

Asterisk » Python » Chromium

LXFDVD

LINUX

FORMAT

Главное в мире Linux

Июль 2009 № 7 (120)

ubuntu

Jaunty Jackalope
и девять других:
прочтите про них
и попробуйте сами

Сними это!

Лучшие программы для записи того,
что происходит на рабочем столе с. 14



Intel вырывается вперед

Nehalem Xeon + vPro + новая
виртуализация = хорошо с. 38

Open CASCADE

Почему в Linux до сих пор
нет аналога AutoCAD? с. 30

Заметки

Вырастим дерево
из ваших мыслей с. 50

VirtualBox и Python

Автоматизируйте управление
виртуальными машинами с. 70

34

страницы
учебников
для всех!



« Хороший коала знает,
как разглядеть
за деревьями лес. »

Марк Шаттлворт про Ubuntu 9.10 с. 27

В каталоге агентства «Роспечать» — подписной индекс 20882
В каталоге «Пресса России» — подписной индекс 87974



Простая в освоении и надежная
операционная система

Mandriva Linux



Mandriva Powerpack

Дистрибутив Mandriva Powerpack предназначен для корпоративных рабочих станций и небольших серверов. Mandriva Powerpack включает все необходимые офисные приложения (офисный пакет OpenOffice.org, интернет-приложения, графические, мультимедийные приложения и др.), а также wine@etersoft для запуска отечественных бизнес-приложений (1С, «Гарант», «Консультант») и серверные приложения. Mandriva Powerpack совместим с серверной версией «1С: Предприятие». Дистрибутив Mandriva 2008 Spring Powerpack сертифицирован ФСТЭК.

Mandriva Flash

Mandriva Flash — дистрибутив GNU/Linux, загружающийся и работающий прямо с USB-носителя объемом 8 Гб. Дистрибутив включает необходимый набор офисных приложений и допускает обновление и установку новых приложений. На USB-носителе достаточно места для хранения данных пользователя, и дистрибутив представляет собой полноценное мобильное рабочее место, работать с которым можно на любом компьютере, поддерживающем загрузку с USB-устройства, не оставляя следов на этом компьютере. Mandriva Flash сертифицирован ФСТЭК.

Mandriva Corporate Server

Дистрибутив Mandriva Corporate Server предназначен для создания корпоративного сервера, на базе него можно создать интернет-сервер, веб-сервер, почтовый, сервер печати, сервер баз данных, сервер приложений и др. Дистрибутив поддерживает стандарт LSB, что делает его совместимым с приложениями, адаптированными для других ведущих мировых дистрибутивов. Mandriva Corporate Server 4 Update 3 сертифицирован ФСТЭК.

Корпоративные продукты Mandriva Linux

Mandriva Directory Server

Mandriva Directory Server (MDS) — это простой в использовании инструмент для централизованного управления учетными записями и конфигурацией таких служб, как электронная почта, прокси сервер, служба доменных имен. MDS является основой для системы управления идентификацией пользователей, разграничения доступа пользователей к интернет ресурсам, квотирования почтовых ящиков, полностью заменяет Microsoft Windows NT4, IIS, m-Daemon, работает с Windows, Linux и Mac.

Linbox Rescue Server

Linbox Rescue Server (LRS) — пакет программ, предоставляющий функции локального и удаленного управления ИТ-инфраструктурой предприятия. Он может быть использован как в одном подразделении, так и в десятках. LRS включает функции инвентаризации программного и аппаратного обеспечения, удаленного управления компьютерами и серверами, а также резервного копирования. Администратор может управлять ИТ-инфраструктурой через веб-браузер с любого локального или удаленного компьютера.

