашение задавать все имена переменных в верхнем регистре. Однако присутствует множество встроенных переменных и переменных окружения, и задавая имена собственных переменных в нижнем регистре, часто можно избежать конфликта имен.

Так как переменная является строкой, установка значения «нечто+1» просто допишет к строке '+1'. Об арифметических операциях мы вкратце поговорим потом.

Знак \$ перед именем переменной внутри двойных кавычек приведет к тому, что ее значение будет раскрыто. А внутри одиночных кавычек *Bash* не делает никаких подстановок, и вы получите просто литерал.

Для раскрытия переменной кавычки также можно вообще опустить. Это довольно удобно, так как означает, что можно сохранять и выполнять команды из переменных, в чем мы убеждаемся в последнем примере.

Нечто полезное

Прежде чем погрязать в деталях, рассмотрим одну штуку, способную пригодиться при написании скрипта. Вот полный скрипт, который можно запустить на домашнем компьютере. Всякие там переменные делают в нем свои делишки. Наберите этот код в текстовом редакторе, сохраните под именем `example1.sh` и запустите из оболочки командой `bash example1.sh`:

```
#!/bin/bash
ROOT_UID=0
ERROR_PERMISSION=77
if [ "$UID" != "$ROOT_UID" ]
then
echo "You must run this script as root"
exit $ERROR_PERMISSION
fi
exit 0
```

Первая строка начинается с комбинации символов, часто называемой «шалаш» [shebang] – `#!`. Это особый вид строки комментария, которая сообщает вызвавшей ее оболочке, что если строку выполнять как команду, ее нужно передать исполняемому файлу,

путь к которому указан далее. Короче говоря, это просто способ выполнить скрипт. Перейдем к более интересному...

Помните, мы говорили, что из *Bash* можно обращаться к переменным окружения? `UID` – одна из этих переменных; она содержит идентификатор пользователя (в виде числа), запустившего оболочку *Bash*. Это число равно 0 для суперпользователя-`root`, а для обыкновенных пользователей зависит от используемого

дистрибутива; обычно для первого пользователя берется 501. Выражение `if` – стандартный оператор условия. Анализируемое условие помещается в квадратных скобках. Здесь мы используем оператор сравнения `!=`, который означает «не равно» (и его также можно записать как `-ne`). Отсюда ясно, что если идентификатор пользователя равен значению переменной для `root`, условие внутри квадратных скобок ложно и скрипт переходит к выражению `fi`, после сравнения. Если значения не равны, выполнение переходит к командам после `then`, мы получаем сообщение и скрипт завершается.

Не считая ряда зарезервированных значений (0–2, 126–165 и 255), жестких правил насчет кодов выхода нет. По соглашению, для ошибок, связанных с отказом в доступе, часто используется 77, но можно употреблять и свои коды выхода; главное – помнить, что они означают.

Итак, наш скрипт ничего не делает, но может стать началом другого, более плодотворного. Он достаточно безопасен – обычно из *Bash* нельзя изменить значение `UID`, но более полезен как проverka перед выполнением неких действий, требующих прав доступа `root`. *Bash* поддерживает конструкции `else` и `elif`; также существуют особые операторы сравнения. Следующий маленький скрипт проверяет, существует ли заданный каталог, и выводит соответствующее сообщение:

```
#!/bin/bash
CHECKDIR="examples"
if [ -d $CHECKDIR ]
then
echo "examples directory exists"
else
echo "examples directory does not exist!"
fi
exit 0
```

Параметр `-d` внутри квадратных скобок – специальный оператор, проверяющий, существует ли каталог в файловой системе. Следующий аргумент – имя каталога; в данном случае мы передали его через переменную. Другие операторы сравнения см. во врезке справа вверху на соседней странице.

Параметр `-d` внутри квадратных скобок – специальный оператор, проверяющий, существует ли каталог в файловой системе. Следующий аргумент – имя каталога; в данном случае мы передали его через переменную. Другие операторы сравнения см. во врезке справа вверху на соседней странице.

Защиваемся

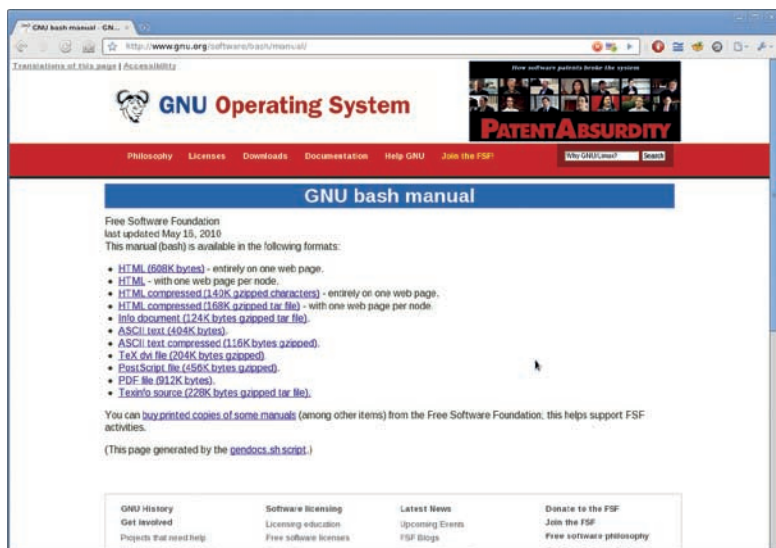
Мы уже познакомились со стандартным оператором сравнения. Другая конструкция, часто применяемая в программировании – цикл. В *Bash* существуют различные типы циклов, но чаще всего используется `for...do...done`. В *Bash* его можно сделать очень похожим на цикл в C или в других языках, которыми вы, возможно, пользовались:

```
for (( i = 0 ; i < 10 ; i++ ))
do
echo $i
done
```

Впрочем, как и в Python, итерация в *Bash* обычно осуществляется по списку значений – в нашем примере его генерируют



Стоит ввести в привычку заканчивать скрипт командой `exit 0`, если все действия завершились успешно. При связке скриптов друг с другом условия выхода важны.



Если у вас возникнут вопросы, то на домашней странице *Bash* есть ссылка на довольно удачное руководство: www.gnu.org/software/bash/manual.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

внутренние скобки. Чаще всего вы увидите циклы с итерациями по массиву или по списку, сгенерированному из других значений. Например:

```
for i in $(ls)
do
echo $i
done
```

Bash, как всегда, вычисляет содержимое скобок. В данном случае это команда, и *Bash* ее выполняет и создает список результатов, по которому затем будет производиться итерация. Позже мы подробнее поговорим о массивах (которые в *Bash* по сути одномерные и больше похожи на объект списка в Python), но чаще всего они используются таким образом:

```
> array=(becky tabitha sue)
> echo $array
becky
> echo $array[1]
becky[1]
> echo ${array[1]}
tabitha
> echo ${array[@]}
becky tabitha sue
> for i in ${array[@]}
> do
> echo $i
> done
becky
tabitha
sue
```

Просекли фишку? Посмотрите снова: квадратные скобки используются для ссылки на массив по индексам, но *Bash* не обязан знать, что приставленное в конце чего-то **[1]** означает индекс. В порядке обхода, нужно после **\$** окружить все, что относится к переменной, в фигурные скобки, гарантируя обработку этого как единой сущности. Если попробовать получить доступ к массиву без индекса, вы просто получите первый элемент (а индексы начинаются с 0). Индекс **[@]** — особое значение, которое расширяется до всех значений, хранимых в массиве, и мы используем его в последнем цикле **for** для генерации списка.

И в итоге

Вооружившись изученными приемами, напишем нечто полезное: короткий скрипт. Будучи запущен в каталоге, он найдет все файлы изображений определенного типа и переместит их в собственный подкаталог **images**:

```
#!/bin/bash
filetypes=( jpg png gif )
IMGDIR="images"
if [ -d $IMGDIR ]
then
echo "directory already exists, continuing..."
else
mkdir $IMGDIR
fi
for type in ${filetypes[@]}
do
for i in $( ls *. $type )
do
echo "this file $i is being moved"
mv $i $IMGDIR
done
done
exit 0
```

Это можно сделать и компактнее, но по коду, структурированному таким образом, проще понять, что происходит. Мы создали мас-

Другие полезные сравнения

Для других вариантов условий помещайте в скобки оператора сравнения такие конструкции:

- [-a \$FILE] Истинно, если **\$FILE** существует.
- [-d \$FILE] Истинно, если **\$FILE** существует и это каталог.
- [-f \$FILE] Истинно, если **\$FILE** существует и это обыкновенный файл.
- [-h \$FILE] Истинно, если **\$FILE** существует и это символическая ссылка.
- [-r \$FILE] Истинно, если **\$FILE** существует и его можно прочитать.
- [-s \$FILE] Истинно, если **\$FILE** существует и его размер больше нуля.
- [-w \$FILE] Истинно, если **\$FILE** существует и в него можно писать.
- [-x \$FILE] Истинно, если **\$FILE** существует и он исполняемый.
- [-O \$FILE] Истинно, если **\$FILE** существует и его владельцем является пользователь с действующим идентификатором пользователя.
- [-N \$FILE] Истинно, если **\$FILE** существует и изменился с момента последнего чтения.
- [\$FILE1 -nt \$FILE2] Истинно, если **\$FILE1** изменился позже **\$FILE2** или если **\$FILE1** существует, а **\$FILE2** нет.
- [\$FILE1 -ot \$FILE2] Истинно, если **\$FILE1** старше, чем **\$FILE2**, или если **\$FILE2** существует, а **\$FILE1** — нет.
- [\$FILE1 -ef \$FILE2] Истинно, если **\$FILE1** и **\$FILE2** ссылаются на один и тот же файл.

сив для хранения типов файлов, которые нужно отобразить; затем воспользовались оператором **if...**, чтобы понять, существует ли каталог назначения, или создать его в противном случае.

Первый цикл пробегает по элементам массива. В начале второго цикла выполняется команда **ls** для получения списка соответствующих файлов, затем с помощью команды **mv** файлы переносятся в каталог назначения.

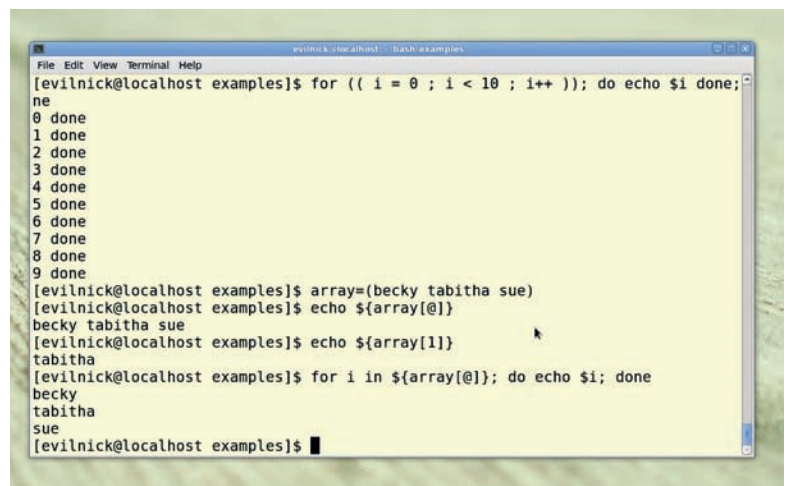
Сохранив файл, выполните команду

```
chmod ugo+x sortimage.sh
```

Она установит бит исполнения для скрипта, и его можно будет запускать напрямую, без команды **bash**.

В текущей версии скрипта эта команда будет получать рабочий каталог из любого места в файловой системе, откуда вы ее запускаете. Файл можно было бы сохранить в **/usr/sbin** или где угодно еще в пути команды и вызывать ее откуда вам заблагорассудится.

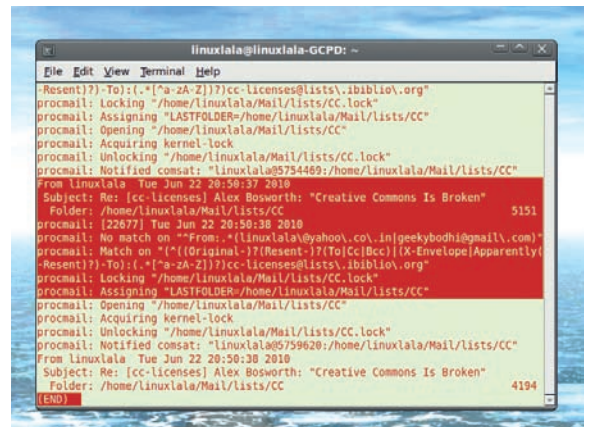
Итак, мы обнаружили, что вы знаете о *Bash* гораздо больше, чем вам казалось, узнали, как пользоваться переменными, и познакомились с основными конструкциями скрипта. На следующем уроке мы поговорим о включении в скрипт внешних команд и их запуске через *Bash*, и добавим интерактивности вашим программам, написав собственный скрипт резервного копирования. До встречи! **LXF**



➤ Для создания скриптов *Bash* не требуется ничего, кроме терминала. Настройте его понагляднее и создайте каталог для экспериментов.

Prostmail: Почте

Шашанк Шарма снова резвится в сияющем раю командной строки и находит *Prostmail* идеальным решением для фильтрации почты.



» Файл *procmailer.log* сообщает, в какую папку попал каждый файл – ищите там сообщения, угордившие не туда.

Наш эксперт

Шашанк Шарма пишет о свободном ПО более четырех лет, для различных изданий, включая Linux.com. Он также является соавтором книги "Beginning Fedora" издательства Apress.

Необходимость отдельного решения для фильтрации почты прямо пропорциональна количеству писем в вашем ящике. *Prostmail* – мощная утилита обработки сообщений, способная производить сортировку почты на основе любого количества правил. Входящие сообщения можно распределить по папкам на основе заранее заданного критерия, такого как тема сообщения, отправитель и т.д. Применение в правилах регулярных выражений и возможность использования множества правил для сортировки писем превращают *Prostmail* в весьма точный почтовый фильтр. Кроме того, он легко отсеивает спам и сообщения с исполняемыми файлами во вложении, а также выступает в роли идеального автоответчика.

Так как активная разработка *Prostmail* прекращена, он поставляется по умолчанию лишь в немногих дистрибутивах. Но пусть это вас не беспокоит: он весьма стабилен и популярен, и его легко добыть в репозиториях.

На типичном почтовом сервере одновременно работают множество различных утилит и технологий. Они включают агента доставки почты (MDA), агента получения почты (MRA) и пользовательского почтового агента (MUA). Последние – это почтовые клиенты, типа *Mutt* или *Thunderbird*. Агенты получения почты – программы вроде *Fetchmail* или *Getmail*; они получают сообщения с удаленной учетной записи. *Prostmail* – агент доставки почты, отвечающий за сортировку сообщений и передачу их в почтовый ящик в зависимости от настроек.

За действия *Prostmail* по фильтрации почты отвечает файл *.procmailer* в вашем домашнем каталоге. Правила в этом файле называются рецептами. Задолго до того, как люди начали обмениваться фильмами и музыкой, они обменивались друг с другом рецептами *Prostmail*, которые все еще можно найти с помощью обычного поисковика. Со временем ваши рецепты вырастут и изменятся, но, взглянув на то, как другие пользуются *Prostmail*, вы сможете лучше обработать свои письма.

В зависимости от настроек дистрибутива, может потребоваться создать и файл *.forward* – так вы гарантируете, что *Prostmail* примет за обработку ваших писем до их попадания в ваш почтовый ящик. Наберите

```
/usr/bin/procmailer
```

в текстовом редакторе и сохраните файл под именем *.forward* в своем домашнем каталоге. Все входящие сообщения будут перенаправлены в *Prostmail*.

За конфигурацию большинства утилит командной строки отвечают файлы настройки, которые хранятся в каталоге */etc/*. Однако у обычных пользователей нет прав доступа или привилегий для редактирования файлов, принадлежащих *root*. Поэтому утилиты вроде *Prostmail* и *Fetchmail* позволяют пользователям создавать собственные файлы настройки в их домашних каталогах.

На нашем уроке мы воспользуемся *Prostmail* для фильтрации писем, полученных из почтового ящика Gmail. И *Fetchmail*, и Gmail поддерживают IMAP, но при загрузке сообщений из почтового ящика Gmail мы будем использовать протокол POP, и *Prostmail* сможет отфильтровать сообщения.

Получение почты

Первый шаг в настройке *Fetchmail* для взаимодействия с удаленной почтовой учетной записью – создание файла *.fetchmailrc*

» Месяц назад Мы занимались *Mutt*, почтовым клиентом командной строки.



В ПОМОЩЬ

в вашем домашнем каталоге. Откройте текстовый редактор и введите следующий текст:

```
poll pop.gmail.com with proto POP3 and options no dns
user 'usernamea@gmail.com' with password 'ваш_пароль'
is 'user' here
options keep ssl sslcertck sslcertpath '/etc/ssl/certs'
mda '/usr/bin/procmail -d %T'
```

Это все, что нужно от вас *Fetchmail*. Может также потребоваться еще включить перенаправление POP в настройках удаленной учетной записи (для Gmail и подобных сервисов).

Первые две строки – на самом деле одна; мы разбили ее для удобства. Она велит *Fetchmail* связаться с учетными записями **username@gmail.com** или **username@yourisp.com** и сбросить почту локальному пользователю. В третьей строке есть несколько опций. Первая, **keep**, не велит *Fetchmail* удалять сообщения с удаленной учетной записи. Без этой опции *Fetchmail* удалит с учетной записи все сообщения, которые загрузит. Остальные опции относятся к шифрованию SSL. Последняя строка фрагмента кода велит *Fetchmail* перенаправить все входящие сообщения *Procmail*.

Существует два способа настройки автоматического получения новых сообщений вашей учетной записи *Fetchmail*. Первый – добавить строку

```
set daemon 300
```

в файл `~/fetchmailrc`. Это заставит *Fetchmail* проверять наличие новых писем каждые пять минут (300 секунд). Установите подходящий интервал на свое усмотрение.

Второй способ – создать задание *Cron* для запуска *Fetchmail*. Запустите **crontab -e** из терминала и введите

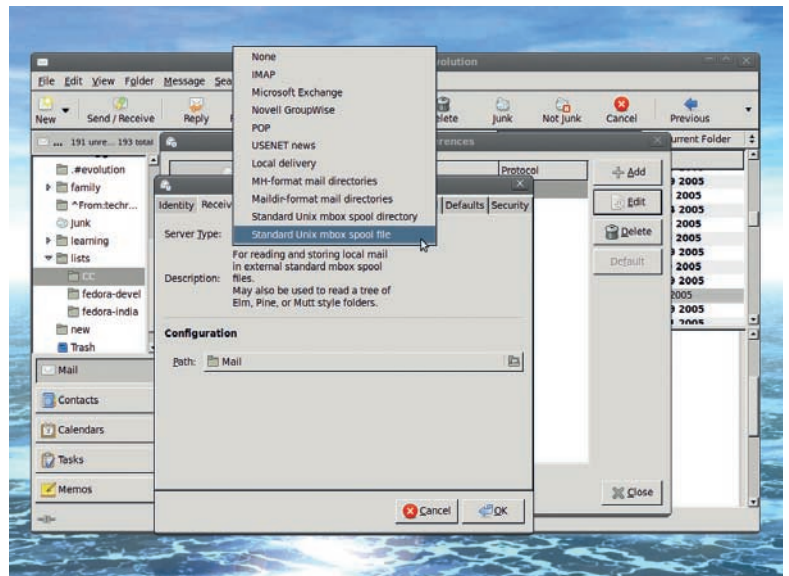
```
*/* * * * * /usr/bin/fetchmail -s
```

в файл `crontab`. *Fetchmail* будет запускаться раз в пять минут в «тихом» режиме – то есть все сообщения о ходе этого процесса и результате будут скрыты и не попадут в стандартный вывод.

Fetchmail очень строг к правам доступа к файлу `~/fetchmailrc` и требует, чтобы они были не выше 710. Поэтому измените права на 710 или меньшие командой **chmod** таким образом:

```
chmod 710 ~/fetchmailrc
```

Файл настройки `~/fetchmailrc` можно проверить командой `fetchmail -vk`



с терминала. *Fetchmail* запустится в подробном режиме (опция **-v**), а письма не будут удалены с удаленной учетной записи. Опцию **-k** (**keep**) можно не указывать, если она уже есть в файле `~/fetchmailrc`.

Следующий шаг – сообщить *Procmail* о различных рецептах фильтрации. Но прежде чем создавать правила для сортировки писем от друзей, семьи и из списков рассылки, нужно создать уникальные каталоги для каждой из этих категорий. Мы хотим, чтобы *Procmail* доставлял все сообщения в каталог `~/Mail`, поэтому создадим внутри него дополнительные директории.

```
mkdir ~/Mail
mkdir ~/Mail/new
mkdir ~/Mail/family
mkdir ~/Mail/lists
mkdir ~/Mail/lists/fedora-devel
mkdir ~/Mail/lists/debian-legal
mkdir ~/Mail/lists/gnome
```

Теперь мы готовы заполнить файл `~/procmailrc` правилами сортировки входящих сообщений. В самом базовом варианте файл `~/procmailrc` должен содержать следующую информацию:

```
SHELL=/bin/bash
PATH=/usr/sbin:/usr/bin
MAILDIR=$HOME/Mail
DEFAULT=$MAILDIR/new
LOGFILE=$HOME/procmail.log
LOG=""
VERBOSE=yes
```

Выше приведены только переменные, содержащие данные для работы *Procmail*. **MAILDIR** – рабочий каталог для *Procmail*; внутри него будет храниться вся почта. Письма, не относящиеся ни к одному из четырех рецептов, попадают в **DEFAULT**. Затем мы

» Можно настроить *Evolution* на считывание писем, отфильтрованных *Procmail*, из почтового ящика Mbox.



При запуске в «тихом» режиме (опция **-s**) *Fetchmail* все равно выводит все сообщения об ошибках.

Регексп на гарнир

Регулярные выражения [regex] – неотъемлемая часть файла `~/procmailrc`. Если использовать их со знанием дела, они существенно повышают эффективность рецептов. Следующий рецепт ищет в теле сообщения фразу и перемещает его в заданную папку.

```
:OB:
```

```
* [0-9]+ Wassup
$MAILDIR/silly
```

Найдя слово 'wassup', предваренное одной или более цифрами, в любом месте тела сообщения, *Procmail* переместит сообщение в папку **silly**.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

присваиваем имя файла журнала, в который *Procmail* будет записывать отладочную информацию, переменной **LOGFILE**. Установка **VERBOSE** в **yes** гарантирует, что *Procmail* сохранит подробную запись своих действий — это помогает в отладке своих рецептов.

Переменная **PATH** задает папку, где *Procmail* ищет все программы, нужные ему как часть рецепта. Например, вам потребуется проигрыватель, чтобы воспроизвести бравурный марш при получении того долгожданного предложения о работе.

Каждый из рецептов может содержать произвольное количество правил для выбора сообщений. Они включают отправителя или получателя, тему сообщения или даже его содержимое. Как только сообщение отобрано на основе определенных вами правил, можно заставить *Procmail* выполнить над ним определенные действия: например, переместить в заданную папку, ответить отправителю, перенаправить сообщение на другой адрес и т. д.

Анатомия рецепта

Каждый рецепт *Procmail* состоит из трех частей. Все рецепты должны начинаться с символа двоеточия (:). Следующая за этой строкой содержит условия или правила отбора сообщений, и, наконец, третья строка говорит *Procmail*, какое действие предпринять, если найдено сообщение, соответствующее правилам в рецепте.

```
:0:
* ^From:*(linuxlala@yahoo\.co\.in|geekybodhi@gmail\.com)
$MAILDIR/family
```

Этот рецепт велит *Procmail* проверить все входящие письма на соответствие условию (вторая строка) и переместить все письма, удовлетворяющие условию, в каталог **family**. Правила начинаются с символа *****. **^From** — встроенный макрос *Procmail*. **^To** и **^Subject** — похожие встроенные макросы, которые облегчают поиск соответствующих сообщений. Строка, следующая за строкой условия — строка действия. В большинстве случаев строки действия просто говорят *Procmail*, в какую папку поместить сообщения.

Заметили символ **|** в строке условия? Он необходим в тех случаях, когда в условие нужно добавить несколько проверок. Здесь мы хотим проверить на соответствие условию **^From** множество адресов. Поэтому все сообщения с адресов **linuxlala@yahoo.co.in** и **geekybodhi@gmail.com** будут доставлены в папку **family**.

Порядок рецептов в файле **~/.procmailrc** очень важен. Все входящие сообщения проверяются на соответствие приведенным рецептам, пока не обнаружится совпадение. И как только сообщение удовлетворит условию рецепта, над ним немедленно выполняются действия. Поэтому рецепт по умолчанию идет последним в списке. Так, если какое-то сообщение соответствует более чем одному рецепту, будут выполнены действия первого рецепта из файла **~/.procmailrc**, а до других рецептов дело никогда не дойдет.

В строке действия не нужно указывать полный путь до каталога, как это сделали мы. Если вы уже определили пере-

менную **\$MAILDIR**, достаточно использовать **family** вместо **\$MAILDIR/family**. Соответствующие сообщения все равно попадут в должную папку, независимо от того, указан ли полный путь или только имя каталога.

Также обратите внимание на символ **** перед **@** и **.** в адресе электронной почты, так как два этих символа — регламентированные ключевые слова.

Над письмом можно выполнить несколько проверок, чтобы гарантировать отсутствие ложных срабатываний. Следующий рецепт проверит поля **From** и **Subject** перед выполнением действий над соответствующими сообщениями.

```
:0:
* ^Subject:.*techrepublic
^From:techrepublic@direct.cnet.com
$MAILDIR/lists/techrepublic
```

Пока мы рассмотрели только простейшие рецепты, но при необходимости можно выполнять и более одной операции над сообщением. Например, если вы хотите перенаправить другу все сообщения об уроках французского и сохранить копию себе, можете воспользоваться следующим рецептом:

```
# forward French lessons to a friend
#and also keep a copy
:0
* ^Subject:.*(French)
{
:0 c
! myfriend@buddy.com
:0
learning
}
```

Здесь вместо строки действия используется вложенный блок (заклучен в фигурные скобки). В него можно поместить несколько рецептов, которые будут применяться только в том случае, если сообщение соответствует родительскому рецепту. Родительский рецепт в данном случае — проверка наличия слова 'French' в любом месте строки темы.

В каждом из рецептов блока нет ни одного условия. В первом действии для копирования сообщения используется флаг **c**. Обычно, если сообщение соответствует первому рецепту, над ним будут выполнены действия только из этого рецепта. Поэтому если не использовать этот флаг, мы не сможем применить к сообщению действия из второго рецепта. Восклицательный знак (!) перед адресом электронной почты означает, что мы хотим перенаправить сообщение на указанный адрес. Наконец, второй рецепт в блоке доставляет сообщение в каталог **learning**.

Обработка спама

Понятие спама или нежелательных сообщений может отличаться у разных людей. Мне не нравятся вложения PPT, поэтому такие письма всегда отбрасываются. Также игнорируются письма, тема которых написана заглавными буквами, согласно рецепту, приведенному ниже:

```
:0
* ^Subject: [A-Z]
/dev/null
Второй способ — создать «белый список» известных и доверенных адресатов и сохранять в своем ящике сообщения только от них:
# Check my whitelist
:0
* ? formail -x"From" -x"From:" -x"Sender:" -x"X-Sender:"
-x"Reply-To:" -x"Return-Path" -x"To:" |
```



Скорая помощь

Если для получения сообщений используется Агент получения почты, такой как *Fetchmail*, то файл **.forward** для перенаправления сообщений *Procmail* вам не требуется.

Выбор почтового ящика

При выборе формата почтового ящика есть два варианта: **mbox** и **maildir**. В примерах нашего урока мы пользуемся типом **mbox**, определенном в **~/.procmailrc** как **MAILDIR=\$HOME/Mail** и **DEFAULT=\$MAILDIR/new**. Если бы мы добавили замыкающий слэш в конец пути, как в **\$HOME/Mail/** и **\$MAILDIR/new/**, у нас получился бы тип **maildir**.

В почтовом ящике типа **mbox** все сообщения хранятся в текстовом формате в одном файле; начало сообщения отмечается строкой **From**, а конец сообщения — пустой строкой.

Формат **maildir** подразумевает хранение сообщений в отдельных файлах в каталогах **cur**, **new** и **tmp**; при необходимости они перемещаются из каталога в каталог.

» Пропустили номер? Узнайте на с. 107, как получить его прямо сейчас.


```
egrep -i -f $HOME/trusted
$MAILDIR/trusted
```

В этом рецепте мы выделяем только указанные заголовки с помощью *Formail* и передаем их утилите *egrep*, которая затем сравнивает их с адресами, перечисленными в файле *.trusted*. *Formail* – внешняя утилита, используемая для обработки сообщений и выделения информации из их заголовков; в нашем рецепте ее вывод передается *egrep*. Файл *~/trusted* содержит список всех доверенных адресов, а ключ *-i* в вызове *egrep* делает сравнение нечувствительным к регистру.

В порядке альтернативы можно поддерживать список известных нарушителей и создать рецепт, удаляющий все сообщения от них: просто замените файл *.trusted* на *.spammers* и отправьте сообщения в */dev/null* вместо *\$MAILDIR/trusted*. Но помните, что эти сообщения нельзя будет восстановить или просмотреть позже – они утратятся навсегда. Также, поскольку эти сообщения не пишутся в файл, блокировки не требуется, поэтому начните рецепт с *:0* вместо *:0:*.

Если вы хотите еще тщательнее обрабатывать спам, настройте *Procmail* для совместной работы со *SpamAssassin*; тогда перед настройкой *Procmail* нужно установить *SpamAssassin*. После его установки создайте файл */etc/procmairc*, чтобы перенаправить все входящие сообщения в *SpamAssassin*:

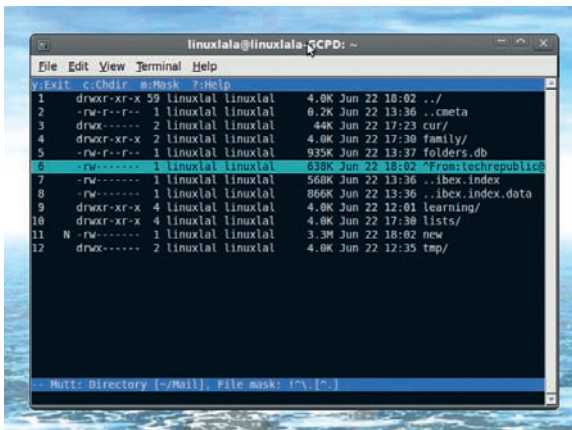
```
#Send message to SpamAssassin
:0 fw
* < 256000
| /usr/bin/spamc
```

Параметр *fw* в строке с двоеточием означает для *Procmail*, что этот рецепт является фильтром и может изменить исходное сообщение. В строке условия мы ищем сообщения, меньше 256 КБ по размеру, так как спам-сообщения обычно невелики. Это поможет нам избежать проверки всех сообщений, сэкономив немного ресурсов. В третьей строке мы передаем сообщение *SpamAssassin*.

Когда *SpamAssassin* идентифицирует сообщение как спам, он вписывает в поле с его темой специальный тэг. Итак, теперь вы можете добавить следующий рецепт в файле *~/procmairc*:

```
:0H:
* ^X-Spam-Status:.*Yes
spam
```

Можно было бы точно так же направить сообщение в */dev/null*, но лучше сохранить его в локальном каталоге: так вы сможете просмотреть его содержимое, чтобы убедиться, что ни одно сообщение не было ошибочно помечено как спам.



» При чтении почты с помощью *Mutt* можно нажать “с” для переключения между ящиками, в которые сортируются письма.

Когда блокировать файлы

Блокировка файлов – функция *Procmail*, позволяющая избежать порчи сообщений во время доставки. Представьте, что *Procmail* одновременно нужно записать два сообщения в файл. Когда это происходит, есть риск того, что два сообщения создадут путаницу в формате mailbox. Чтобы этого избежать, начинайте рецепт с *:0:*. Завершающее двоеточие гарантирует, что только одно сообщение будет записываться в файл в один момент времени.

Если рецепт перенаправляет все найденные сообщения или передает их внешнему приложению, блокировка файлов не нужна, так как *Procmail* не будет писать в них сообщения. *:0:* автоматически блокирует файл, когда *Procmail* определяет, куда пишется сообщение, и создает файл блокировки. Если файл блокировки существует, *Procmail* подождет некоторое время и снова попытается создать блокировку.

Настройка автоответчика

В *Procmail* очень часто используется функция автоответчика, известного также как ответ на время отпуска или auto-reply. В его основе – утилита *vacation*, и если она еще не установлена в вашей системе, пора это сделать.

Затем создайте в своем домашнем каталоге файл *.vacation* и введите туда сообщение – просто «Я в отпуске. Прочту ваше сообщение, когда вернусь. Удачи!» либо нечто более витиеватое. Сохраните файл *.vacation*; теперь создайте рецепт для автоответчика:

```
:0 Whc: vacation.lock
* .*for <linuxlala@gmail.com>
* !^FROM_DAEMON
* !^X-Loop: linuxlala@gmail.com
# Store sender's address in vacation.cache
| formail -rD 8192 vacation.cache
# If sender's address not in cache, send
# mail containing vacation message.
:0 ehc
| (formail -rA"Precedence: junk" \
-A"X-Loop: linuxlala@gmail.com" ; \
cat ~/vacation \
) | /usr/lib/sendmail -oi -t
```

Возможно, рецепт не совсем понятен, но если разбить его на две части, все будет очевидно. Рассмотрим первый рецепт. В нем параметр *Whc* обеспечивает три вещи. *W* ожидает возвращения из *Formail*, *h* обозначает поиск в заголовках, а *c* велит *Procmail* сохранить копию сообщения.

Далее идут три строки условия. Первая гарантирует, что в ответ на письма, присланные вам, будет высылаться ответное письмо.

```
* .*for <linuxlala@gmail.com>
```

Вторая предотвращает отправку ответа в списки рассылки.

```
* !^FROM_DAEMON
```

Третья предотвращает отправку ответа на письма, отправленные самому себе.

```
* !^X-Loop: linuxlala@gmail.com
```

В следующей строке мы велим *Formail* сохранить адрес отправителя в файле *vacation.cache*.

Следующий рецепт начинается с *:0ehc*. *e* означает, что данный рецепт выполнится только в том случае, если первый рецепт скажет, что в файле *vacation.cache* нет адреса отправителя. Затем мы велим *Sendmail* отправить сообщение *.vacation* отправителю.

Эти два рецепта гарантируют, что отправители не будут получать автоответ на каждое отправленное нам письмо: раз их адреса попали в кэш, автоответы больше не отправляются. Загляните в *man procmairc* и в *man*-страницу *Procmail* для более подробной информации об автоответчиках, затем откиньтесь на спинку кресла и расслабьтесь в жизни, свободной от спама. **LXF**

Скорая помощь

Число, следующее за : в рецепте, означает количество действий. Сейчас оно вычисляется автоматически, но для совместности все-таки нужно писать :0.

» **Через месяц** Бежим в леса! Надвигается демон FreeBSD!

ОТВЕТЫ

Есть вопрос по открытому ПО? Пишите нам по адресу answers@linuxformat.ru

В этом месяце мы ответим на вопросы про:

- 1 Загрузочный USB-брелок
- 2 Загрузку с невидимого диска
- 3 Перенос принтера
- 4 Перенос диска
- 5 Миграцию с MS
- 6 Безопасное перемещение на нетбуках
- 7 Компиляцию на нетбуках
- 8 Шифрование
- 9 Слежение web-камерой

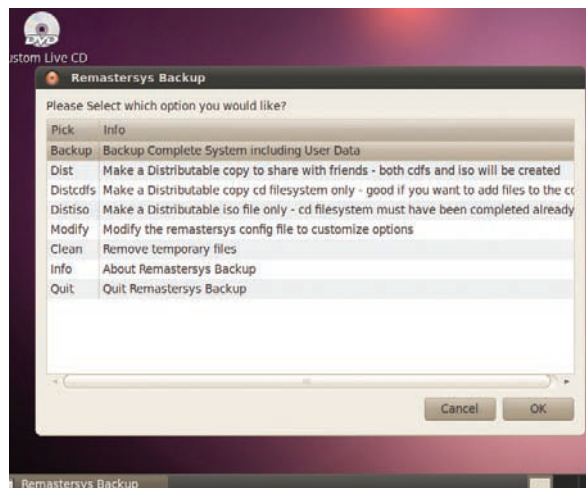
1 Личный USB-брелок

Вам надо создать загрузочные копии Linux на USB-брелках, при загрузке автоматически предлагающие пользователю зарегистрироваться в системе, для доступа к шифрованной папке на USB с отчетом в формате HTML и ассоциированными с ним файлами. Затем пользователь просматривал бы документы через интерфейс HTML. В Windows мы обычно достигали этой цели, применяя *Encryptex* для шифрования HTML-файлов на CD, а пользователь получал доступ к этим файлам, и дешифрация осуществлялась «на лету». Теперь хотелось бы перейти на Linux, загружаемый с USB, чтобы избежать доступа к локальному жесткому диску и обрабатывать большие объемы данных.

Мы довольно успешно применили стандартный дистрибутив Ubuntu и его встроенные возможности шифрования. Сейчас хотелось бы ускорить процесс загрузки, не монтируя ненужные устройства. Кроме того, нужно выдать доступ к ряду приложений, а именно *00o* и простейшим просмотрщику графики, медиа-проигрывателю (сейчас это *VLC*) и браузеру (сейчас — *Firefox*). Я перерыл свои архивы и нашел там немало полезного, но теперь мне опять нужна помощь. В частности: какие файлы нужно редактировать, чтобы упростить процедуры запуска?

Джон Шерленд [John Sharland]

Существует несколько инструментов, позволяющих собрать индивидуальную версию дистрибутива — например, *Ubuntu Customization Kit*, устанавливаемый через *Synaptic*, или *SUSE Studio*. Они позволяют создать специфически настроенный ISO-образ в режиме онлайн. В основном эти инструменты нацелены



Remastersys позволяет превратить установленную копию Ubuntu в загрузочный Live CD, DVD или USB-носитель.

на создание индивидуально настроенного инсталлятора, а не Live CD. Лучшим вариантом для Вас был бы перенос рабочей копии Linux с жесткого диска на носитель USB. В Ubuntu и Debian сделать это можно с помощью инструмента *Remastersys Backup* (<http://remastersys.sourceforge.net>) — его разработчики вдохновлялись идеями аналогичных инструментов для Mandriva и PCLinuxOS, но мы остановимся на Ubuntu, раз уж Вы с ним работали.

Первым шагом будет установка дистрибутива на жесткий диск. Можно сделать это через двойную загрузку, но установка на виртуальную машину (например, *VMWare* или *VirtualBox*) будет проще: Вы сможете пользоваться своим стандартным дистрибутивом во время сборки образа. Кроме того, можно провести тестовую загрузку новой инсталляции, а также подстраивать базовую инсталляцию, когда захочется.

С помощью *Synaptic* деинсталлируйте все, что Вам не нужно. Здесь очень пригодятся возможности «моментальных снимков» [snapshots], предлагаемые виртуальными машинами — Вы сможете запросто откатиться назад, если по ошибке удалите что-то важное. Блокируйте все ненужные приложения, стартующие при запуске системы. Поскольку Вы намерены предоставлять лишь базовый набор функций, можно «облегчить» и всю систему, заменив Gnome на более легковесную настольную среду, или даже просто запускаться с Xubuntu (*Xfce*) или Lubuntu (*LXDE*, это сейчас наша любимая облегченная настольная среда). При этом не мешает создать еще одного пользователя, потому что первый пользователь, для которого создается учетная запись, получает права *sudo*. В виртуальной среде такой

Наши эксперты

Мы найдем ответы на любой вопрос — от проблем с установкой системы или модемом до сетевого администрирования; главное — спросить!



Нейл Ботвик

Владелец ISP и экс-редактор дисков для нашего журнала, Нейл считает, что в Linux он от скуки на все руки.



Пол Хадсон

Пол — местный супер-программист, и он может и хочет управиться со всеми вашими проблемами по части web и баз данных.



Евгений Крестников

10 лет работал системным администратором. Он поможет вам разобраться с нетривиальными вопросами настройки Linux, а также со встраиваемыми системами.



Майк Сондерс

Майк был одним из создателей прототипа LXF — Linux Answers. Его специальности — программирование, оконные менеджеры, скрипты инициализации и SNES.



Грэм Моррисон

Когда Грэм не обозревает кучи программного обеспечения и не халтурит с *MythTV*, он готов к ответам насчет любого оборудования и проблем виртуализации.



Юлия Дронова

Если компьютер у Юлии не занят выполнением команды *emerge*, она спешит применить его для модернизации www.unixforum.org.

Куда посылать вопросы

Пишите нам по адресу answers@linuxformat.ru или спрашивайте на форуме: www.linuxformat.ru

пользователь нужен, но наделять столь крупными полномочиями всех, кто будет загрузиться с USB-носителя, вовсе незачем (они смогут активировать автоматическую регистрацию в системе, что сведет на нет Ваши усилия по созданию шифрованного домашнего каталога).

Скачайте Deb-файл **Remastersys** с web-сайта, а затем установите его следующей командой:

```
sudo remastersys backup
```

или активируйте из меню System > Administration. Выбирайте первую из опций – создать полную резервную копию системы, так как при этом сохраняются все настройки. Пока создается резервная копия, компьютер лучше не трогать. По завершении, протестируйте систему, загрузив виртуальную машину *VirtualBox* с ISO-образа. Убедившись, что все работает, как планировалось, скопируйте образ на USB-накопитель с помощью *Unetbootin*. **MC**

2 Грузим невидимый диск

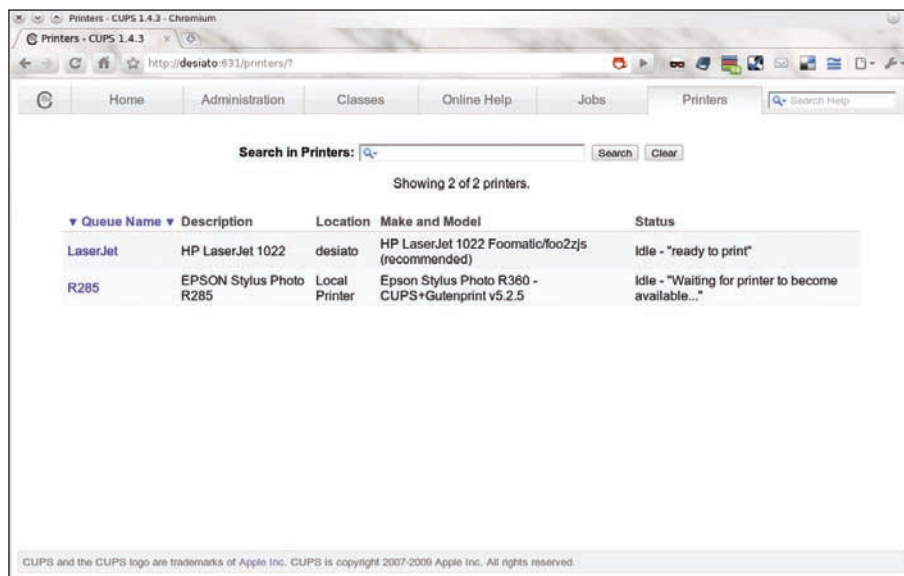
В Я недавно установил Ubuntu 9.10 с LXF DVD на настольный ПК с двумя жесткими дисками, отведя ему раздел на втором жестком диске объемом 320 ГБ, так как первый жесткий диск, на 30 ГБ, поставляемый с компьютером, был полностью забит. Установка прошла нормально, но, перезагрузившись, я получил ошибку *Grub* – `/grub/boot` не обнаружился. После долгих изысканий я выяснил, что второй диск, где был установлен *Grub*, невидим для BIOS, что и вызвало ошибку. Второй диск – это накопитель SATA, подключенный к контроллеру PCI SATA, для которого, по-моему, сперва надо загрузить драйверы, а значит, как загрузочный диск он не годится.

Нет ли способа загружать Linux со второго диска?

Дэйв Белтон [Dave Belton]

Способ есть, и он может быть либо очень простым, либо очень хлопотным, в зависимости от того, входят ли в Ваш дистрибутив драйверы для контроллера SATA. Впрочем, Ubuntu Live CD явно содержит такой драйвер, иначе инсталляция на Ваш второй диск не прошла бы. При загрузке система в первую очередь загружает первый этап *Grub* из MBR на первом диске, а на втором этапе ищет `/boot/grub` и файлы настроек. Затем через меню *Grub* загружается ядро, а оно загружает `/sbin/init` из корневого раздела дистрибутива. В Вашем случае, загрузка достигает первого этапа и завершается неудачей, потому что контроллер диска невидим для BIOS (драйверы тут ни при чем: на данном этапе еще не загружено никакой операционной системы).

Первым делом проверьте все опции своей BIOS, а также переключки на Вашем контроллере, чтобы выяснить, нельзя ли сделать его видимым для BIOS. Если это удастся, проблема будет решена. В противном случае потребуется на третьем этапе загружать ядро и драйверы, чтобы диск SATA стал видимым. Следовательно, все файлы *Grub* и ядро должны находиться на первом диске. Сделать это легко, потому что все, что Вам требуется, расположено в каталоге `/boot`. Создайте небольшой раздел ext2 на диске IDE – 10 МБ



Многие дистрибутивы имеют собственные программы установки принтеров, но можно работать с CUPS и через его web-интерфейс.

должно хватить. Затем загрузитесь с Live CD и скопируйте содержимое загрузочного каталога с Вашего большого диска на этот раздел. Отредактируйте в нем файл настройки *Grub*, позаботившись, чтобы параметр `root` в строке, задающей конфигурацию ядра, указывал на большой диск. Эта строка должна выглядеть примерно так:

```
root=/dev/sdb1
```

Поскольку файл настройки ссылается на ядро как `/boot/vmlinuz-xyz`, а у Вас путь к ядру изменился, став `/vmlinuz-xyz`, Вы можете либо вручную изменить каждое вхождение, либо создать символическую ссылку. Перейдите в каталог `/boot` и введите команду

```
sudo link -s . boot
```

Затем настройте *Grub* в MBR так, чтобы использовать ядро и файлы настройки в `/boot`, скопировать

```
sudo grub-install /dev/sda
```

Теперь перезагрузитесь и повторите попытку. При неудаче, *Grub* обычно выводит номера ошибок. Чтобы преобразовать их в осмысленные сообщения, обратитесь к документации по *Grub*. Обычная причина неудачи – некорректно указанный путь к файлам настройки или ядру. Если модуль драйвера не загружается, потребуется модифицировать скрипты инициализации или перекомпилировать ядро, встроив в него драйвер Вашей карты SATA вместо использования его в виде отдельного модуля. Здесь у нас нет места на рассмотрение этих вариантов. Загрузившись с Live CD, дайте следующую команду:

```
sudo lspci -k
```

Это простейший способ узнать, какой модуль Вам на самом деле нужен. **HB**

3 Перенос принтера

В Еще со времен Windows у меня остался принтер Canon Pixma MP500. Теперь я пользуюсь *Crunchbang* и очень доволен. Мне нравится не слишком минималистский пользовательский интерфейс. А вот что мне не нравится –

драйвер принтера не установился автоматически, и я не могу найти его онлайн. Что интересно, последние две версии Linux Mint находят драйвер на Live CD. Где в CUPS хранятся драйверы, и не могу ли я просто списать драйвер с Live CD Mint в мою установленную копию *Crunchbang*?

Патрик Бадеберг [Patrick Buddeberg]

Копирование файла драйвера из одного дистрибутива в другой – не очень хорошая идея. Драйвер может оказаться несовместимым с ПО на другом компьютере или потребовать другие файлы. Гораздо лучший подход – посмотреть, какой драйвер используется в Mint, и «убедить» *Crunchbang* использовать его. Причина, по которой *Crunchbang* не использует правильный драйвер, может заключаться либо в том, что не работает автоматическое обнаружение, что маловероятно, либо в том, что правильный драйвер не включен в стандартную инсталляцию. Тогда задача сводится к установке правильного пакета драйвера и повторном запуске конфигурации принтера.

Для упомянутого Вами принтера имеется драйвер в коллекции драйверов *Gutenprint* (ранее *Gimp-print*). В *Crunchbang* он не включен в стандартную инсталляцию, в отличие от Mint. Чтобы решить проблему, установите в *Crunchbang* пакет `foomatic-db-gutenprint`, затем повторно запустите конфигурацию принтера, выбрав опцию System > Printer Settings, и выберите драйвер `stp-bjc-MULTIPASS-MP500`, если он не выберется автоматически. Как вариант, можно поработать непосредственно с CUPS, указав в браузере следующий адрес: <http://localhost:631>. **ПХ**

4 Перенос диска

В Чувствую, что прошу о невозможном, хотя не понимаю, почему. Я хочу установить в мой компьютер новый жесткий диск объемом 160 ГБ и перенести на него все содержимое имеющегося диска объемом 80 ГБ (включая ОС). **»**

Прямое копирование кажется мне подозрительно простым вариантом. Буду благодарен за любые советы. Новый жесткий диск раньше использовался на моем другом компьютере, и я хочу просто стереть его содержимое и установить на новый компьютер.

Catgate, с форумов



У Вас есть три возможности. Первая и самая простая – просто не копировать ОС. Если в корпусе Вашего компьютера есть место для дополнительного диска, почему бы не оставить подключенными оба (если у Вас нет других планов насчет старого диска)? Тогда у Вас будет 240 ГБ дискового пространства. Весь второй диск можно отвести под один большой раздел `/home` или выбрать схему разбиения на разделы по Вашему усмотрению. На старом диске сохранится ОС, а дополнительное пространство можно использовать для экспериментов с другими дистрибутивами.

Второй вариант – скопировать содержимое старого диска на новый с помощью *Clonezilla*. Как следует из названия, *Clonezilla* выполнит клонирование (побайтное копирование) содержимого старого жесткого диска на первую половину нового диска. По завершении копирования Вы можете воспользоваться *GParted* для изменения размеров существующих разделов или создания новых, чтобы эффективно использовать дополнительное пространство. Преимущество использования *Clonezilla* в том, что будет скопирован и загрузчик. Загрузитесь с CD, запустите *Clonezilla* и выберите опцию Device-Device, а затем – опцию Disk To Local Disk из меню Beginners. Выберите исходный и целевой диски и предоставьте *Clonezilla* всю прочую работу.

Третий вариант – самый трудоемкий, зато предоставляет наиболее гибкие возможности. Разбейте новый диск на разделы с помощью *GParted* или аналогичного средства. Выберите любую схему разбиения на разделы (разумный минимум – схема с разделами `swap` и `/home`). Затем понадобится вручную скопировать содержимое каждого раздела. Хорошим выбором для этой цели будет *rsync*. Лучше всего, прежде чем выполнять эту задачу, загрузитесь с Live CD, потому что копирование системных файлов при работающей ОС чревато подводными камнями. Загрузившись, смонтируйте Ваши разделы и выполните копирование, командами

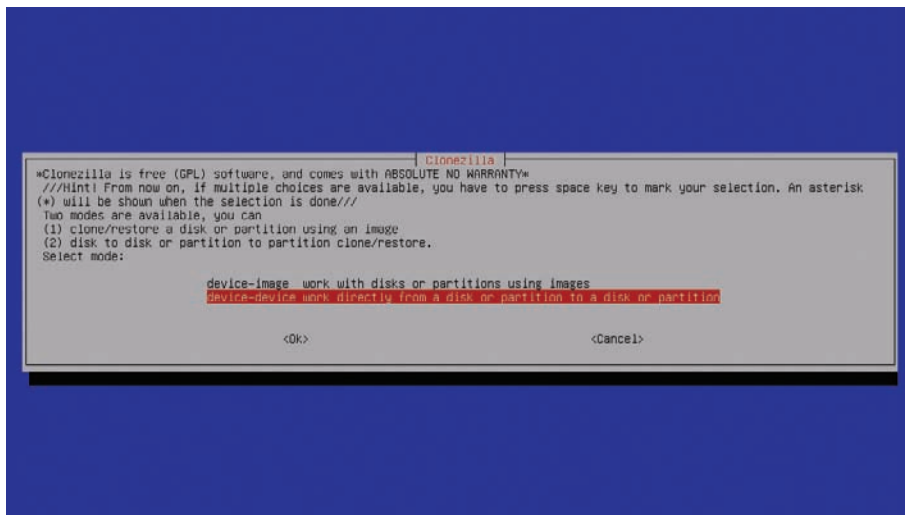
```
rsync -ax /mnt/oldroot/ /mnt/newsroom
```

```
rsync -ax /mnt/oldhome/ /mnt/newhome
```

К ним можно добавить опцию `-v`, чтобы следить за ходом копирования. Отображение прогресса иногда создает иллюзию скорости, хотя копирование на самом деле идет медленнее. По завершении копирования отключите питание, снимите 80-гигабайтный диск и снова загрузитесь с Live CD. Откройте терминал и дайте следующую команду:

```
sudo grub-install /dev/sda
```

Она установит загрузчик на новый и ныне единственный жесткий диск, и Вы будете загружаться уже с него. **ГМ**



Clonezilla можно применять для резервного копирования жесткого диска в файл образа или для прямого переноса на другой диск.

5 На путь истинный

У меня есть ноутбук с Windows XP Professional, я использую его для работы – в основном с *MS Office* и кое-какими другими приложениями. Хочу приобрести новый ноутбук, по возможности, без Windows и Office. Могу я в такой ситуации перейти на Linux? Большинство моих клиентов работают с Windows и приложениями Office, и мне придется конвертировать мои документы либо в формат PDF, либо в один из форматов MS, причем без потери функциональности.

Teo Povaan [Theo Rowaan]



Короткий ответ – «да». А вот подробный зависит от того, какие приложения Windows Вам требуются. Если это стандартные приложения Office, то для них существуют Linux-аналоги: *OpenOffice.org* вместо *MS Office*, *GIMP* вместо *Photoshop*, и т.д. *OpenOffice.org* может сохранять документы в различных форматах Microsoft и экспортировать их в PDF, так что с этим проблем не будет. Работа с Интернетом в Linux ничуть не хуже, чем в Windows: и *Firefox*, и *Chromium* – отличные web-браузеры, доступные на обеих платформах (как и *OpenOffice.org*). Кстати, Вам стоит начать пользоваться одной из этих программ на платформе Windows и убедиться, что они действительно Вам подходят, а уж потом переключаться на Linux.

Большинство приложений Windows имеют Linux-эквиваленты. Таблица соответствий между популярными программами Windows и их аналогами в мире Linux имеется на сайте http://wiki.linuxquestions.org/wiki/Linux_software_equivalent_to_Windows_software. Если Вы пользуетесь каким-нибудь узкоспециализированным ПО для Windows, подходящего эквивалента может и не найтись. Тогда у Вас будет несколько вариантов выбора. Во-первых, можно организовать систему с двойной загрузкой. То есть дистрибутив Linux и Ваша установленная копия Windows будут сосуществовать на одном компьютере, и при его включении Вы будете выбирать, какую систему загрузить. Инсталляторы всех стандарт-

ных дистрибутивов распознают имеющуюся установку Windows, позволяют переразбить жесткий диск так, чтобы выделить пространство для обеих операционных систем, и создают загрузочное меню. Это работает хорошо, и единственный минус – необходимость перезагружать компьютер для переключения с одной ОС на другую. Если Вам требуется запускать приложения Windows лишь изредка, тут может подойти *Wine* – среда совместимости, поставляемая со многими дистрибутивами и позволяющая запускать из-под Linux некоторые приложения Windows (точнее, приличное их количество). *CrossOver Office* (<http://www.codeweavers.com/products/cxlinux>) – платный, но недорогой вариант *Wine*, содержащий список программ, которые гарантированно должны работать и поддерживаются.

Как вариант, можно использовать *VirtualBox* (этот пакет тоже включен в большинство дистрибутивов) для запуска виртуальных машин Windows с рабочего стола Linux. Этот эмулятор менее удобен, чем *Wine*, но поскольку в виртуальной машине запускается полная версия Windows, Вы сможете запускать почти любое ПО для Windows, какое только Вы установите в виртуальной среде. Обычная практика – использование комбинации этих подходов, причем потребность в системе с двойной загрузкой снижается по мере накопления опыта работы в Linux. Просто помните, что Linux – это не Windows, и многие задачи выполняются здесь по-другому (а иначе в чем смысл перехода с одной системы на другую?). Естественно, на освоение новой для себя операционной системы требуется время. **ГМ**

6 Безопасный переезд

Я работаю с Ubuntu 10.04 и хочу переместить каталог `/home` на новый раздел. У меня для этого все настроено и готово, но мой домашний каталог зашифрован, так что копирование файлов с помощью *rsync* не работает. Нет ли простого способа решить мою задачу? В идеале, я хотел бы, чтобы домашний каталог был зашифрован и на новом разделе.

Можно ли скопировать с помощью *rsync* все, кроме папок Private и *ecryptfs*, а затем снова установить шифрование, или есть способ переместить все содержимое целиком?

Жаумас, с форумов

Для шифрования своего домашнего каталога Ubuntu использует *ecryptfs*. При этом шифрование на уровне файловой системы не используется. Вместо этого Ubuntu шифрует отдельные файлы (и их имена) внутри каталога, и Вы видите обычный каталог, который выглядит как заполненный мусором. Затем используется *ecryptfs* для подмонтирования зашифрованного каталога к другому, в данном случае, */home/user*, и все выглядит нормально.

Вы не хотите копировать содержимое Вашего домашнего каталога, а только зашифрованные файлы, но они скрыты, когда примонтирован каталог *home*, и Вы не можете его отмонтировать, пока зарегистрированы в системе. Это серьезно усложняет Вашу задачу в Ubuntu, где нельзя просто зарегистрироваться как суперпользователь-root. Простейший вариант – загрузка с Live CD. Примонтируйте исходный корневой и новый домашний разделы и скопируйте содержимое Вашего раздела */home* при посредстве *rsync*:

```
rsync
--archive /mnt/oldpartition/home /mnt/newpartition/
```

Сделав это и отредактировав */etc/fstab*, чтобы примонтировать новый домашний каталог, перезагрузитесь с жесткого диска и убедитесь, что Вы используете новый раздел *home* и все Ваши файлы присутствуют на местах и корректны. Затем снова перезагрузитесь с Live CD и удалите содержимое исходного каталога *home* — но не сам каталог, иначе Вы не сможете примонтировать новый раздел. **НБ**

7 Встроенный компилятор

В Linux я новичок, и у меня бывают проблемы с загрузкой программ типа *Tar* на мой нетбук Ubisurfer BusyBox. Я не могу заставить их работать из-за явного отсутствия компилятора C. По этой причине у меня до сих пор не работают принтер, звуковая карта, CD-ROM и т.д.

В Web предлагается много различных (мини-) компиляторов C, но похоже, что и их заставить работать тоже нелегко. То же самое относится и к *uClibc*, предлагаемому BusyBox — я не могу использовать *Git*. Не могли бы вы мне помочь?

Джон Эллиот (John Elliott)

Малые устройства вроде Вашего и вправду не слишком подходят для компиляции. Дело здесь не только в маломощном процессоре, но и в нехватке объема ОЗУ и дискового пространства. Обычно компиляторы требуют немало того и другого. Как дополнительную дисковую память, можно бы использовать флэш-карту SD, но компилятор будет работать раздражающе медленно. Даже если Вам и удастся заставить его заработать, все равно останутся проблемы с компиляцией пакетов, потому что во встроенных системах нет ни библиотечных заголовочных файлов, нужных для компиляции устанавливаемого ПО, ни места для их хранения. Если Вы упорно хотите компилировать и собирать ПО на таком оборудовании, то Ваш путь будет тернист.

Лучший способ компиляции ПО для таких систем – это кросс-компиляция на другом компьютере. Архитектура таких компьютеров не обязательно должна совпадать: *GCC* легко справляется с кросс-компиляцией – сборкой ПО, предназначенного для одной архитектуры, на компьютере с другой архитектурой. Впрочем, настройка кросс-компиляции тоже процесс трудоемкий, хотя

скрипт *crosstool*, доступный по адресу <http://www.kegel.com/crosstool>, и упрощает этот процесс.

Что касается звуковой карты, то она должна бы работать сходу. Возможно, не следует ожидать, что с нетбуком сработаются все устройства, подключаемые извне (например, принтеры); но встроенные должны работать уже при включении нетбука, или Вы приобрели дефектный продукт. Печатайте лучше всего через сеть, а не подключать принтер непосредственно к Ubisurfer. Тогда надо заботиться только об универсальной поддержке печати, потому что все остальное, включая драйверы конкретного принтера, управляется компьютером, к которому подключен принтер. Если у Вас есть работающий принтер, применяющий CUPS на настольной Linux-системе, отредактируйте */etc/cups/cupsd.conf* и убедитесь, что там содержится такие строки:

```
Listen *:631
Allow @LOCAL
Browsing On
BrowseAllow @LOCAL
```

Они велят CUPS прослушивать все сетевые интерфейсы и разрешать установку соединения со всех компьютеров локальной сети, так что Вы должны иметь возможность печати через Вашу беспроводную сеть (но не через GPRS-соединение).

Если с Вашим нетбуком работают флэш-накопители USB, Вы сможете без проблем подключить к Вашему нетбуку через USB внешний привод CD или DVD. Впрочем, удобнее писать аудио- и видеотреки на карту SD – применение внешнего привода лишает нетбук преимущества компактности. **ПХ**

8 Проверка шифрования

Устанавливая Ubuntu 10.04, я выбрал опцию шифрования домашней папки. Теперь хотелось бы найти какой-нибудь



Часто задаваемые вопросы

WPA

» Что такое WPA?

WPA – это более защищенный алгоритм шифрования, замена WEP.

» Спасибо, что ответили коротко – но не совсем ясно.

WPA (Wi-Fi Protected Access) – это система защиты данных при их передаче по беспроводным сетям.

» А оно мне надо?

Если вы живете не в пещере или экранированном бункере – почти наверняка да. В отличие от традиционной проводной сети, где ваши данные не покидают безопасных пределов проводов, подключенных к компьютеру, беспроводные сети передачи данных распространяются на сотни метров во всех направлениях. А значит, любой обладатель ноутбука и беспроводной карты может прочитать данные с той же легкостью,

как если бы он был подключен к вашему сетевому концентратору.

» Дом у меня большой, и мой ноутбук даже не видит сети во всех комнатах – разве это не значит, что он в безопасности?

Нет! Ваш ноутбук, может, и не в силах подключиться к вашей точке доступа на некотором расстоянии, но другим аппаратным средствам это по зубам. Добавив антенну к своей беспроводной карте, вы получите гораздо больший радиус приема сигнала. Правильно доработанная система может дать огромный прирост зоны охвата. Если ваша компания использует беспроводную сеть со слабой безопасностью, неужели вы хотите, чтобы кто-то читал ваши личные файлы из автомобиля на парковке рядом с офисом?

» Вы тут поминали какой-то WEB...

WEP – сокращение от Wired Equivalent Privacy (конфиденциальность на уров-

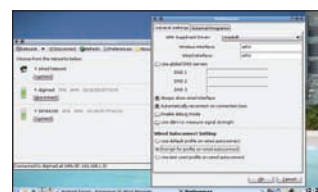
не проводных сетей). Это была предыдущая попытка обезопасить беспроводную сеть. Хотя это гораздо лучше, чем полное отсутствие защиты, взлом не особо сложен. Так можно пресечь случайное попадание вашего соседа в вашу точку доступа (это случается – вот так люди и узнают, что соседи используют незащищенную сеть), но хакеру не воспрепятствует.

» Нужно ли для WPA специальное оборудование?

Все новые беспроводные сетевые карты и точки доступа поддерживают и WPA, и WEP. Если у вас старое устройство, WPA может не быть.

» А как насчет ПО?

Вам нужна поддержка WPA на двух уровнях: в драйверах устройств, которая обычно имеется, и в различных программах, реализующих WPA. Самая популярная программа для этого – *wpa_supplicant*, обычно она идет с вашим дистрибутивом, а также доступна на <http://hostap.epitest>.



» *Network Manager* – дело хорошее, но *Wicd* тоже справляется с управлением беспроводными соединениями.

[fi/wpa_supplicant](http://wpa_supplicant). Она выполняет согласование ключей и аутентификацию.

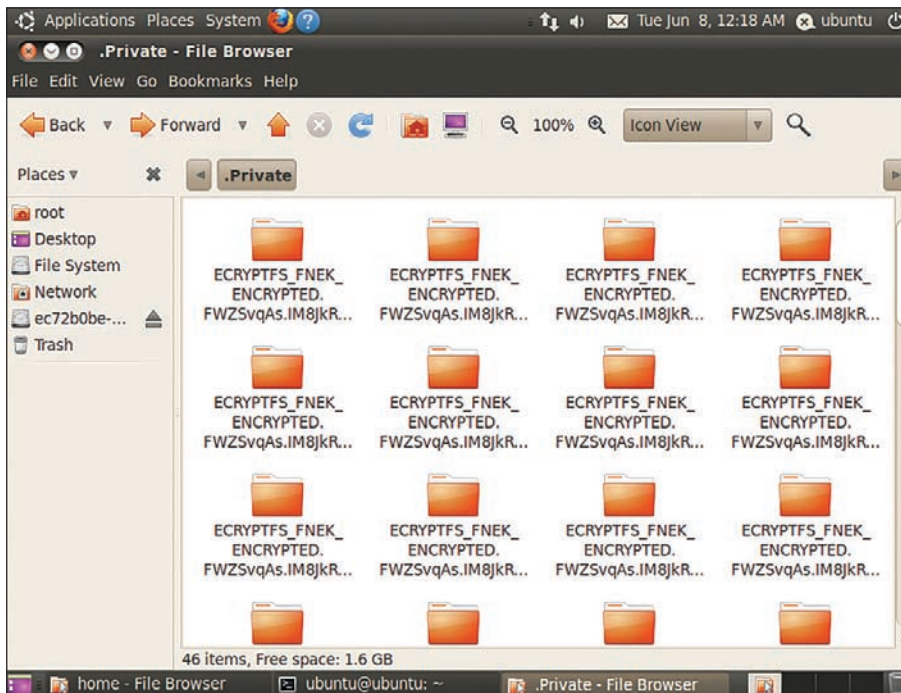
» Подозреваю, что для использования WPA в Linux придется читать ман-страницы и редактировать файлы настройки? На здоровье, если вы энтузиаст *Vim*. К счастью, есть графические утилиты конфигурации, облегчающие жизнь. Известнейшая из них – *NetworkManager* для Gnome (www.gnome.org/projects/NetworkManager); она имеет интерфейсы и для KDE и других рабочих столов.

способ, позволяющий убедиться, что данные действительно зашифрованы...

Ричард Филлипс (Richard Phillips)

Ubuntu использует файловую систему `ecryptfs` для шифрования содержимого отдельного каталога и дает возможность просмотра незашифрованного содержимого каталога, монтируя его в другой точке. При шифровании домашних каталогов в Ubuntu, зашифрованные данные содержатся в папке `/home/user/.Private`, которая при Вашем входе в систему монтируется к папке `/home/user`. Учтите, что Вы не должны активировать автоматическую регистрацию, не то Ваши зашифрованные данные будут доступны любому, кто ни включит Ваш компьютер.

Для проверки, что данные действительно зашифрованы, загрузитесь с Live CD и просмотрите содержимое Вашего `/home` – того, что расположен на жестком диске, а не виртуального домашнего каталога в среде Live CD. Перейдите в меню `Places` и выберите корневой каталог своей установки Ubuntu. Просмотрите содержимое каталога `/home` – Вы увидите каталог своего пользователя, в составе которого будет пара текстовых файлов и каталог `.ecryptfs/user`; хотя для того, чтобы увидеть этот каталог, понадобится включить опции `View > Show Hidden Files`. Там должны содержаться все Ваши файлы, но и имена, и их содержимое должны быть зашифрованы.



➤ Это зашифрованные домашние папки в Ubuntu в заблокированном виде — даже спецслужбам нелегко будет добраться до смысла.

Чтобы просмотреть эти файлы, потребуется либо зарегистрироваться в Ubuntu, введя свои имя пользователя и пароль (вот почему

не надо допускать автоматическую регистрацию в системе), или примонтировать каталог в другой среде, используя парольную фразу [passphrase] к этому каталогу. Парольная фраза – это случайным образом сгенерированная строка, которая была выведена для Вас при первой регистрации в системе. Следовало ее записать и сохранить в безопасном месте. Но не горюйте, если Вы этого не сделали. Загрузите Ubuntu, зарегистрируйтесь в системе, откройте терминал и скопируйте команду `ecryptfs-unwrap-passphrase`

Вам предложат ввести пароль, с которым Вы входите в систему, а в ответ на него эта команда выведет Вам парольную фразу. Чтобы примонтировать каталог с помощью парольной фразы, дайте из Ubuntu следующую команду:

```
ecryptfs-mount-private
В другом дистрибутиве Live CD, введите
mount -t ecryptfs -o
key=passphrase:passphrase_passwd=парольная_фраза /home/user/.Private /home/user
```

Учтите, что этот метод небезопасен, потому что пока файловая система примонтирована, парольная фраза будет видима любому зарегистрированному пользователю, который даст команду `ps`. Более защищенный вариант выглядит так:

```
mount -t ecryptfs -o
key=passphrase:passphrase_passwd_file=/mnt/usbstick/somefile /home/user/.Private /home/user
```

Здесь файл `mnt/usbstick/somefile` содержит следующий текст:

```
passphrase_passwd=парольная_фраза
```

Он может весьма пригодиться, если из-за несчастного случая Ваша система Ubuntu откажется загружаться и Вам потребуется извлечь свои файлы, поэтому сразу же запишите парольную фразу и сохраните ее в безопасном месте. **FM**



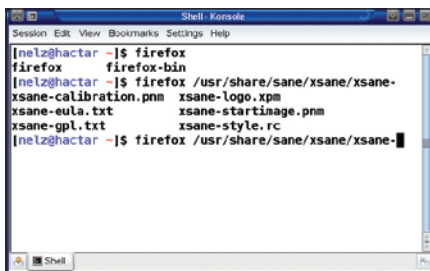
Коротко про...

Автодополнение

У линуксоидов есть масса причин избегать командной строки. Во-первых, это сложность запоминания команд. Вдобавок при вводе длинных команд и имен файлов кажется, что так выйдет медленнее, чем щелкать мышью в графическом интерфейсе пользователя. Наконец, если вы плохой наборщик, всегда есть риск, что введенная команда не будет работать из-за опечаток.

Хорошая (или плохая – как взглянуть) новость заключается в том, что все эти причины в общем неважны. В командной оболочке имеется исключительно дружелюбная к пользователям функция – автодополнение ввода команд по нажатию клавиши `Tab`. Что же это такое? Проще всего объяснить на примере. Пусть вы собрались читать файл `/usr/share/sane/xsane/doc/sane-xsane-fax-doc.html` (у нас в компьютере такой и правда есть). Войдите в оболочку и наберите `fire[TAB]/us[TAB]sha[TAB]sa[TAB]x[TAB]d[TAB]sa[TAB]x[TAB]j[TAB]`

Первое нажатие клавиши `TAB` ищет путь к подходящей команде (`firefox`). Следующие нажатия вызовут попытки автоматически завершить имя файла, найдя для него совпадение в текущем каталоге. Это не только намного уско-



➤ Автодополнение по клавише `Tab` ускоряет ввод длинных команд и имен файлов.

рывает ввод с клавиатуры, но и исключает опечатки: ведь дополнение идет только до существующих имен файлов.

Если совпадут начала нескольких команд или имен файлов, оболочка дополнит имя по максимуму возможного, а следующее нажатие `Tab` покажет список вариантов. После этого вводите еще один-два символа, а затем снова жмите `Tab`. Аналогично можно просматривать доступные команды. Оболочки вроде `Bash` или `Zsh` расширяемы так, что автодополнение применяется и в других целях: например, для просмотра имен хостов или аргументов программы. Но тут уж потребуется чтение нового набора `man`-страниц.

БОЛЬШОЙ ВОПРОС Как удаленно просмотреть материал с web-камеры?

9 Управление камерой

В Я бы хотел с помощью web-камеры приглядывать за детьми по вечерам, когда они уложены спать. Я планирую оставлять в их спальне нетбук и выводить отснятое web-камерой на мой настольный ПК. Моя web-камера неплохо ладит с Linux, но как заставить ее вывести изображение на другом компьютере, работающем под Ubuntu 9.10?

Найджел Гриффитс (Nigel Griffiths)

О Вам нужна программа *webcam-server*: ее имя говорит само за себя. В ней два компонента: сервер, передающий вывод web-камеры через сеть, и апплет Java, который можно встроить в web-страницу для отображения поступающей на него через сеть информации в реальном времени. Итак, придется запустить web-сервер – например, *Apache*. Но зато Вы сможете открыть доступ извне Вашей домашней сети, с необходимыми предосторожностями, и через браузер просматривать вывод web-камеры откуда угодно.

Первый шаг – установка программы *webcam-server* из *Synaptic*, а затем ее запуск из сеан-

са терминала. Если Ваша web-камера – устройство */dev/video*, параметры не нужны – хватит имени; правда, программа не сможет вести запись в файл журнала. Эта проблема решается применением опции *-l* для ведения журнала в вашем домашнем каталоге. Если web-камера «подвешена» на другое устройство, воспользуйтесь опцией *-d*. А опция *-g* задает разрешение картинки – по умолчанию, 320×240. Вот пример запуска программы *webcam-server* с опциями:

```
webcam-server -d /dev/video1 -l ~/webcamserver.log -g 640x480
```

Теперь запустите на том же компьютере браузер и введите адрес <http://localhost:8888>; Вы должны увидеть статическое изображение. Обновите его, нажав на кнопку Reload. Это показывает, что сервер работает. Если Вы получаете сообщения об ошибках *get_cam_image()* при загрузке изображения в браузер – это потому, что *webcam-server* требует библиотеки совместимости *Video4Linux*, а программа не может ее найти. Поищите библиотеку командой

```
locate v4lcompat.so
```

Затем укажите путь к библиотеке из командной строки:

```
LD_PRELOAD="/usr/lib/libv4l/v4l1compat.so" webcam-server -other options
```

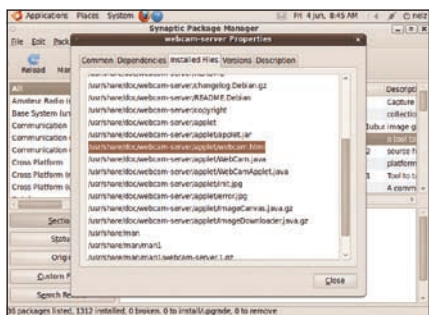
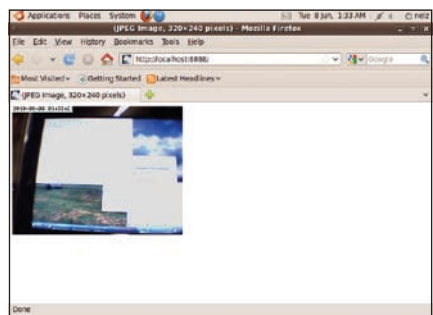
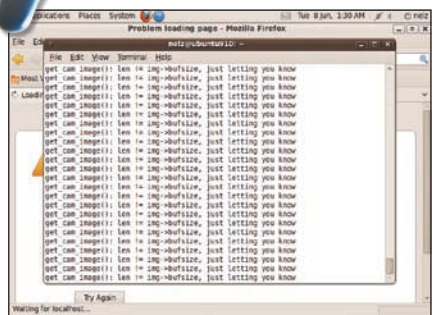
Чтобы не набирать все это каждый раз заново, запишите команду в скрипт оболочки:

```
#!/bin/sh
LD_PRELOAD="/usr/lib/libv4l/v4l1compat.so" webcam-server -d /dev/video1 -l ~/webcam-server.log -g 640x480
```

Apache обслуживает файлы из каталога */var/www*, поэтому скопируйте все файлы из */usr/share/doc/webcam-server/applet* туда, а в web-браузере укажите адрес <http://localhost/webcam.html>, чтобы увидеть изображение, поступающее с web-камеры и сменяемое раз в секунду. В файле *webcam.html* можно изменить частоту смены кадров и размеры изображения. Но учтите, что нужно будет изменять размер изображения в двух местах этого файла. Чтобы просматривать апплет с другого компьютера, подключенного к сети, измените настройку URL, указав имя хоста или его IP-адрес вместо строки *localhost*.

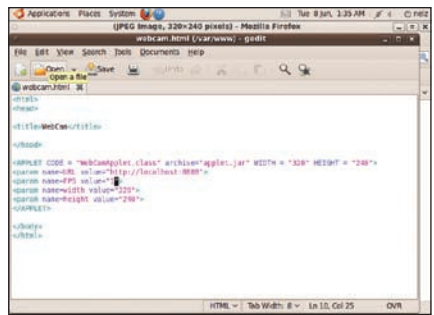
А можно и обеспечить доступ к web-серверу извне – для этого следует настроить аутентификацию по паролю. Соответствующая информация имеется в документации по *Apache*. **НБ**

Шаг за шагом: Настраиваем сервер webcam



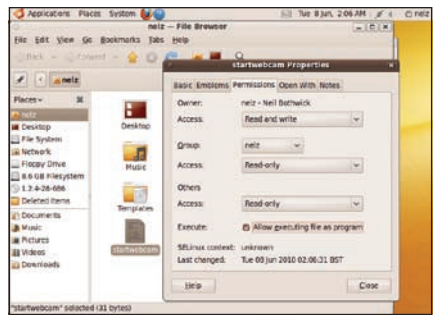
1 Заготовьте библиотеку

Если вы получаете ошибки *get_cam_image*, пытаясь посмотреть изображение с web-камеры, значит, надо сперва загрузить библиотеку совместимости *V4L1*.



2 Быстрая проверка

Чтобы увидеть изображение, поступающее с web-камеры, укажите в браузере адрес <http://localhost:8888>. Статическая картинка – это тест для сервера.



3 Скопируйте апплет

Окно свойств в *Synaptic* показывает, где установлен апплет. Скопируйте его в свой каталог web-сервера *Apache*. Обычно это */var/www*.



4 Настройте HTML

Чтобы изменить размер изображения или скорость воспроизведения, измените настройки в файле *webcam.html*. Туда же можно добавить поясняющий текст.

5 Добавьте скрипт

Поместите в скрипт команду со всеми опциями и сделайте скрипт исполняемым – вы сможете запускать сервер, дважды щелкнув мышью по значку.

6 Следите извне

Чтобы просматривать вывод web-камеры извне, перенаправьте порт 80 на компьютер, где работает сервер. Не забудьте установить систему контроля доступа. **ЛХФ**

LXF HotPicks



Ник Вейч

Ник компилирует HotPicks, используя текстовый редактор и головной мозг. А иногда — только текстовый редактор.

Eagle Mode » Rakarrack » jGnash » Knitter » Dfeta » Scribus » VirtualBox OSE
» Widelands » SDL Sopwith » Pybag » BleachBit

Среда рабочего стола/файловый менеджер

Eagle Mode

Версия 0.78.0 Сайт <http://eaglemode.sourceforge.net>

По-вашему, современная жизнь пресновата? Слишком все просто: знай щелкай по иконкам на рабочем столе, верно? А ведь вам только и нужно, что перейти в мир масштабируемой векторно-ориентированной среды, где все действия — сплошное увеличение или уменьшение масштаба да неистовая прокрутка панорамы.

Итак, *Eagle Mode* [англ. «режим орла»], выходи! Благодаря радикальной переладке рабочего пространства, все охвачено орлиным глазом (это я серьезно: достаточно сильно отдалив экран по умолчанию, вы его там увидите). Показаны панель инструментов и все файловое пространство, и для получения нужного уровня детализации надо лишь изменить

масштаб. Для навигации по масштабируемому изображению применяются и мышь, и клавиши курсора; вам, возможно, придется несколько часов попрактиковаться в управлении, чтобы освоить переходы с одной части файловой системы на другую. К счастью, на сайте достаточно документации, снабженной диаграммами, и она поможет вам в этом разобраться.

Вот основные действия: масштаб изображения увеличивается или уменьшается колесиком мыши; для прокрутки панорамы изображения нажмите и удерживайте кнопку скроллинга; клавиши курсора используются для смены объекта в фокусе.

Вам вовсе не придется отказываться от привычного комфорта вашей среды рабочего стола. *Eagle Mode* работает



» Орел славится остротой зрения, но людям придется увеличить масштаб, чтобы прочесть эти файлы.

и в Gnome, и в KDE, но если вы хотите насладиться всеми его безграничными возможностями, отдайте ему все экранное пространство. Попробуйте увеличить масштаб директории с файлами, и вы увидите, как они постепенно превращаются в предпросмотр собственного содержимого в совместимых типах мультимедиа. В *Eagle Mode* не нужен просмотрщик картинок: можно просто масштабировать файл изображения до требуемого размера.

Другие уловки

Однако это не просто файловый менеджер. Здесь имеются также и масштабируемые приложения: например, симпатичная программка для просмотра фракталов Мандельброта, программа для игры в шахматы и часы. Увеличьте масштаб глобуса на циферблате часов — и вы увидите местное время.

Автор не тешился иллюзиями, что все ринутся использовать его подход к масштабируемому интерфейсу пользователя (zoomable user interface, ZUI), но предыдущие релизы уже набрали приличную пользовательскую базу, так что его концепция не лишена будущего.

В Ubuntu имеются двоичные пакеты *Eagle Mode*, в Fedora и Suse — RPM, в Slax — LZM, а в Gentoo и Sabayon — ebuild. Есть также и живой ISO-образ, который можно скачать, так что вам, видимо, не придется собирать его из исходника.

Исследуем интерфейс Eagle Mode

Основные инструменты

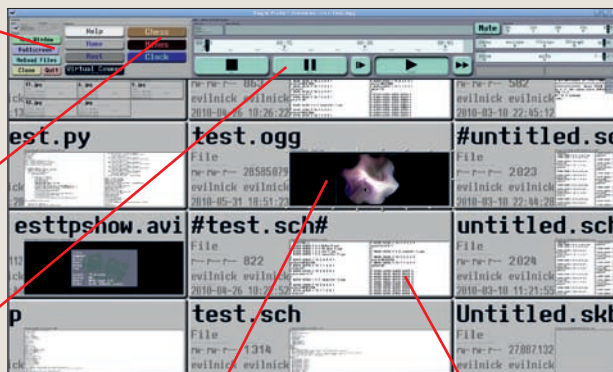
Инструменты основного меню располагаются здесь. Чтобы получить более подробную информацию, увеличьте масштаб.

Быстрый доступ

Клавиши быстрого запуска для некоторых масштабируемых приложений.

Контекстная зависимость

Здесь появляется контекстно-зависимый инструмент. В данном случае — для воспроизведения выделенного медиа-файла.



Здесь и сейчас

При желании можно воспроизводить медиа-файлы прямо на месте. Этот OGG при воспроизведении демонстрирует визуализацию.

Прочти меня

Для текстовых файлов не нужна программа просмотра. Просто увеличивайте масштаб, пока не сможете начать чтение.

Гитарные эффекты

Rakarrack

Версия 0.5.8 Сайт <http://rakarrack.sourceforge.net>

Возможно, вы думаете, что «ва-ва» — это звук, вырывающийся у Майка, когда он прищипит пальцы ящиком, но вы ошибаетесь (на самом деле он пищит «И-ии», а иногда просто всхлипывает и обещает в другой раз быть аккуратнее). Тогда вам можно и пропустить этот важный фрагмент HotPicks, потому что речь в нем пойдет исключительно о гитарных эффектах.

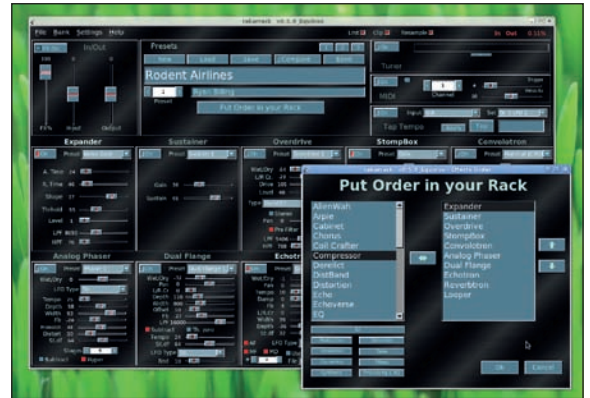
Раньше они зависели от педали и всякой дорогой электроники с Дальнего Востока, но наш век возвестил эру, когда они с тем же успехом достигаются и на компьютере (возможно, тоже восточной сборки). Rakarrack существует в Linux уже несколько лет. Используя отличную звуковую архитектуру Jack, он завоевал свою аудиторию, благодаря качеству звука на выходе и удивительной простоте в использовании. Все кнопки и слайдеры для встроенных эффектов выложены на основную панель, и вы можете настраивать их, пока не полу-

чите идеальный для вас звук (тут и гитара-то не нужна). Однако надо немного разбираться в Jack и настроить систему Jack еще до попыток извлечь хотя бы один звук из Rakarrack. Если возникнут проблемы, на сайте имеется хорошая документация.

Новые эффекты

Этот релиз включает в себя множество новых эффектов и улучшений, в том числе двойной флэнжер, конволотрон, перетасовку и вокодер, изменения в формате библиотек, увеличение частоты дискретизации, упрощенное соединение с Jack и множество огибающих. Новый эхотрон ничуть не хуже настоящей панели для реализации функции множественной за-

«Этот релиз включает множество эффектов и улучшений.»



▶ Забудьте об эмоциях — а ну-ка еще разок безумную реверберацию! Ваш выход, Rakarrack...

держки [multi-tap delay] — потягайтесь с U2, воспроизводя *Where the Streets Have No Name*. Возможно, многие читатели имеют свои понятия о применении эффектов, но, честно говоря, с ними и просто так повозиться очень круто.

Rakarrack уже имеется в виде пакета для нескольких дистрибутивов, но по тем или иным причинам эти пакеты несколько устарели. Если вы хотите собрать Rakarrack сами, понадобятся пакеты разработки для FLTK, Jack, *libsndfile* и *libsamplerate* — с ними все должно пройти гладко.

Персональный финансовый инструмент

jGnash

Версия 2.35 Сайт <http://jgnash.sourceforge.net>

Отслеживание своих финансов стало крайне важным в последнее время, и всегда приятно встретить программу, готовую взвалить на себя часть тяжелой работы. Это самодостаточное приложение создаст базу данных и автоматически выстроит все дерево, полное всевозможных счетов, абсолютно готовое и ожидающее только заполнения вами. Информацию можно вводить с клавиатуры или, что весьма удобно, импортируя эти удобные файлы *.qif*, *.oxf* или *.csv*, которые вы, возможно, скачали в своем банке. Они тщательно и аккуратно заполняют любой счет, с которым вы их соедините. И вы много осознаете, насколько вы обнищали.

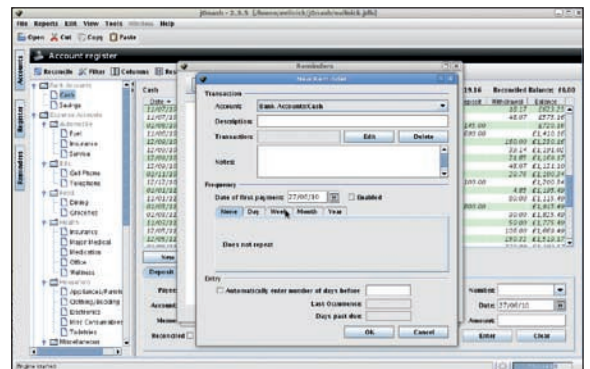
Помимо стандартных текущих счетов и сберегательных депозитов, можно вводить информацию по ценным бумагам и инвестиционным счетам. Путешественники по разным странам могут даже настроить счета с разными валютами. Функции отчетности просто отменные, так

что вы можете построить секторную диаграмму-«торт», чтобы наглядно отобразить, сколько вы потратили на торты, а потом экспортировать подробности вашего экспорта. Здесь даже имеется инструмент, чтобы распечатывать чеки.

Возможно, *jGnash* копает чересчур глубоко — глубже, чем обычно требуется; но он удивительно быстр, приятен на вид и очень несложен в использовании. Это кросс-платформенное приложение Java, но настройки поддерживают разные виды тем, и можно настроить его так, чтобы он выглядел как дома на рабочем столе Gnome, если это вас порадует.

Мы не столь часто описываем Java-приложения в HotPicks, так что кое-кто

«jGnash быстр, приятен на вид и несложен в использовании.»



▶ Настройте оповещение в jGnash, чтобы следить за своими финансами. Тогда уж вы не забудете, сколько у вас на счете.

из вас, вероятно, не слишком знаком с их работой. Это, как и многие другие, распространяется в виде Java-архива (файл *.jar*). Его не обязательно распаковывать: можно исполнить его прямо в Java, просто добавив переключатель *-jar* вот таким образом:

```
/usr/java/jre1.6.0_20/bin/java -jar jgnash2.jar
```

В *jGnash* раздражает только то, что ему крайне важна версия Java, с которой он собирается сыграть. Проще говоря, все, что отличается от официального релиза Sun 1.6, скорее всего, вызовет проблемы.

Программа для вязания

Knitter

Версия 0.5.3 Сайт <http://knitter.sourceforge.net>

Вы вяжете? Да, я знаю, что вы читаете *Linux Format*. Нет, я говорю «вяжете» не насчет «вязать лыко» — тут у нас сомнений нет. Короче, рассказав в HotPicks о приложении для вышивания, мы породили волну претензий от представителей других школ работы с нитями на то, что их хобби имеет куда более продвинутое, технически более детализованное — а стало быть, более умное — ПО. Оно, может, и так, но, при всей изощренности их инвентаря, нельзя не признать, что изображение слегка их подводит, когда надо выбирать имя приложения [*Knitter* — *англ.* вязальщик].

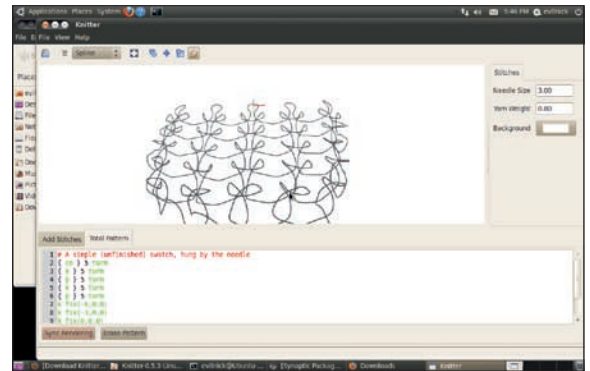
Knitter не просто помогает создать рисунок, но и демонстрирует процесс его изготовления во всем блеске OpenGL. Введите свой рисунок — и любуйтесь, как невидимые спицы плетут его из эфира, причем быстрее целой палаты бабушек в роддоме. При помощи области отображения OpenGL и четырех режимов

визуализации можно видеть не только изготовление всяких петель и накидов, но и как изделие драпируется с учетом силы тяжести.

Оглядите продукт

Здесь пригодится 3D-отображение; и хотя управление довольно рудиментарно, получить представление о готовом изделии очень просто. Настройки номера спиц и толщины пряжи позволяют варьировать вид изделия.

Нижняя панель отображает узор. Если вы разбираетесь в инструкциях по вязанию, можете ввести собственный рисунок или внести свои изменения и убедиться в том, что плетеные косич-



➤ Это не полная нейронная карта мозга Майка, а простенькая схема вязания. Хотя перепутать их легко.

ки отлично смотрятся или что ромбики на носках сочетаются с чулочной вязкой полотна. Условная запись в вязании — это как если бы кто-то попытался напечатать Lisp на половине клавиатуры; но для знакомых с ней она исполнена смысла.

Knitter использует набор инструментов *wxWidgets* и его расширения OpenGL; и то, и другое вам понадобится, если вы хотите компилировать с нуля. Имеется также двоичный дистрибутив, но здесь у вас могут быть самые разные результаты. У нас он отлично работал на Ubuntu, а на Fedora 13 — нет.

«Knitter демонстрирует процесс изготовления узора во всем блеске.»

Дисковая утилита

Dfeta

Версия 1.0 Сайт <http://cade.datamax.bg/away/dfeta>

Кому бы не пригодилось умение взглянуть в будущее? Помимо предотвращения карточных долгов и прогнозирования событий в сериале про Доктора Кто, вы бы также знали наилучшее время для расширения RAID.

Этот чудный инструмент командной строки не поможет вам с электричкой на 11:30 или с тем, уломаете ли вы ту рыженькую на Рождество, зато подскажет, когда переполнится ваш жесткий диск. Возможно, это и не перевернет вашу жизнь, но явно будет полезным, если вы затеваете сканирование семейного архива фотографий или скачивание всех дистрибутивов Linux.

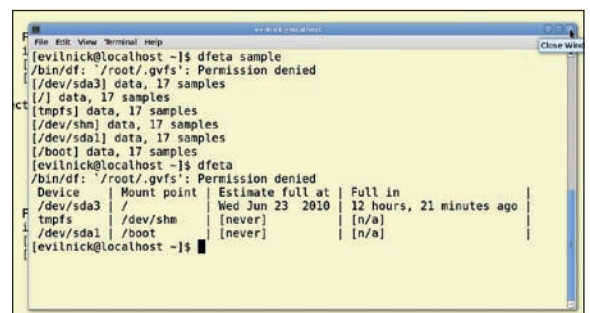
Концепция проста. *Dfeta* замеряет ваш жесткий диск через равные промежутки времени, производит несколько вычислений и сообщает, в какой момент ваш диск объявит вам об отсутствии на нем свободного места. Вроде одного из этих сайтов —

«узнай день своей смерти», только не так жутко. Даже если вы не являетесь ИТ-специалистом, отвечающим за хранение данных, этот удобный маленький инструмент вовремя напомнит вам о необходимости очистить корзину и удалять мусор почаще.

Делайте как надо

Естественно, чтобы он заработал, вам нужно регулярно проверять жесткий диск. Чем больше замеров вы делаете, тем точнее они будут, поскольку из общего значения будут исключены случайные отклонения. Частота его использования зависит от вас; файл, где хранятся данные, очень маленький, хотя имеет резервные

«Этот инструмент напоминает о необходимости очистить корзину.»



➤ *Dfeta* следит за быстротой заполнения жесткого диска, производит некоторые расчеты и прогнозирует, когда памяти придет конец.

копии (что со временем приводит к самоисполняющимся прогнозам). Но, поскольку он достаточно быстр и прост, вы даже не узнаете об этом. Самое лучшее — установить его где-то в **PATH** (*/usr/local/bin/* или где это будет наилучшим в вашей системе) и задать процедуру *Cron*. Одного раза в день должно быть вполне достаточно, но вы сами должны решить, сколько крошечных файлов данных вам нужно в *~/dfeta*.

Единственный момент, оставшийся неразъясненным — как это произносить. Ди-Эф-И-Ти-Эй? Деф-Ита? Ди-Фи-та? Хм-м...

Настольная издательская система

Scribus

Версия 1.3.7 Сайт www.scribus.net

Впервые мы описывали *Scribus* как проект на подходе во глубине истории HotPicks. И с тех пор он стал одним из лучших образцов офисного ПО с открытым кодом. Со времени его последнего полного обзора (LXF124) прошел уже почти год, и хотя с тех пор особых изменений по части функций не произошло, внутри проведена огромная работа.

Одной из извечных проблем в этой серии релизов была стабильность. Имелись серьезные трудности с надежностью функции отката, а также куча всяких мелочей, отказывающихся работать должным образом. И хотя это пока что релиз на стадии разработки (в преддверии итогового 1.4), разработчики потратили последние шесть или семь месяцев на обеспечение достойного функционирования движка PDF и ликвидацию всяких малопримечательных ошибок.

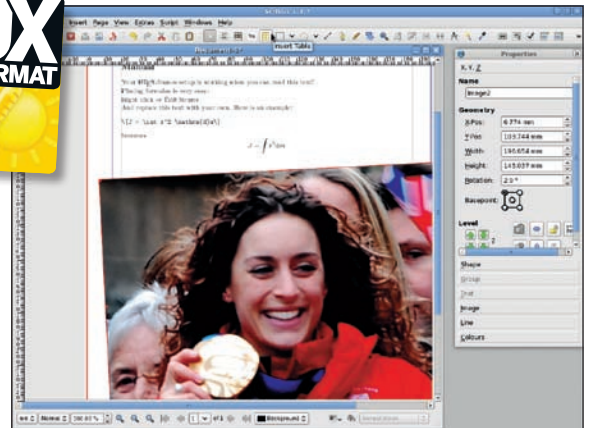
Если раньше вы не занимались настольными издательскими системами

в Linux, впечатление, видимо, будет сильным. В *Scribus* есть все функции, необходимые для дизайна профессиональных публикаций.

Про-мощность

Мы создали четырехстраничный материал в LXF96, используя только *Scribus* и другие инструменты с открытым кодом – он имеется на сайте www.linuxformat.ru/archives.phtml. Если ваши потребности скромнее, вас все равно удивит мощь инструментов и качество результатов. Внесены изменения по настройке PDF так, чтобы генерировать 1.5-совместимые файлы, а инструменты генерации интерактивных PDF (кнопок, текстовых полей, флажков

«Это один из лучших образцов офисного ПО с открытым кодом.»



► Помимо отличной обработки графики, *Scribus* имеет еще и режим отрисованного текста для вставки формул, нотных записей и многого другого.

и т.д.) не имеют себе равных в программах Linux. Кроме того, новый текстовый рендеринг использует внешние инструменты, такие как *Gnuplot*, *LaTeX* и *Lilypond*, для внедрения формул, графиков и прочего непосредственно в ваши документы.

Единственное предостережение – для пользователей стабильной версии *Scribus* (1.3.3.x): формат файлов изменился, и новые версии больше не отличаются совместимостью. Возможно, проще будет оставить две версии *Scribus*, чем отказываться от всех ваших старых документов.

Инструмент виртуализации

VirtualBox OSE

Версия 3.2.6 Сайт www.virtualbox.org

Мы здесь в *Linux Format* большие фанаты утилиты *virt-manager*, а также виртуальных машин *KVM/QEmu*. За последние 18 месяцев они бурно развивались, вплоть до момента, когда настройка виртуальной машины стала делом тривиальным. Еще одна альтернатива с открытым кодом – *VirtualBox*, проект с довольно сложной историей. Изначально разрабатываемый InnoTek, он был приобретен Sun, который ныне продан Oracle.

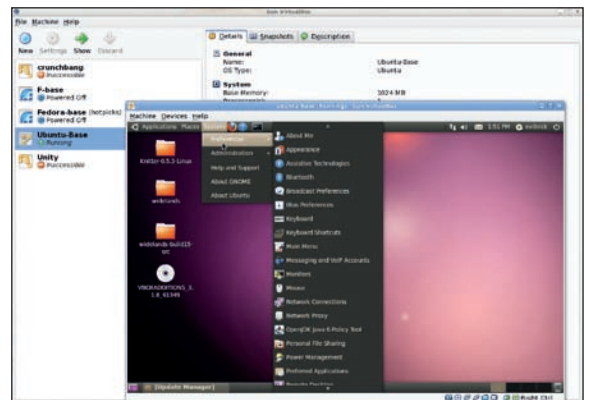
За это время возникли две версии ПО: вариант с закрытым кодом и более знакомый вариант с открытым кодом, о котором и пойдет речь. Основное их различие – в графических возможностях. Закрытая версия использует для отображения проприетарный код для RDP, а открытая – стандартный VNC-сервер. Проприетарная версия также поддерживает USB, позволяя системе-гостю управлять периферийными устройствами USB, такими как сканеры,

приводы DVD или что там еще бывает нужно подключить.

Отличная скорость

По части прочих функций, *VirtualBox* – зрелый и полнофункциональный конкурент *KVM*. Простой интерфейс и простая настройка обуславливают и необычайно простое использование, и если у вас достаточно системных ресурсов для выделения под гостевую ОС, вы не вдруг отличите виртуальную машину от реальной. Дополнительные модули ядра для гостевых дополнений, предлагаемые для гостевых Linux-систем, облегчают настройку сети и графической среды для вашей

«Вы не вдруг отличите виртуальную машину от реальной.»



► *VirtualBox OSE* отлично подходит для знакомства с новыми дистрибутивами и для создания экспериментальных программ.

виртуальной машины, а настройки настолько просты, что вам вряд ли придется консультироваться с обширной документацией. Возьмите образ установочного диска и следуйте инструкциям мастера создания виртуальной машины.

Если крайности нет, собирать программу самому особо незначит. Есть пакеты для всех основных дистрибутивов, и все они поддерживаются в обновленном состоянии. Когда вы будете читать этот материал, *VirtualBox 3.2.6* должен без всяких проблем найтись через ваш менеджер пакетов.

HotGames Развлекательные приложения

Settlers-подобная игра

Widelands

Версия 15 Сайт <http://launchpad.net/widelands>

Эта игра уже рассматривалась в HotPicks, но в ней произошел ряд улучшений. В последний раз, когда мы о ней рассказывали, игра была многообещающей, но ее песня не в меру часто портилась ошибками. Теперь она стала куда более играбельной и значительно улучшила графику и звук.

Вряд ли есть смысл погружаться в хитросплетения сюжета – руководство по игре поможет вам разобраться в различных типах строений и порядке их сооружения. Если вы когда-либо играли в старую классическую *Settlers II*, у вас уже есть отличное представление о *Widelands*, ибо она взята за основу. Строите, прокладываете пути, а затем наблюдаете, как ваши крошечные рабочие снуют по ним, доставляя всякие вещи. Соорудив достаточно строений

разного типа, со временем вы сможете создать апофеоз цивилизации – сэндвич с ветчиной. Это не только продемонстрирует ваши успехи в логистике, но и – как однажды сказал Наполеон – подпитает военную машину, к ее вящей славе.

Мило и стабильно

Игра очень захватывающая, в стиле старых версий *Settlers*. Некоторым игрокам нравится сравнительно медленный темп и напряженный микро-менеджмент подобных игр, однако это не для любителей быстрых и неистовых действий.

«Со временем вы сможете создать апофеоз цивилизации.»



» *Widelands* основана на *Settlers*, и все, кто играл в одну из классических версий, почувствуют себя как дома.

Хотя над некоторыми меню не помешало бы еще поработать, игра выглядит просто фантастически, и в нее очень интересно играть – как в режиме одиночного игрока, так и по сети.

Widelands также устроила новоселье на Launchpad, что несколько упростило разработчикам и пользователям способ добраться до самых свежих сборок. Для большинства сборок имеются бинарники, но покамест не гарантируется, что они будут работать на всех дистрибутивах Linux.

Тренажер полетов

SDL Sopwith

Версия 1.7.3 Сайт <http://sdl-sopwith.sourceforge.net>

Скорее это даже не имитатор, а путешествие в более простое и доброе время, когда мужчины были мужчинами и летали на машинах из дерева и парусины, паря наравне с орлами и останавливаясь, чтобы сбросить бомбу на все, что было им не по душе.

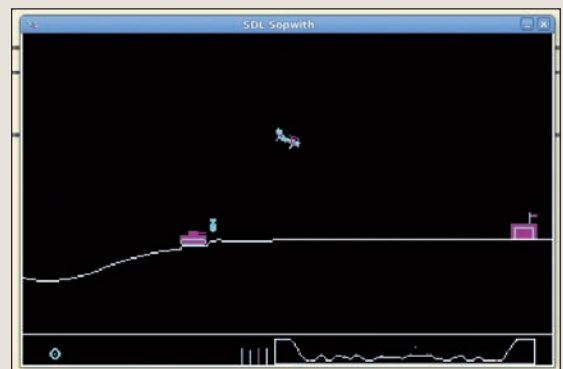
Sopwith – новое воплощение старой классики. Представьте себе *Microsoft Flight Simulator*, но с пушками и бомбами и без всяких там панелей управления, шкал или 3D-графики. Вернее, почти совсем без графики. В *Sopwith* все состоит из блоков ярких цветов, в палитре черного, белого, бирюзового и пурпурного. У игрока – бирюзовый цвет с пурпурными кантами, и он должен сразить злобного и мерзкого противника пурпурного цвета с бирюзовыми кантами, каковые нестерпимы в законопослушной, чистой и доброй вселенной.

Летная школа

Самая важная кнопка двумерного мира *Sopwith*, которую надо помнить – стрелка вниз: она разворачивает ваш самолет так, чтобы вы могли лететь справа налево, не повиснув вверх тормашками. Так же, как и с любым самолетом, чем круче угол подъема, тем медленнее он движется – вы же не хотите заглохнуть. У вас безграничный запас оружия, и никто не ждет, что вы приземлитесь. Потяните рычаг, ощутите рывок скорости – и разлетитесь в разноцветную пыль, если вас собьют.

Если вы сочли сражение своего рода испытанием огнем, переходите в режим

«Представьте *Microsoft Flight Simulator*, но с пушками и бомбами.»



» Убрать колодки! Куда опять запропастилась кнопка сброса бомб?

одиночного игрока: на первом уровне вам будет вредить только ваша неумелость. На более высоких уровнях придется освоить управление стрельбой ракетами, сбросом бомб и уходом от огня. Во Второй мировой войне средняя продолжительность жизни пилота Военно-воздушных сил Великобритании составляла 87 летных часов. В *Sopwith* 87 секунд – уже довольно приличный результат.

Программа синхронизации файлов

Pybag

Версия 0.3.1 Сайт <http://pybag.sourceforge.net>

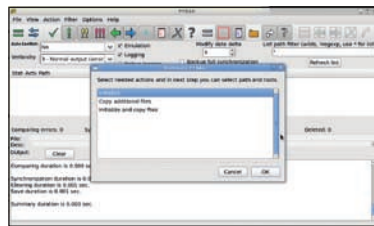
Вы, наверное, подумали, что это механизм, используемый Майком для транспортировки своего обеда [bag – *англ.* мешок]; но на самом деле все куда хитроумнее. Знаете задачу: у вас есть файлы на компьютере А, файлы на компьютере В и файлы на некоем устройстве – MP3-плеере, USB-брелке или SD-карте. Как получить самые последние файлы, где бы вы ни находились, не погрязнув в проблеме проблем – какой файл и куда скопировать в первую очередь?

Одно из возможных решений – хранить все свои данные на секретном массиве данных RAID на геостационарном спутнике. А еще можно применить *Pybag*. Эта скромная утилита Python не займет много места на ваших целевых дисках и избавит вас от сомнений насчет того, какой файл каким является, синхронизировав их для вас.

Здесь есть множество опций – пожалуй, их даже слишком много, поскольку интерфейс сначала может показаться довольно путаным. Главное – не забыть сперва

скопировать на съемный носитель файл **pybag.py** и запустить его оттуда. Вы получите запрос на инициализацию местоположения, что вы и должны сделать. Затем в меню File выберите Add Root, чтобы добавить директорию на локальной машине, которую решили синхронизировать.

Далее, отмените выбор Emulation Mode и выберите Synchronise в меню, чтобы скопировать файлы в свою директорию «bag». Разделы всестороннего фильтра и опций разрешения конфликтов охраняют вас от возможных проблем, но чтобы насладиться ими по максимуму, придется обстоятельно знакомиться с документацией.



➤ Пусть оно отчасти и путаное, но все же *Pybag* – полезное небольшое приложение Python.

Программа очистки диска

BleachBit

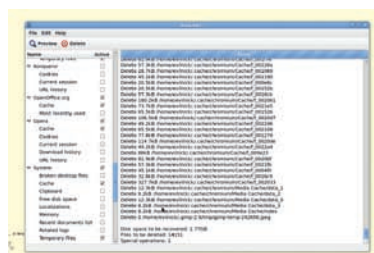
Версия 0.8.0 Сайт <http://bleachbit.sourceforge.net>

Отбеливатель [bleach – *англ.* отбелить] убивает 99,9% известных микробов. Наповал. Нас всегда интересовало: а как насчет неизвестных? Мы размышляли об этом, продираясь по своим домашним папкам в поисках ненужных файлов – чтобы их удалить и освободить место. И когда чистили cookies браузеров, тоже об этом думали.

Новый 2-в-1 *BleachBit* проникает в вашу файловую систему и уничтожает (если вы его об этом попросите) все, что существовало до срока его реализации, и множество этих полезных, но порой опасных cookies, сохраненных паролей и прочего. Многие браузеры в наше время включают некую разновидность режима сохранения конфиденциальности или программу для собственной очистки, но намного проще стереть все за один присест – и ваши учетные записи так будут сохранены, да еще и высвободится поразительно много места на диске.

Мы уже говорили о *BleachBit* в HotPicks – давно, в дурашливо-оптимистичные дни LXF117. С тех пор программа улучшена, и теперь поддерживает намного больше приложений, в том числе *Chrome* (и *Chromium*), *OpenOffice.org*, *Skype* и *Pidgin*.

Работать с ним просто, но вас всегда будет снedaть дума, не удалите ли вы что-нибудь важное. Можно, конечно, проверять файлы перед удалением, но это подорвет его репутацию простоты и быстроты. Неплохо бы сделать пробу и убедиться, что фильтры задерживают то, что нужно. LXF



➤ Верните себе ценное дисковое пространство, стерев накопившийся мусор.

Также вышли

Новые и обновленные программы, тоже достойные внимания...

➤ Worker 2.17.8

Двухпанельный файловый менеджер, теперь с более точной статистикой объема.

www.boomerangsworld.de/worker

➤ Gjoits 2.3

Простая программа для заметок древовидной структуры.

<http://bhepple.freeshell.org/gjoits>

➤ Clickity 0.2.0

Имитатор мыши для тех, у кого ограничена подвижность руки.

<http://sourceforge.net/apps/mediawiki/clickity>

➤ Cdrtools 3.0

Крупное обновление программы прожига дисков в Linux, включая двухслойные DVD и Blu-ray.

<http://cdrecord.berlios.de/private/cdrecord.html>

➤ Untangle 7.3.1

Превратите свою машину Linux в сетевой шлюз с подключаемыми модулями.

www.untangle.com



➤ Держите свою сеть в порядке, *Untangle* вам в помощь.

➤ Mairix 0.22

Индексация и поиск по файлам электронной почты.

www.rc0.org.uk/mairix

➤ Wireshark 1.4.0rc1

Самый лучший сетевой анализатор с открытым кодом.

www.wireshark.org

➤ Arista 0.9.5

Простая в использовании утилита транскодера.

www.transcoder.org

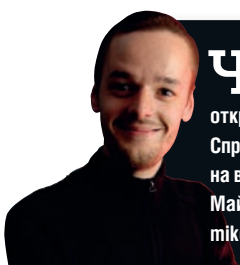
➤ Mission X 1.0

Отличная стрелка с боковой прокруткой.

<http://missionx.sourceforge.net>

На диске

Множество дистрибутивов и программ для изучения...



Читайте, чтобы найти всю информацию, которая вам понадобится для использования этого DVD! Если вы новичок в Linux, откройте `index.html` на диске (Сторона 1) и перейдите в раздел Справка: там вы найдете мини-учебник, дающий ответы на вопросы справа.
Майк Сондерс, редактор диска
mike.saunders@futurenet.com

- » Что такое Linux?
- » Что такое дистрибутив?
- » Загрузка ПК с DVD
- » Разбиение жесткого диска на разделы
- » Навигация по файловой системе
- » Использование командной строки
- » Установка программ

Дистрибутив Linux

Slackware 13.1

Дистрибутивы приходят и уходят. Одни задерживаются ненадолго; другие появляются под фанфары, а потом исчезают в никуда; третьи перерождаются в иной дистрибутив. Но ни один не сравнится по выживаемости со Slackware, главным долгожителем в мире Linux. Появившийся на свет в 1993 году, Slackware имеет самых преданных сторонников из всех дистрибутивов Linux – и нетрудно понять, почему.

Slackware – это настолько чистый Linux, насколько таковой возможен без поворота на путь Linux From Scratch. Его пакеты почти полностью основаны на первичном исходном коде, без всяких специфических для дистрибутива заплат, осложняющих жизнь. Это просто файлы `.tgz`, и вы можете

работать с ними, не пачкаясь с `cpio` или `ar`. Скрипты загрузки просты, система хорошо документирована, и все исполнено смысла. Как гласит старинная поговорка, «Используя Ubuntu, познаешь Ubuntu. Используя Slackware, познаешь Linux».

Slackware нацелен на средних и продвинутых пользователей Linux; всю работу по установке берет на себя текстовый инструмент `Ncurses`. Это пред-

У нас имеется самый последний релиз Slackware, 13.1, на **LXFDVD** в каталог Distros. Он включен в виде трех ISO-образов для прожига на CD-R. (Помните, что вы должны записывать их именно как ISO-образы, а не просто копировать содержимое файла). Изготовив диски, загрузитесь с первого, чтобы запустить программу установки.

Мы здесь не углубляемся в процесс установки, равно как и в тонкости использования дистрибутива: все это объясняется в фантастической Slackware Book, которая помещена на **LXFDVD** в разделе Distros/Slackware в виде **slackbook-2.0.pdf**. Она описывает установку, настройку системы, сеть, X Window System и многое другое.

В типичных традициях Slackware, она полна специфических шуток

«Slackware нацелен на средний и продвинутый уровень в Linux.»

полагает знание командной строки и отсутствие боязни редактирования файлов конфигурации. А главное, Slackware предполагает, что вы работаете со своим Linux осознанно, и не суется в ваши дела, норовя «помочь» вам перезаписью ваших файлов конфигурации или творя темные делишки в фоновом режиме. Для бывалых линуксоидов это сущее благословение.

и ссылок, и это – действительно увлекательный способ стать подлинным гуру в Linux. Это англоязычный материал, но к нему существует слегка устаревший перевод на русский язык, находящийся на сайте <http://jack.kiev.ua/docs/slackbook/>. Более развернутую информацию про Slackware вы сможете получить на www.slackware.com.

Важно

ВНИМАНИЕ!

» Перед тем, как вставить DVD в дисковод, пожалуйста, убедитесь, что вы прочитали, поняли и согласились с нижеследующим.

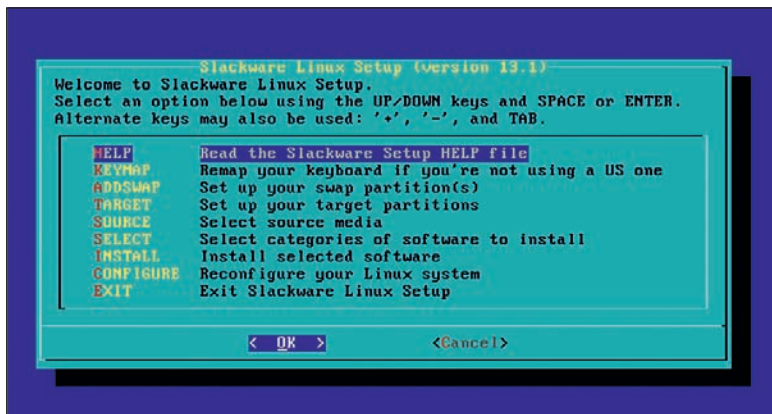
Диски *Linux Format* DVD тщательно проверяются на предмет отсутствия на них всех известных вирусов. Тем не менее, мы рекомендуем вам всегда проверять любые новые программы надежным и современным антивирусом.

Хотя процесс отбора, тестирования и установки программ на DVD проводится со всем тщанием, редакция *Linux Format* не несет никакой ответственности за повреждение и/или утрату данных или системы, могущее произойти при использовании данного диска, программ или данных на нем. Настоятельно рекомендуем вам создавать своевременные и надежные резервные копии всех важных файлов.

Чтобы узнать об условиях использования, просим вас прочесть лицензию.

Бракованные диски

В маловероятном случае обнаружения бракованного диска *Linux Format*, просим связаться с нашей группой поддержки по адресу disks@linuxformat.ru для получения содействия.



» Установщик Slackware – текстовый, но навигация достаточно проста.

Приложения рабочего стола

KOffice, HomeBank, K3b

KOffice – забавное старое приложение. С одной стороны, хорошо, что есть относительно небольшой и легко интегрируемый офисный пакет, бросающий вызов мощному (но увесистому) *OpenOffice.org*. С другой – он, похоже, пребывает в состоянии непреходящей недоделанности, о чем вы можете прочитать в нашем материале на стр. 12. Незвизрая на его недостатки, многие поклонники KDE считают его вполне достойным пакетом, и мы включили его последний исходный код в каталог Desktop на DVD.

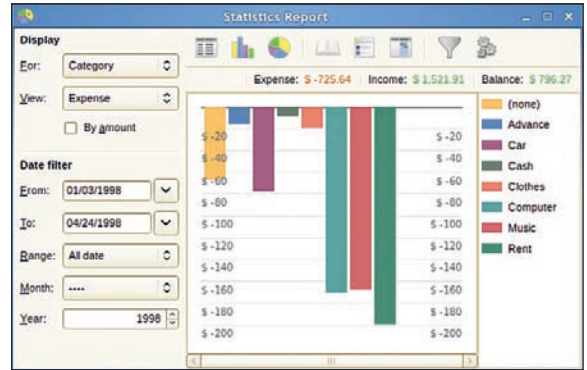
Далее идет *HomeBank*, отлаженное персональное финансовое приложение. *HomeBank* существует уже долгие годы, появившись на свет в 1995 году на Amiga, и на сегодняшний день это, несомненно, самое мощное приложение для финансового менеджмента под Linux, с поддержкой множества файловых форматов для импорта и экспорта, множественных счетов, различных форматов отчетов и отображением годового бюджета. Вер-

сия 4.3 снабжена новым «мастером кошелька» для настройки вашего первого счета, а также предварительной настройкой категорий и исправлением ошибок.

Для установки *HomeBank 4.3*, скопируйте файл **homebank-4.3.tar.gz** с LXF DVD в домашнюю директорию, затем откройте Help/New To Linux на диске и прочитайте руководство по компиляции программ. Если вы уже собирали приложения из исходных текстов, просто сделайте вот что:

```
tar xfvz homebank-4.3.tar.gz
cd homebank-4.3
./configure
make
make install
```

Помните, что финальную команду нужно ввести от имени root. Ввод **homebank** будет запускать приложение. (Если вам не хватает каких-либо зависимостей, требуемых для компиляции исходного кода, вам будет сообщено об этом на стадии **./configure**, так что найдите их в вашем менеджере пакетов и установите.)



➤ Ну на кой было покупать столько яиц Фаберже?! Пусть *HomeBank* покажет весь масштаб глупости ваших приобретений.

Еще один яркий момент в разделе Desktop этого месяца – *K3b 2.0*. Матерых KDE'шников могло огорчать, что данный исключительно разносторонний инструмент для прожига и риппинга дисков малость приотстал от KDE 4, но версия 2.0 их утешит. Внешне дизайн такой же, как и в прошлых релизах, но в нем есть отличные новые функции, типа поддержки Blu-ray и рационализированного интерфейса.

Другие программы

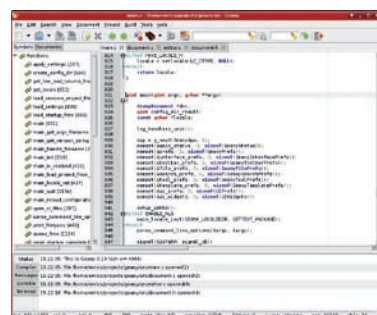
Интернет, разработка

В разделе Internet имеются клиенты BitTorrent, описанные в нашем Сравнении на стр. 14. А если вы уже прочли материал про *Firefox*, вам, небось, не терпится узнать, как новая версия этого браузера смотрится на фоне конкурентов; поэтому мы включили на DVD версию разработчиков для грядущего релиза 4.0. Перейдите в папку **dev-preview** в *Internet/Firefox* и скопируйте **firefox-4.0b3.tar.bz2** в домашнюю директорию. Неплохо было бы сохранить настройки вашего *Firefox*, скопировав **.mozilla** (в вашей домашней директории) в, например, **.moz_backup**. Затем распакуйте файл **.tar.bz2**, перейдите в полученную директорию и запустите **firefox**. Пока он не очень отличается от линейки 3.x, но вы оцените большее быстродействие и улучшенные возможности HTML 5.

В каталоге Development есть несколько инструментов для кодеров. *Codemetre* – инструмент для анализа; он умеет сравнивать две версии программного проекта, описывая, чем они отличаются в плане исходного кода. Он читает исходные файлы Ada, C, C++ и Eiffel и отвечает комментариями

и пустые строки – подсчеты производятся по исходному коду в чистом виде.

По части редакторов, у нас есть *Geany*, симпатичный редактор *GTK 2* с основными функциями интегрированной среды разработки (Integrated Development Environment, IDE). Он поддерживает подцветку синтаксиса большого количества языков, многопанельную раскладку и просмотр сообщений компилятора прямо в программе. Редактор более старой школы – *Vim*, сочетающий «преимущества Vi» с удобствами GUI.



➤ *Geany*: все хорошие функции IDE, без лишнего балласта. Нам нравится.

И это ещё не все!
Linux Gamers Live

Раздела игр на DVD этого месяца у нас нет, зато есть кое-что получше – *Linux Gamers Live*, превосходный дистрибутив, предоставляющий 17 лучших образчиков развлекательных программ с открытым кодом. Загрузите компьютер с LXF DVD, и после выбора языка вы попадете на рабочий стол. Внизу располагается панель с кнопками быстрого запуска имеющихся и щелкайте по ним, чтобы их запустить. Закончив развлекаться, щелкните правой клавишей по рабочему столу и выберите Shutdown в меню.

Если вам нужны скоростные гонки, попробуйте *Extreme Tux Racer*: в ней надо управлять всеми любимым пингином, скользя вниз с горы и собирая

➤ Спасите пингинов от самобуйства в *Pingus*.

рыбешек. *Neverball* – напоминание о *Super Monkey Ball*, где вы гоняете по арене мяч с целью насобирать объекты и манипулируете пейзажем с помощью мыши; а вот клавишей по рабочему столу и выберите Shutdown в меню.

Foobillard – более спокойный спортивный тренажер. Затем есть еще *Pingus* (головоломка в стиле *Lemmings*), и – конечно же – вариант классического *Tetris* в виде *LTris*. Буквально на любой вкус.

На диске

Самый популярный в мире дистрибутив — в мобильной форме!

Дистрибутив Linux

Ubuntu Netbook Edition 10.04

У вас нетбук, ноутбук или слабый компьютер с Linux? Любите Ubuntu, но считаете его чересчур требовательным для своего крупнозернистого дисплея или малого ОЗУ? Или просто ищете быстрый и легкий дистрибутив Linux со свежим новым интерфейсом пользователя? Ubuntu Netbook Edition (бывший Netbook Remix) — вот ответ. Он обладает всеми прелестями обычного Ubuntu, но нацелен на машины, более ограниченные в плане ресурсов. Вы получаете прекрасную поддержку оборудования, широкий выбор программ и приятный, отлично настроенный интерфейс, пользоваться которым — сплошное удовольствие. Некото-

рые нетбуки содержат собственные версии Linux, но если вам нужен более известный дистрибутив с большим количеством программ в онлайн-репозиториях пакетов, берите Ubuntu Netbook Edition. Вот рекомендуемые системные требования:

- » 1,6-ГГц Intel Atom CPU
- » 512 МБ ОЗУ
- » 4 Гб места на жестком диске

Сравните их с 1 Гб ОЗУ и 15 Гб для настольного варианта. 1,6 ГГц может показаться перебором для легковесного дистрибутива, но это просто современный стандарт для CPU нетбуков — он может неплохо работать и на более старых чипах. А самое главное — это рабочий стол: поскольку Netbook Edition концентрируется на высокоскоростной работе портативного компьютера без всех этих наворотов настольных ОС, он просто летает (да и время загрузки тоже впечатляет). И это не значит, что выглядит он убого — наоборот, его GUI очень мил (взгляните на экранные снимки).

Нетбуки Live

Ubuntu Netbook Edition работает в «живом» режиме непосредственно с LXF DVD, и вам



» В Ubuntu Software Center множество программ.

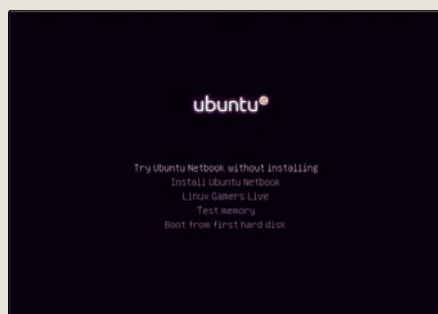
не придется устанавливать его, чтобы познакомиться с ним. Просто загрузите компьютер с DVD и нажмите на Enter в меню загрузки. Если вы захотите установить Ubuntu на жесткий диск вашего нетбука, следуйте инструкциям внизу. Можно установить его вместе с Windows, настроив меню двойной загрузки, и выбирать из него нужную вам ОС.

Основная черта отличного интерфейса Netbook Edition — панель категорий внизу слева, обеспечивающая доступ к крупным значкам приложений справа. Программы запускаются в полноэкранном режиме — щелкните по значку Ubuntu вверху слева, чтобы минимизировать все приложения и вернуться к виду значков быстрого запуска. Значки текущих программ появляются справа от логотипа Ubuntu, подобно мини-панели задач.

» Netbook Edition включает самые популярные свободные приложения, такие как Firefox, F-Spot и Rhythmbox.



Шаг за шагом: Ubuntu Netbook Edition



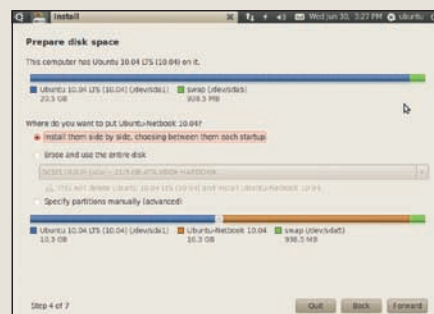
1 Загрузка

Загрузите компьютер с LXF DVD и выберите язык. Перед вами появится это меню загрузки — просто нажмите на Enter, чтобы продолжить.



2 Рабочий стол

На рабочем столе вы сможете познакомиться с наличными программами. Нажмите на значок Install (в Favourites и System), чтобы запустить программу установки на жесткий диск.



3 Разбиение на разделы

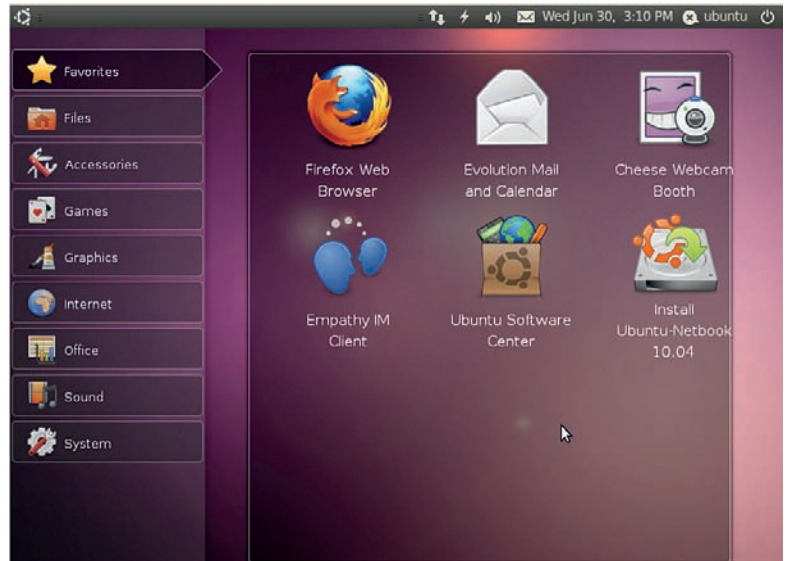
Следуйте инструкциям, чтобы выбрать свое местоположение. В открывшемся экране разбиения на разделы можно отвести весь диск под Linux или сделать установку наряду с другой ОС.

Как бы мне?..

Если вы – новичок в Linux, то после установки Ubuntu вы можете слегка недоумевать по поводу того, как справиться с определенными задачами. Вот список наиболее общих задач и способов их решения...

- » **Работа в Сети** Щелкните по Favourites > Firefox, чтобы запустить второй в мире по популярности браузер, подкрепленный сотнями расширений.
- » **Работа с файлами** Откройте Files – и получите доступ к своим документам, музыке, изображениям и закамкам.
- » **Воспроизведение музыки и видео** В категории Sound вы найдете *Rhythmbox* для воспроизведения музыки и *Movie Player* для просмотра видеофайлов.
- » **Редактирование документов** Ubuntu Netbook Edition включает *OpenOffice.org*, офисный пакет, содержащий текстовый редактор, редактор электронных таблиц и инструмент для презентаций, в категории Office.
- » **Чат онлайн** В Internet запустите *Empathy*. Это изобилующий функциями клиент для обмена сообщениями; он поддерживает все популярные протоколы (MSN, AIM, Google и т.д.). Там также имеется *Pino*, для пользователей Twitter.
- » **Настройка системы** Щелкните по категории System: сверху вы увидите инструменты администратора для управления оборудованием, добавления новых пользовательских учетных записей, и т.д. Прокрутите вниз, чтобы увидеть предпочтения – темы, горячие клавиши и т.п.
- » **Поиск новых прогограмм** В Favourites откройте Ubuntu Software Centre, чтобы загрузить программы из Интернет. Вы можете просматривать категории программ и щелкнуть по значку Install, чтобы скачать приложение.
- » **Выключение** Щелкните по значку питания сверху справа экрана, чтобы выключить компьютер, выйти из системы или перейти в спящий режим.

Главная страница Ubuntu Netbook Edition – www.ubuntu.com/netbook, но там находятся в основном материалы рекламного характера. Если вам нужна помощь по работе с дистрибутивом, или вы хотите присоединить свои усилия к усилиям разработчиков, загляните на www.ubuntu.com/community, где имеются ссылки на большое количество ресурсов. Документацию по дистрибутиву вы найдете на другом полезном ресурсе – www.ubuntuguide.org.



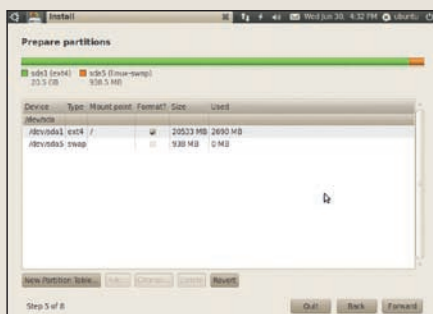
При проблемах с Ubuntu Netbook Edition, загляните на форумы русскоязычного сообщества Ubuntu (forum.ubuntu.ru). Они совершенно бесплатны для подписчиков, и там много постоянных пользователей, способных помочь вам сориентироваться. **LXF**

» Он пурпурный, как и «большой» Ubuntu, но сконцентрирован на супер-быстром доступе к вашим любимым приложениям.

Не пропустите..

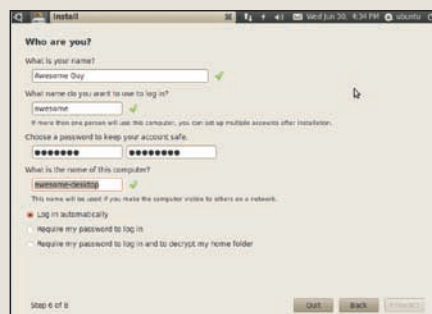
Cheese
Улыбнитесь! Общайтесь через свою web-камеру и Cheese, в котором есть множество потрясающих эффектов.

Pino
Поддерживайте связь с друзьями по Twitter и Identica через Pino, клиент для микроблогов.



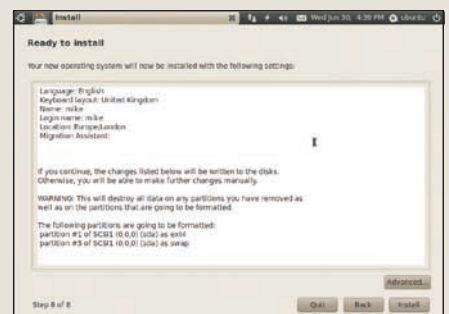
4 Разбиение вручную

Если вы решили разбивать на разделы вручную, создайте раздел минимум 3 ГБ в формате ext4 с точкой монтирования / и раздел swap, в два раза превышающий по размеру ваше ОЗУ (но не более 2 ГБ).



5 Учетная запись

Вам будет предложено создать пользовательскую учетную запись для входа в систему. Помните, что в имени пользователя и в пароле важен регистр – и не забывайте их!



6 Подтверждение

Программа установки попросит подтвердить, что вы хотите продолжить (щелкните по Advanced, если надо изменить настройки программы загрузки). После этого вы перезагрузитесь уже в новую систему!

СИСТЕМНЫЙ администратор

Клонировем Windows с помощью Symantec Ghost

Насколько неувязима ваша беспроводная сеть?

Active Directory вместо рабочей группы

Настраиваем DSPAM – ваш личный спам-фильтр

Как спасти данные, если отказал жесткий диск

Модифицируем BIOS

Все ли возможности ClamAV вы используете?

Что важно знать об IP-телефонии

Админские сказки

www.SAMAG.ru

В «Системном администраторе» вы не прочтете о:

- котировках валют
- сплетнях
- погоде
- политике
- развлечениях



В вашем распоряжении:

- опыт лучших IT-специалистов
- новые идеи и полезные советы
- самые эффективные решения в области системного и сетевого администрирования



Подпишитесь сейчас!

Роспечатать – 20780, 81655
 Пресса России – 87836
 Online-подписка – www.linuxcenter.ru



Академическая программа для учебных заведений

Mandriva Linux

Mandriva.Ru предоставляет учебным заведениям лицензию, дающую право на неограниченное по числу рабочих станций использование дистрибутива Mandriva Linux на всех компьютерах в образовательном учреждении, всех компьютерах преподавателей и всех компьютерах учащихся, в том числе и домашних.

По этой программе учебное заведение получает:

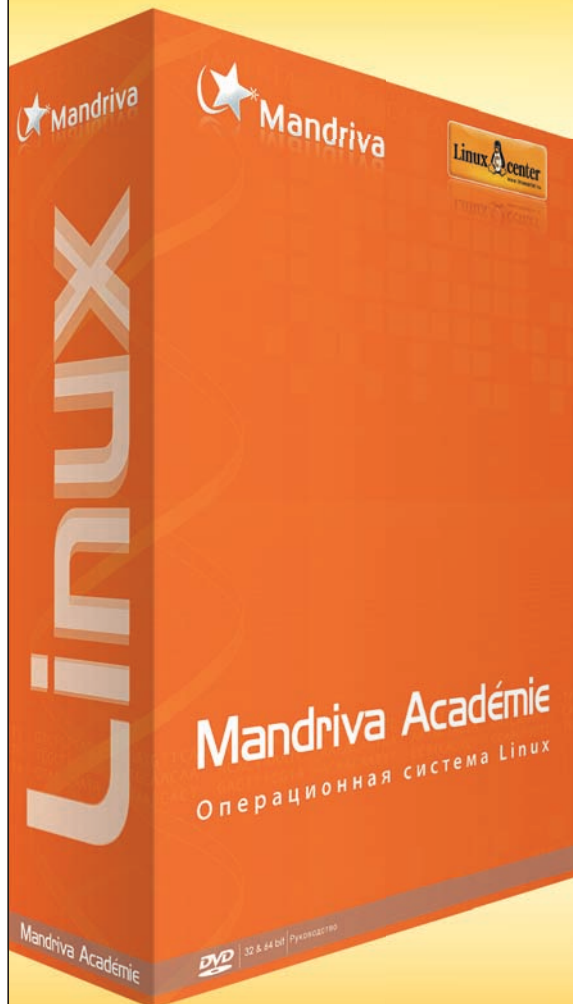
- ★ свежие версии дистрибутива Mandriva Linux (дважды в год)
- ★ доступ к обновлениям системы
- ★ техническую поддержку

Комплект поставки:

Mandriva Linux Powerpack 2009.1 Spring — 32- и 64-битные версии (2 DVD), а также печатное руководство ★ Mandriva Free 2009.1 Spring ★ Mandriva One 2009.1 Spring ★ Репозиторий Mandriva 2009 — бинарные пакеты для платформы x86 (4 DVD) ★ EduMandriva (1 DVD) — дополнительное ПО для образования ★ Академическая лицензия

www.mandriva.ru

Тел.: (812) 309-06-86, (499) 271-49-55
info@mandriva.ru



Сентябрь 2010
LXF DVD 135

LINUX
FORMAT

Сентябрь 2010
LXF DVD 135

LINUX
FORMAT

RHELinuxOS 2010.07

Настольная операционная система —
LiveCD с возможностью установки
на жесткий диск. Множество разновидностей!

- » RHELinuxOS KDE Рабочий стол KDE 4.4.5
- » RHELinuxOS Gnome Рабочий стол Gnome 2.30.2
- » Проприетарные драйверы и кодеки в комплекте



А ТАКЖЕ: RHELinuxOS LXDE, RHELinuxOS XFCE и RHELinuxOS Openbox
для старых или маломощных компьютеров

Информация о диске

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
- » **имя_программы-1.0.1.tar.gz** — обычно это исходный код;
- » **имя_программы-1.0.1.tgz** — тот же файл, что и выше по списку: «tgz» — это сокращение от «tar.gz»;
- » **имя_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;
- » **имя_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
- » **имя_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
- » **имя_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
- » **имя_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Внимательно прочтите это перед тем, как использовать LXF DVD!

Slackware 13.1

Дистрибутив «старой школы» — для тех,
кто хочет изучить Linux, а не Ubuntu



- » **Ubuntu Netbook Remix 10.04**
Система для нетбуков или маломощных компьютеров
- » **Linux Gamers DVD**
Самые популярные игры для Linux в удобной упаковке
- » **Koffee 2.2**
Новая версия офисного пакета для KDE

А ТАКЖЕ: пре-релиз Firefox 4, средства разработки, тестирование...

Страница 1

РАЗВОЙ СТОЛ

- Desktop** — открытый дистрибутив на GTK+
- elm** — менеджер событий и задач
- GNUP** — ресторатор пакетов
- GNOME Do** — утилита для быстрого запуска приложений
- HomeBank** — приложение для учета финансов
- Hyper** — редактор текстов
- ImageMagick** — набор программ для манипуляций файлами, изображениями
- КЗР** — интерфейс для загрузки дисков
- KOffice** — офисный пакет для среды KDE
- Lumina** — графический интерфейс для создания HDR-изображений
- mp3ENC** — текстовая энциклопедия
- PaZiP** — инструмент для создания таблиц
- Planner** — утилита для планирования в календаре проектов
- RevelioBook** — движок для загрузки мультимедиа и других вещей
- Shovel** — скрипт для организации фотографий
- Ubuntu Tweaks** — утилита тонкой настройки Ubuntu

РАЗБОТКА

- CodeMeter** — утилита для сравнения версий
- devel-pp** — редактор объектов
- devel** — среда разработки
- ink-config** — легковесная замена GNU autotools

- Paco** — пакетный менеджер для Linux-систем
- Vim** — vi-подобный редактор

АДМИНИСТРИВТИВЫ

- Ubuntu Netbook** — редакция Ubuntu для нетбуков
- Linux Gamers Live** — игровой дистрибутив с играми
- Slackware** — старейший из ныне живых

НОУТБУКИ

- Beasabn** — утилита для удаления ненужных файлов
- Dietz** — программа для сканирования свободных пространств
- Ernie Mode** — решение для коммуникации между человеком и машиной
- Idnash** — свободный персональный менеджер файлов
- Killer** — программа для создания и визуализации резервных шаблонов для загрузки
- RVBAG** — кроссплатформенная программа для синхронизации и резервного копирования файлов
- Reckitack** — функциональный микрософтовый процессор
- Scalix** — окружение программы для профессиональной версии
- SQL-Support** — классическая компьютерная игра
- Win7toolbox** — виртуальная машина
- Winlands** — окружение стратегии реального времени

Пожалуйста, перед использованием АСНОНО ДИСКА ознакомьтесь с опубликованной в журнале инструкцией!

КОММЕНТАРИЙ Просьба к вашим пожеланиям и предложениям по электронной почте: info@linuxformat.ru

ДЕФЕКТИВНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь пожалуйста по адресу disk@linuxformat.ru

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле `/etc/default/cdrecord`. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor=0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?
Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что если у вас нет устройства, с помощью которого можно было бы записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков: подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика.

Страница 2

PCLinuxOS во всех возможных вариантах

- Интернет**
- Deluge** — BitTorrent-клиент на Python/RUBY
- Flexbox** — свежая версия популярного веб-браузера
- KTorrent** — BitTorrent-клиент для среды KDE
- Monsieur Project** — клиент BitTorrent на C#
- Protonai** — обработчик почты
- qBittorrent** — BitTorrent-клиент на Qt
- Torlim** — минималистичный и простой клиент BitTorrent
- Transmission** — первый torrent-клиент для Linux и OS X
- Uzix** — BitTorrent-комбайн
- WordPress** — популярная CMS

СИСТЕМА

- Bochs** — утилита для тестирования производительности жестких дисков
- Busshot** — маленькая панель датчиков BIOS, отображающий загрузку каждого процессора
- GRUB2** — редактор загрузочных записей
- Кальпи** — первый GUI для KDE4
- RRDtool** — открытый стандарт для хранения и обработки линейных данных
- Xtcc_Task_Manager** — менеджер задач

Настоящий документ содержит информацию и материалы на русском языке, подготовленные для использования в дистрибутиве Linux Format. Мы также рекомендуем всегда иметь под рукой инструкцию по использованию Linux Format, так как она может быть полезна в случае возникновения проблем. Мы не несем ответственности за любые повреждения, разрушения или иные убытки, которые могут возникнуть в результате использования этого DVD, предоставленных на нем программ или данных. Перед тем, как устанавливать какое-либо ПО на компьютер, пожалуйста, ознакомьтесь с лицензией и условиями использования. Этот документ является собственностью Linux Format. © 2010, г. Екатеринбург. Студенческая ул., д. 9. Лицензия МПР ВАФ 77-15.

Пропустили номер?

» Мир свободного ПО богат и разнообразен, а потому далеко не все можно вместить в рамки одной статьи. Linux Format обходит эту проблему, публикуя серии статей по самым актуальным вопросам; но что делать, если вы поймали интересующий вас материал на середине? Обратитесь в Линуксцентр по адресу www.linuxcenter.ru и закажите желаемый номер журнала! Он доставляется как в печатной, так и в электронной форме, поэтому с момента открытия браузера и до получения нужного вам выпуска LXF может пройти не более нескольких минут!

Прямо сейчас для заказа доступны следующие номера:



LXF132
Июнь 2010

- » **Пробуй любой дистрибутив!** Ваше пошаговое руководство по виртуализации в Linux.
- » **ARM** Обзор архитектуры, приобретающей все большую популярность.
- » **Живой звук** Создайте себе собственную систему реального времени.
- » **Файлы журналов** Где искать информацию, если что-то идет не так.

LXF DVD: PC-BSD 8.0, OpenClipart 2.0 и FlightGear 2.0

Печатная версия:
http://shop.linuxformat.ru/lxf_132/

Электронная версия в формате PDF:
http://shop.linuxformat.ru/elxf_132/



LXF133
Июль 2010

- » **Двойная загрузка** Правильный путь к мирному сосуществованию двух ОС.
- » **3D-редакторы** Сравним пять основных инструментов, доступных для Linux.
- » **XZip** Тестируем новый метод сжатия данных.
- » **Маршрутизация по политикам** Неизвестные особенности ядра Linux.

LXF DVD: Ubuntu 10.04 LTS LXF Edition, Sabayon 5.3 KDE & GNOME

Печатная версия:
http://shop.linuxformat.ru/lxf_133/

Электронная версия в формате PDF:
http://shop.linuxformat.ru/elxf_133/



LXF134
Август 2010

- » **Linux против iPhone** Удастся ли смартфонам на базе свободной ОС закрепиться на рынке?
- » **OpenStreetMap** Свободный картографический сервис – находка для скитальца.
- » **Периферия** Запрограммируйте взаимодействие с внешними устройствами.
- » **Quickly** Действительно быстрое создание приложений для Linux.

LXF DVD: Fedora 13, Linux Mint 9, Mepis 8.5, «школьные» дистрибутивы

Печатная версия:
http://shop.linuxformat.ru/lxf_134/

Электронная версия в формате PDF:
http://shop.linuxformat.ru/elxf_134/

Ну, а если вы хотите быть уверенными, что не пропустите ни один номер журнала – оформите подписку! Помните, что все подписавшиеся на печатную версию журнала через www.linuxcenter.ru получают электронную версию в подарок!

Спешите на www.linuxformat.ru/subscribe/!

Специальное предложение!

Январь–июнь 2009

К нам в редакцию периодически приходят письма с вопросами, где можно купить предыдущие выпуски LXF. Если вы тоже озадачены этой проблемой, мы рады сообщить, что в интернет-магазине ГНУ/Линуксцентра продолжается распродажа журналов за 2009 год! Вы можете приобрести подшивки LXF за шесть или двенадцать месяцев со скидкой 30%.

Спешите — журналов осталось не так уж много!

Июль–декабрь 2009

http://shop.linuxformat.ru/lxf2009_01_06/

Январь–декабрь 2009

http://shop.linuxformat.ru/linuxformat_2009/

http://shop.linuxformat.ru/lxf2009_06_12/

Школа LXF

Спонсор рубрики
PingWin Software!
Созданная в мае 2009 года компания занимается поддержкой свободных продуктов, сообществ их разработчиков, пользователей и внедренцев.
www.pingwinsoft.ru

Обмен опытом и передовые идеи по использованию свободного ПО в образовании

СПО и новый стандарт для начальных классов

С появлением образовательных стандартов нового поколения, информатика как предмет еще сильнее закрепляется в начальной школе, теперь уже не только в предметной области «Технология», но и в области «Математика и информатика». Помочь учителю начальных классов вызвался **Александр Казанцев**.



Наш эксперт

Александр Казанцев

К. т. н., доцент кафедры информатики Глазовского государственного пединститута, руководитель проекта EduMandriva, автор локализаций (и разработок) для Mandriva, Klavaro, Gamba2 и LXDE.

Новый образовательный стандарт начальной школы, который начинает действовать с этого учебного года, а полностью вступает в силу с 2011 года, требует применения информационных технологий и средств практически с первых дней нахождения ребенка в школе. Решая верную задачу обучения детей, которые уже в основной своей массе знакомы с компьютерами, с использованием современных средств и технологий, этот стандарт тем не менее привносит множество проблем, начиная от пробелов в знаниях учителей начальных классов и заканчивая проблемами с самим программным обеспечением.

До настоящего времени компьютеры и программы были (за редким исключением продвинутых программ раннего обучения информатике) уделом небольшой группы преподавателей одноименного предмета, который, в свою очередь, опирался на методики и программные средства, рассчитанные на школьников 8–11 классов. В случае же начальной школы мы имеем иную возрастную группу и другие направления использования программных продуктов, а также другую группу учителей – начальных классов – которые в своей массе незнакомы не то что со свободными, но и с платными продуктами.

Давайте разберем, как свободное программное обеспечение может помочь нам в реализации данного стандарта. Особое внимание мы будем уделять соответствию ПО возрасту учащихся, что будет подразумевать его упрощенность, эргономику и дизайн. Также стоит иметь в виду небольшое время работы, которое допустимо проводить учащимся начальной школы за компьютером, что автоматически должно отсеять очень сложное или громоздкое в части интерфейса ПО.

Выбираем базовую платформу

Начнем, как обычно, с платформы для наших программ – операционной системы. Кто-то может сразу возразить, что кроме Windows, ничего не подойдет. Увы – или к счастью – это не так: проприетарная платформа без проприетарных же оболочек также не идеальна для детей 7–12 лет. Linux же в этом свете может оказаться более настраиваемым, хотя беглый обзор рекомендованных к применению в школах дистрибутивов, имеющихся на сайте spohelp.ru, показал неготовность для использования в начальной школе их всех без исключения – с точки зрения вышеизложенных факторов.

Так что теперь делать? Самое простое – это ждать у моря погоды, точнее, реакции разработчиков (надеемся, эта статья станет катализатором данного процесса). Второй вариант – это вспомнить поговорку про спасение утопающих и выполнить необходимую адаптацию и настройку самостоятельно.

Последуем по второму пути и посмотрим, что мы можем сделать. Для этого разберем возможные варианты по ключевым факторам.

» **Безопасность** Один из главных факторов, так как мы имеем дело с неокрепшей детской психикой, а заложенные в стандарте требования (см. врезку) требуют достаточно серьезного использования сети Интернет. Настройка фильтрации контента может выполняться как централизованно провайдером, так и локально на школьных компьютерах. Ее можно выполнить, установив и настроив различные виды ПО (см. статью «Сравнение. Фильтры контента: *FoxFilter*, *WebCleaner*, *DansGuardian*, *Gnome Nanny*, *SquidSafe*, *Privoxy*» в LXFI33), установив приложение *Netpolice* с одноименного сайта в ПСПО, включив родительский кон-

Подшивка LXF со скидкой 30% стр. 107



троль в EduMandriva; но самое простое – это включить DNS-фильтрацию. Для этого необходимо лишь прописать на компьютер, подключенных к Интернету, следующие строки в файле `/etc/resolv.conf`:

```
nameserver 81.176.72.83
nameserver 81.176.72.82
```

» **Надежность** Если спросить лаборантов или преподавателей, которые отвечают за настройку компьютеров, по поводу поломки системы учениками, то они вам расскажут истории, достойные фильмов ужасов, и это учитывая почти взрослый возраст учащихся. В начальных классах – намеренно, а скорее, непреднамеренно – система будет изучена и перекроена вдоль и поперек за пару минут, в силу присущей возрасту любознательности. Но этого можно избежать. Тут опять же есть два варианта: или все запрещать, или не сохранять изменения. Так как вариант с запретом будет мешать работе программ и освоению компьютера, то практичнее будет использовать или гостевой доступ, или присвоение прав «только для чтения» основным конфигурационным файлам. Также как вариант можно использовать метод «восстановления домашнего каталога», который позволит иметь постоянную «чистую среду» для работы. В большинстве случаев вам будет достаточно сменить пользователя на root для файлов конфигураций в домашнем каталоге учащегося – то есть всем каталогам, начинающимся с точки. Для этого достаточно дать следующую команду (если что-то не будет работать, то верните владельца назад, заменив в команде root на имя пользователя):

```
chown -hr root:root /home/user/.*
```

» **Адаптивность** Стоит помнить, что мы работаем с детьми, которые требуют особых условий для работы. Поэтому нужно адаптировать рабочий стол под возраст учащихся. Что для этого нужно делать? Во-первых, стоит выбрать более яркую (понятное дело, не переборщице) цветовую гамму, увеличить размер элементов экрана – панелей, шрифтов. Также стоит зайти на gnome-look.org или kde-look.org и подобрать более «веселую» тему оформления и иконок. Если вы используете Gtk2-совместимые рабочие столы (Gnome, XFCE, LXDE), то обратите внимание на темы значков Somatic или Gartoon Redux. По моему личному мнению, рабочие столы KDE4 и Gnome слишком громоздки для детей, поэтому оптимальным выбором будет XFCE или LXDE. Возможно, стоит обойтись вообще без панели задач, заменив ее доком (типа Awn или Cairo-Dock, см. стр. 46), вынеся туда только нужные детям приложения, хотя это опять же не позволит изучить составляющие части интерфейса операционной системы. Также обратите внимание на шрифты – они должны быть достаточно крупного размера и хорошо читаемы; оптимальны гарнитуры Liberation Sans или замена экранного шрифта Tahoma или Verdana от Microsoft в версии от PingwinSoft (http://www.pingwinsoft.ru/news_posts/pingwin-typography-paket-svobodnyh-shriffov-otkryt-dlya-testirovaniya). Сглаживание шрифтов приветствуется.

Подбираем программы

Если вы еще не прочитали про стандарт нового поколения для начальной школы, то теперь самое время сделать это и перейти как к изучению декларируемых результатов освоения учебной про-



» LXDE в адаптации для младших школьников.

«Вариант с запретом будет мешать освоению компьютера.»

граммы, так и к подбору (или констатации факта его отсутствия или необходимости адаптации или доработки) свободного программного обеспечения. Более подробно об упоминаемых программных продуктах можно узнать из наших предыдущих статей (см. врезку «Мы про это писали») или в Энциклопедии ПО проекта EduMandriva (http://wiki.edumandriva.ru/wiki/index.php/Энциклопедия_ПО).

» **Овладение основами логического и алгоритмического мышления, наглядного представления данных и процессов, записи и выполнения алгоритмов и умение действовать в соответствии с алгоритмом и строить простейшие алгоритмы** – здесь подойдут простейшие исполнители: *LittleWizard*, *Scratch* и, возможно, *Кумир* для более старших детей (также возможна работа с *Черепашкой* в различных вариациях – к примеру, с *KTurtle*).

» **Умение работать с таблицами, схемами, графиками и диаграммами, цепочками, совокупностями** – идеальным решением в этом случае является специально адаптированный для детей 7–12 лет офисный пакет *Oo4Kids* (ooo4kids.org), но, к сожалению, данный продукт не имеет русской локализации (надеемся, пока), и как вариант можно использовать *Gnumeric* как более простой по сравнению с *OpenOffice.org Calc*.

»

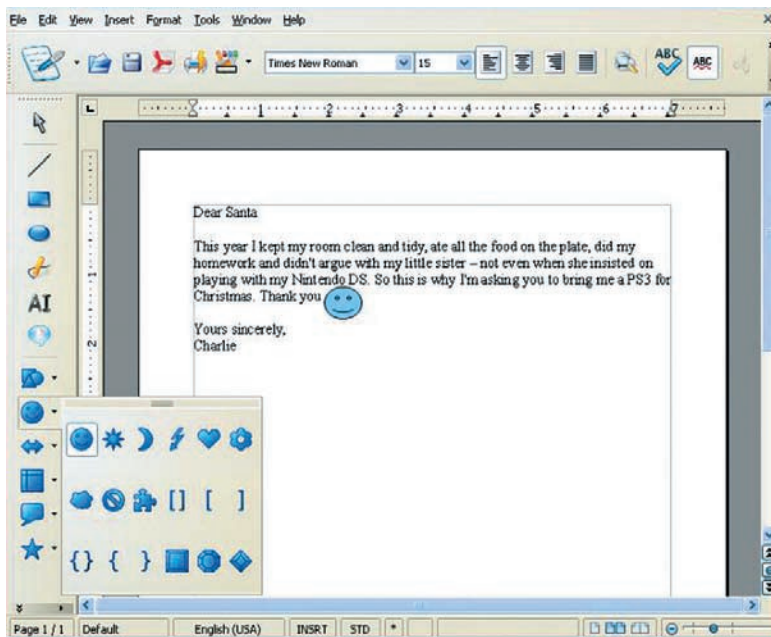
Стандарт нового поколения

В начальной школе Информатика и ИКТ теперь стали неотъемлемой частью не только предмета «Технология» (образовательные программы и пособия по которому есть и доступны), но и других предметов, привнеся многое из области информатики более старших классов. В стандарте декларированы следующие основные задачи обучения:

- » развитие логического и алгоритмического мышления;
- » развитие воображения;
- » обеспечение первоначальных представлений о компьютерной грамотности;
- » решение прикладных задач с использованием знаний, полученных при изучении других учебных предметов.

Результатов освоения образовательной программы, связанных с информатикой и ИКТ, по данной предметной области в стандартах упоминается значительно больше, и эти упоминания идут в несколько неупорядоченном, случайном порядке в разных разделах документа, относящихся как к предметным областям «Математика и информатика» и «Технологии», так и межпредметным результатам обучения. Вы можете видеть их в части подбора программного обеспечения.

» **OpenOffice.org** для детей — теперь это реальность.



» Умение представлять, анализировать и интерпретировать данные — это задача, решаемая с помощью пакетов программирования, текстовых и графических редакторов, пакетов видео- и аудиообработки. Поэтому ее надо рассматривать в комплексе с другими результатами обучения.

» Владение основами пространственного воображения — по сути, здесь надо рассматривать какой-нибудь 3D-редактор. Наиболее простым под Linux является редактор 3D-графики *Plopp*, который под Linux (в чем и состоит прелесть данной платформы) распространяется бесплатно: его можно скачать с сайта <http://planet-plopp.com/>.

» Умение исследовать, распознавать и изображать геометрические фигуры — для этого подойдут образовательные игры разделов «Математика» и «Исследования» пакета *GCompris* и возможности рисования графического пакета *TuxPaint*.

» Приобретение первоначальных представлений о компьютерной грамотности — идеальным средством будут тренажеры раздела

«В идеале в программу обучения можно ввести игры-головоломки.»

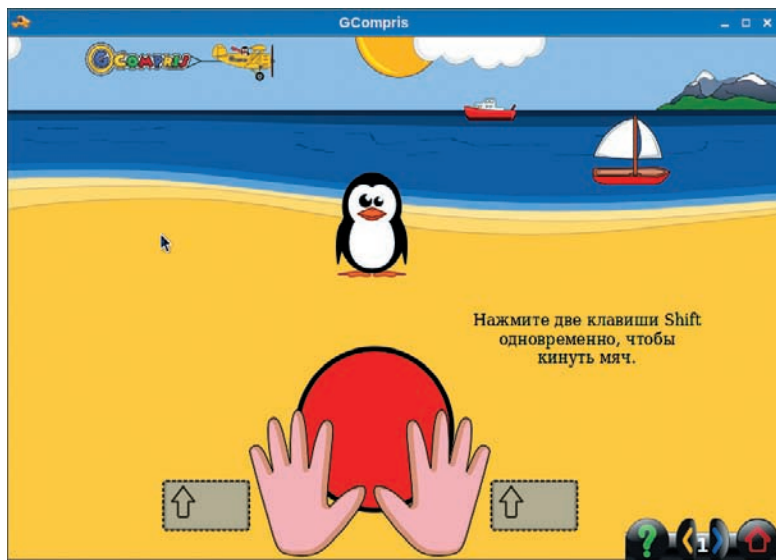
» Владение элементарными практическими умениями и навыками в специфических формах художественной деятельности, базирующихся на ИКТ (цифровая фотография, видеозапись, элементы мультипликации и пр.) — на помощь преподавателю в этом случае придут: в графике — *TuxPaint*, *GIMP*, в анимации — редактор из *GCompris* и *Scratch*, а в работе с аудио-видео — *Kino*, *Audacity* и *OpenShot*. Не забываем про редакторы музыки типа *LMMS* с простым, но мощным интерфейсом.

» Использование знаково-символических средств представления информации для создания моделей изучаемых объектов и процессов, схем решения учебных и практических задач — рисовать схемы

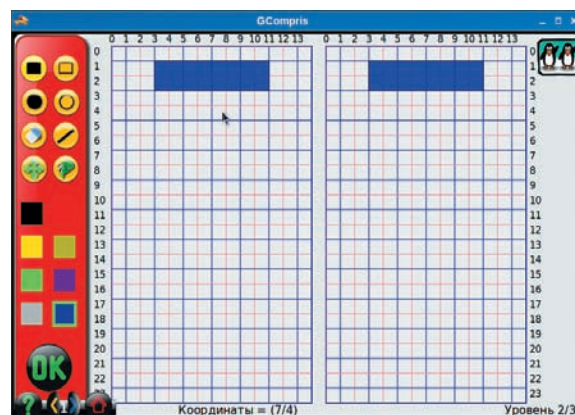
удобнее всего в *OpenOffice.org Draw*, а также в *Dia*.

» Использование различных способов поиска (в справочных источниках и открытом учебном информационном пространстве сети Интернет), сбора, обработки, анализа, организации, передачи и интерпретации информации в соответствии с коммуникативными и познавательными задачами и технологиями учебного предмета — опять же Интернет и браузеры *Firefox*, *Google Chrome* и *Opera*.

» Умение вводить текст при посредстве клавиатуры — здесь на помощь должны прийти (в процессе усложнения программы)



» Основы работы с компьютером для тех, у кого его не было до школы.



» Пытаемся построить похожие фигуры.

Мы про это писали

Как могли заметить некоторые из вас, мы практически не сообщали каких-то новых, неизвестных ранее сведений, а по большей части собрали и интерпретировали уже рассмотренные вещи применительно к данной теме. В рубрике Школа LXF, а также на других страницах журнала, уже поднимались вопросы о программных средствах и средах для учащихся младшего возраста, и вам стоит обратиться к этим статьям, найдя их на нашей вики (<http://wiki.linuxformat.ru>) или в подшивках за прошлые годы (<http://www.linuxformat.ru/archive.phtml>), а также в школьной рубрике журнала *Linux Format* соответствующих номеров этого года.

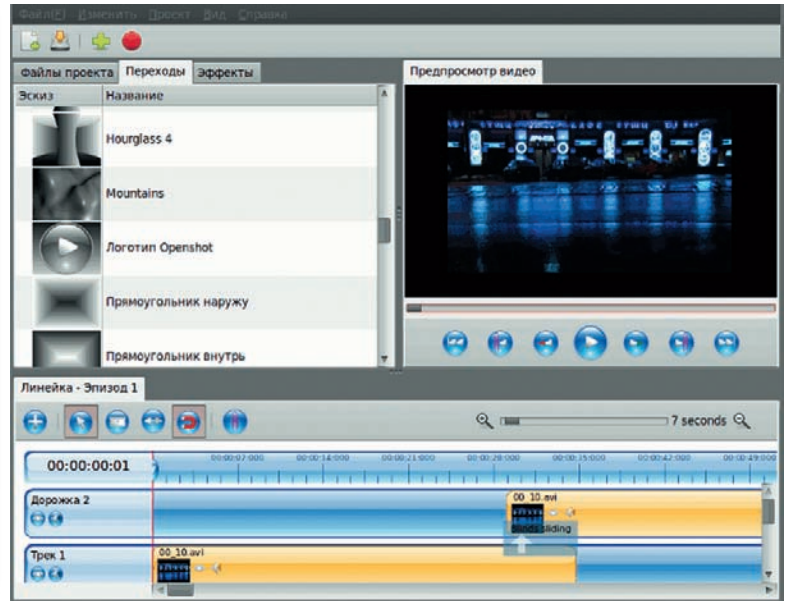
Итак, что вы можете найти по данной теме:

- » LXF83 Ubuntu. Рабочий стол для детей
- » LXF102 Вторая жизнь старого ПК
- » LXF106 Графический редактор *Tux Paint*
- » LXF107 Образовательный пакет *GCompris*
- » LXF126—127 Школа LXF: Исполнители и псевдокод
- » LXF128 Обучение через игру
- » LXF132 Arduino и *Scratch* на уроке информатики

возможности встроенного текстового редактора из *GCompris*, текстовые редакторы (*OOo4kids*, *AbiWord*), и для более старших детей – разнообразные клавиатурные тренажеры: *TuxType*, *Klavaro*, *KTouch*.

» Наблюдение, запись, измерение, опыт, сравнение, классификация и др., с получением информации в открытом информационном пространстве. Умение фиксировать (записывать) в цифровой форме измеряемые величины и анализировать изображения, звуки – аудиоредактор *Audacity*, видеоредакторы *Kino*, *Avidemux* и *OpenShot*. Также для записи и фиксации величин можно использовать внешние платы на основе свободной электроники Arduino (так называемую *ArduinoBoard*) в связке с редактором *Scratch*. Наблюдение можно осуществлять на примере моделирования (взяв за основу физический пакет *Phun* или виртуальные планетарии *Stellarum* и *Celestia*), а также используя цифровые веб-камеру или фотоаппарат для фиксации различных событий (тут на помощь может прийти пакет *Cheese*).

» Умение готовить свое выступление и выступать с аудио-, видео- и графическим сопровождением – редактор презентаций



» Этот простой видеоредактор освоит и первоклассник.

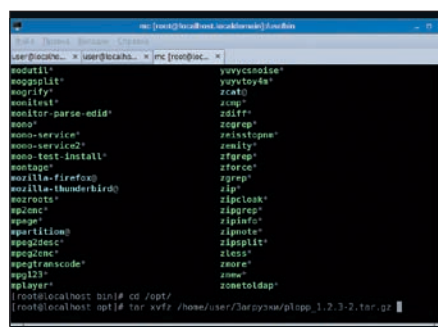
из пакета *OOo4kids*, или (с большими проблемами из-за сложности) *OpenOffice.org Impress*. Также можно создать презентацию в *Scratch*, что в некоторых случаях оказывается более простым делом.

» Умение соблюдать нормы информационной избирательности, этики и этикета – защита на основе контента должна помочь в этом, хотя в большей степени это задача не программного обеспечения.

Мы специально не упомянули в данном списке компьютерные игры. Использовать их или нет, является спорным вопросом: кто-то говорит, что игры класса Картофельного парня (*KTuberling*) или «Расположи три фигуры в ряд» могут помочь в овладении навыками работы на компьютере. С другой стороны, у современных детей дома также есть компьютеры, и они проводят за ними огромное количество своего времени; поэтому это решение остается за преподавателем. В идеале в программу обучения можно ввести игры-головоломки (которых достаточно в пакетах типа *GCompris*), которые помогут в развитии мышления младших школьников. LXF



Шаг за шагом: Устанавливаем Plopp под Linux



1 Скачиваем

Последняя версия программы находится по адресу <http://planet-plopp.com/download/linux.html>; выберите пакет для Debian/Ubuntu или архив для других дистрибутивов.

2 Распаковываем архив

Чтобы программа была доступна всем пользователям, лучше распаковать его в директорию `/opt`. Для этого перейдите в нее, переключитесь на root (или добавьте команду `sudo` в начале) и выполните `tar xvfz /home/путь_к_архиву/plopp_1.2.3-2.tar.gz`

3 Создаем скрипт запуска

Запишем в `/usr/bin` скрипт запуска *Plopp*

```
#!/bin/sh
```

```
/opt/plopp/Plopp.sh
```

и сделаем его исполняемым:

```
chmod +x /usr/bin/Plopp
```

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 5000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор

Валентин Синицын info@linuxformat.ru

Литературный редактор

Елена Толстякова

Переводчики

Илья Аввакумов, Александр Бикмеев, Юлия Дронова, Александр Казанцев, Ольга Кокорева, Светлана Кривошеина, Алексей Опарин, Валентин Развозжаев, Татьяна Цыганова

Редактор диска

Александр Кузьменков

Верстка, допечатная подготовка

Сергей Рогожников

Технический директор

Денис Филиппов

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатели

Виктор Федосеев, Павел Фролов

Отпечатано в ООО «Переплетная фабрика № 1»

196210, Санкт-Петербург, Валентная ул., 11, корп. 2, лит. А
Тел.: (812) 677-98-03
Заказ 3301

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Пол Хадсон [Paul Hudson] paul.hudson@futurenet.com

Редактор обзоров Грэм Моррисон [Graham Morrison]

graham.morrison@futurenet.com

Редактор диска Майк Сондерс [Mike Saunders] mike.saunders@futurenet.com

Художественный редактор Эфраин Эрнандес-Мендоса

[Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

Литературный редактор Эндрю Грегори [Andrew Gregory]

agregory@futurenet.com

Подготовка материалов Джоно Бэкон [Jonno Bacon], Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Энди Ченнел [Andy Chappelle], Алекс Кокс [Alex Cox], Натан Сандерс [Nathan Sanders], Шашанк Шарма [Shashank Sharma], Ник Вейч [Nick Veitch], Евгений Балдин, Семен Есилевский, Александр Казанцев, Евгений Крестников, Александр Маджугин, Павел Протасов, Юлия Синицына, Михаил Смирнов, Виталий Сороко, Алексей Федорчук, Никита Шульгайс, Вячеслав Ястребцев
Художественные ассистенты Стейси Блек [Stacey Black], Фил Хейкрафт [Phil Haycraft], Джон МакЭлистер [John McAllister]

Иллюстрации Крис Хедли [Cris Hedley], Крис Винн [Chris Winn], iStockPhoto

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Tel +44 01225 442244 Email: linuxformat@futurenet.com

РОССИЯ:

Санкт-Петербург (редакция):

Лиговский пр., 50, корп. 15

Тел.: +7 (812) 309-06-86

Представительство в Москве:

Красноказарменная ул., 17, м. «Авиамоторная» (в помещении АТС МЭИ)

Тел/факс: +7 (499) 271-49-54

По вопросам сотрудничества, партнерства, оптовых закупок:

partner@linuxcenter.ru

Авторские права: Статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на CD или DVD-диски, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

LINUX – зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds].

«GNU/Linux» заменяется на «Linux» в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт <http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

<http://www.futureplc.com>

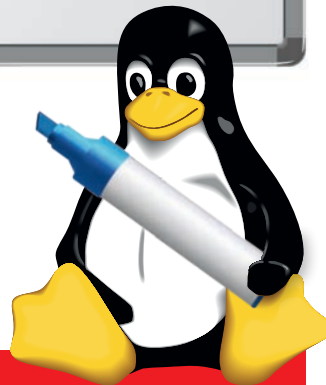
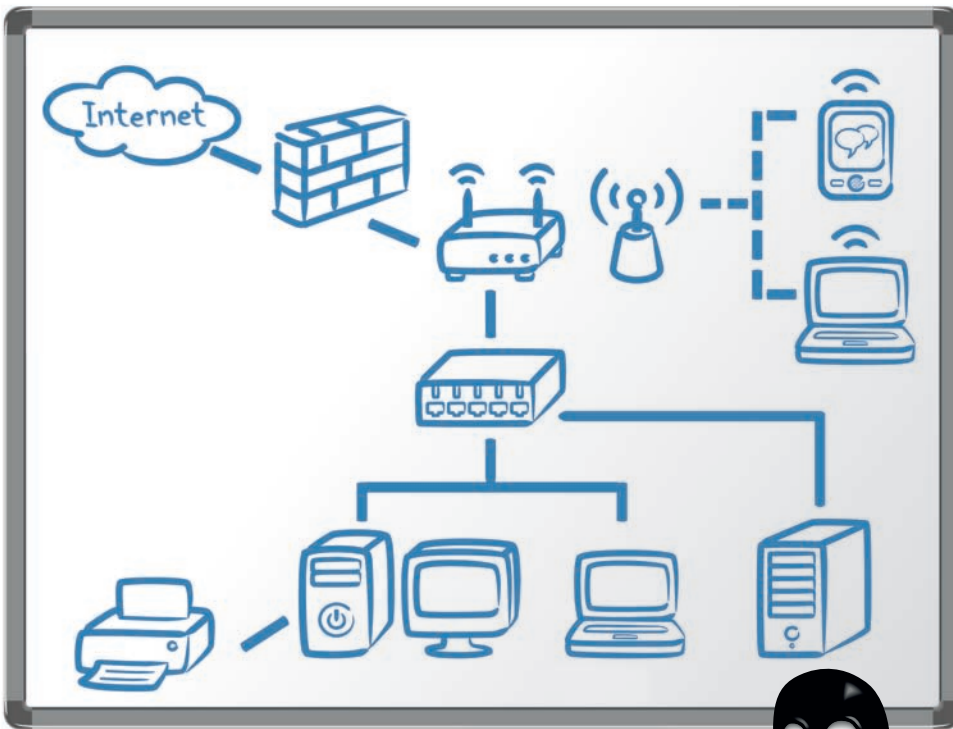
<http://www.futureplc.com>

© Linux Format 2005

© Future Publishing Ltd 2005



BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO



В октябрьском номере

Сети: ЧИНИМ ПРАВИЛЬНО

Если *tracert* показывает звезды, а пинги не проходят, то все, что вам требуется – это грамотное руководство по поиску и устранению неполадок. К счастью, у нас есть как раз такое.

Akademy 2010

Мы ведем репортаж с ежегодной встречи разработчиков KDE и сообщаем, какие решения были приняты на этот раз.

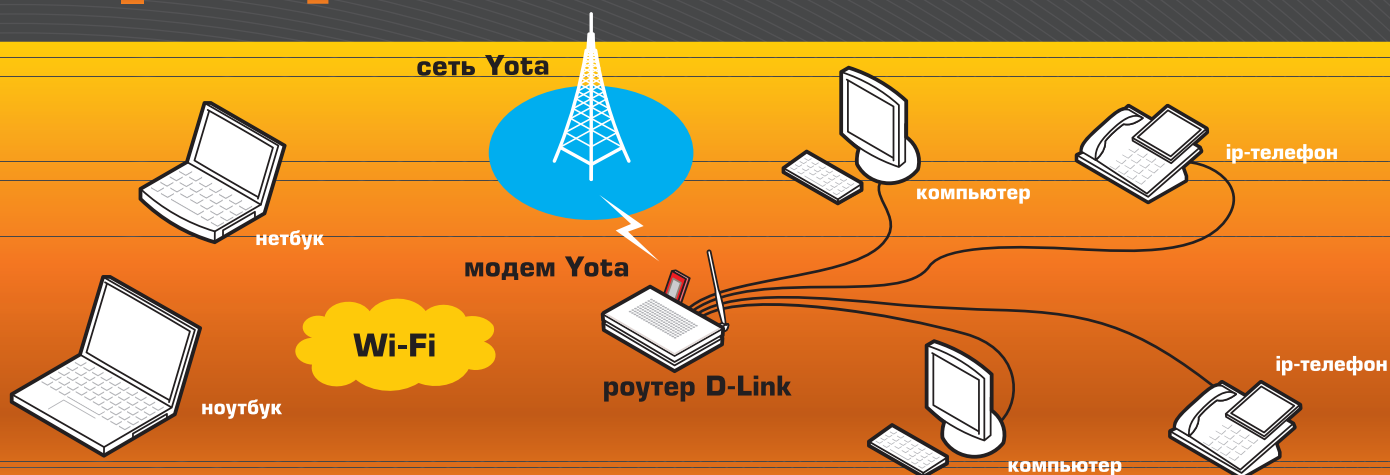
Удаленные рабочие столы

Нужен доступ к рабочему столу компьютера по сети? Мы подберем наилучший инструмент для этих целей.

Содержание последующих выпусков может меняться без уведомления.

WiMaxStore

КОРПОРАТИВНЫМ КЛИЕНТАМ



Компания WiMaxStore рада предоставить вам комплекс услуг по организации доступа к безлимитному беспроводному высокоскоростному интернету Yota WiMAX 4G

- Тестирование скорости интернета (уровня сигнала Yota) в офисе клиента
- Доставка оборудования для организации интернета Yota 4G
- Подключение и настройка оборудования
- Техническая поддержка в течение одного года
- Разработка заказных решений для работы в сетях Yota 4G
- Организация VPN
- Обучение сотрудников клиента

Абонентская плата для юридических лиц, в зависимости от количества абонентов, составляет максимум

1400 руб./мес.

Стоимость указана с учетом НДС.

Интернет-решение организуется на базе роутеров D-Link DIR-320 или ASUS WL 500gP с модемом Samsung SWC-U200.

Оборудование, поставляемое компанией WiMaxStore, имеет все необходимые сертификаты.



 **wimaxstore**

САНКТ-ПЕТЕРБУРГ

+7 812 309-06-86

МОСКВА

+7 499 271-49-54

WWW.WIMAXSTORE.RU



Yota — разработчик и поставщик мобильных сервисов на базе самой современной технологии связи 4G (Mobile WiMAX). Yota — это первая в России сеть Mobile WiMAX (стандарт IEEE 802.16e-2005 на частотах 2,5—2,7 ГГц), развернутая в Москве и Санкт-Петербурге.

HOSTING NEXT LEVEL



Сэкономьте
340 рублей,
как новый
клиент!²

HETZNER
ONLINE
DEDICATED ROOT SERVER
**ЛУЧШЕЕ
ОБОРУДОВАНИЕ!
ЛУЧШИЙ СЕРВИС!
ЛУЧШИЕ ЦЕНЫ!**

HETZNER DEDICATED ROOT SERVER EQ 4

- Intel®Core™ i7-920 Quad-core с поддержкой технологии Hyper-Threading
- 8 GB DDR3 RAM
- 2×750 GB SATA-II HDD (Software-RAID 1)
- Операционная система Linux
- Windows Server от 550 руб. в месяц
- Неограниченный трафик¹
- Восстановление системы
- Установка из образов
- 100 GB пространства для резервных копий
- Без минимального контракта
- Стоимость установки 5900 рублей

1900
рублей в месяц

HETZNER DEDICATED ROOT SERVER EQ 6

- Intel®Core™ i7-920 Quad-core с поддержкой технологии Hyper-Threading
- 12 GB DDR3 RAM
- 2×1500 GB SATA-II HDD (Software-RAID 1)
- Операционная система Linux
- Windows Server от 550 руб. в месяц
- Неограниченный трафик¹
- Восстановление системы
- Установка из образов
- 100 GB пространства для резервных копий
- Без минимального контракта
- Стоимость установки 5900 рублей

2700
рублей в месяц

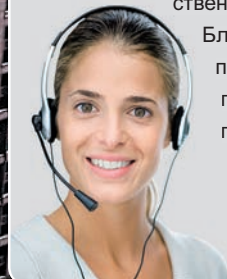
HETZNER DEDICATED ROOT SERVER EQ 8

- Intel®Core™ i7-920 Quad-core с поддержкой технологии Hyper-Threading
- 24 GB DDR3 RAM
- 2×1500 GB SATA-II HDD (Software-RAID 1)
- Операционная система Linux
- Windows Server от 550 руб. в месяц
- Неограниченный трафик¹
- Восстановление системы
- Установка из образов
- 100 GB пространства для резервных копий
- Без минимального контракта
- Стоимость установки 5900 рублей

3300
рублей в месяц

HETZNER ONLINE

Hosting Next Level (Хостинг нового уровня) означает, что компания Hetzner Online готова предоставить вам самые мощные решения для хостинга выделенных серверов из имеющихся сегодня на рынке. Наши предложения были разработаны, чтобы предоставить вам более высокую скорость и чрезвычайно стабильную сетевую инфраструктуру на базе наших собственных дата-центров в Германии. Благодаря лучшим ценам и непревзойденной поддержке, мы превосходим ожидания клиентов по всему миру.



www.hetzner.info
info@hetzner.com

¹ Трафик предоставляется бесплатно. При превышении порога 5000 GB/месяц скорость соединения будет ограничена 10 MBit/s. Стоимость аренды постоянного канала с пропускной способностью 100 MBit/s составляет 290 рублей за каждый дополнительный ТВ.

² Как новый клиент, вы можете сэкономить 340 рублей на первом платеже за любой из рекламируемых здесь продуктов. Просто используйте код ваучера 061109 при совершении заказа. Предложение действительно до 20 октября 2010 года.