Наименование	Код Linuxcenter.Ru	Цена, руб.
Для рабочей станции		
Mandriva 2009.1 Spring PowerPack (DVD-box, техническая поддержка 12 месяцев)	lc4942	1900
Mandriva 2009.1 Spring PowerPack (DVD-box, техническая поддержка 6 месяцев)	lc4923	1300
Мобильное рабочее место		
Mandriva Flash 2008.1 (Mandriva Linux на 8 Гб USB накопителе)	lc3215	1875
Корпоративные продукты		
LRS Сервер (Linbox Rescue Server, платформы Linux/Windows) (отдельно приобретаются лицензии на рабочие станции)	lc2856	1690
Техническая поддержка на Mandriva Directory Server на 1 год	lc3560	5550
Лицензия и техническая поддержка на Mandriva Linux Powerpack 2009 на 50 компьютеров	lc3342	18 750
Пакет начальной поддержки Linbox Rescue Server	lc2821	99 000
Сертифицированные ФСТЭК продукты		
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack	lc3408	2900
Сертифицированный ФСТЭК Mandriva Flash 2008.1 (на 4GB USB накопителе)	lc3409	4900
Сертифицированный ФСТЭК Mandriva Corporate Server 4.0 Update 3	lc3410	10 050

С вопросами по продуктам Mandriva вы можете обращаться в Mandriva.Ru
Тел.: (812) 309-06-86, (499) 271-49-55
Электронная почта: info@mandriva.ru
www.mandriva.ru

Также в Mandriva.Ru доступны:

- литература по Mandriva Linux;
- услуги по установке и настройке корпоративных продуктов Mandriva.

MANDRIVA LINUX ЯВЛЯЕТСЯ ОДНИМ ИЗ САМЫХ ПОПУЛЯРНЫХ ДИСТРИБУТИВОВ ОПЕРАЦИОННОЙ СИСТЕМЫ GNU/LINUX В МИРЕ.

Главные преимущества дистрибутивов Mandriva — дружелюбный интерфейс, простота настройки, возможность легкого перехода пользователей с Windows на GNU/Linux, совместимость с широким спектром аппаратного обеспечения.

Дистрибутивы Mandriva Linux сертифицированы по требованиям ФСТЭК по 5 классу для СВТ и 4 уровню контроля НДВ, что дает возможность использовать их при обработке конфиденциальной информации и персональных данных граждан.

Множество российских государственных и коммерческих организаций успешно используют Mandriva Linux на серверах и рабочих станциях. Среди них: Правительство Московской области, администрация Черноговского района Приморского края, Министерство финансов республики Саха (Якутия), группа компаний ИМАГ и многие другие.



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

Визит в Canonical заставил нас остро почувствовать свою финансовую неполноценность, и мы решили поинтересоваться у команды LXF: а что бы вы сделали, если бы у вас был не миллион — а 500 миллионов долларов?



Грэм Моррисон

Потратил бы их все в местном магазинчике как средство поднятия экономики Уилтшира.



Майк Сондерс

500 миллионов? На эти деньги можно купить 1153846153 пакета лапши. Мммм, лапшичка...



Нейл Ботвик

Я бы потратил их на создание искусственных роботов, которые решили бы проблемы климата и голода и умели делать чай.



Эфраин Эрнандес-Мендоса

Спроектировал бы моему малышу нормальную кровать. Те, что есть — дрянные, а сон бесценен.



Андрю Григори

А я бы склеил эту кровать из мелких купюр, а потом бы взял с Эффи за нее те самые \$500 млн.



Знди Ченел

Я бы жил на вершине башни в центре Лондона, окруженный акулами с лазерами на головах. И пирогами.



Дэвид Картрайт

Я всегда хотел иметь собственный остров — а Исландия, говорят, сегодня стоит недорого. Ура!



Энди Хадсон

Найму детектива — пусть разберется, кто этот Руди в песне Rudie Can't Fail группы Clash. Что я, мало об этом думал?



Д-р Крис Браун

Я купил бы Рональдо и Лионеля Месси для «Шеффилд Юнайтед», а сдачи хватило бы на пинту приличного пива.



Сюзан Линтон

Пустить их на нестандартные ссуды, затем обернуть в ценные бумаги на \$800 млн под будущую прибыль. Схема испытана!



Шашанк Шарма

Узнал бы, сколько надо Джо Кальзаге, чтобы уйти с ринга. Я бы выстоял против него — он не ударит человека в очках.



Маянк Шарма

Ну, раз никто этого не сказал, я профинансирую разработку альтернативных энергоносителей и лекарства от рака.



Ubuntu – десять!

» Если вы — один из тех, кто пишет нам письма с предложением переименовать «Linux Format» в «Ubuntu Format», мужайтесь: ваши страхи стали реальностью! Вы держите в руках номер, посвященный Ubuntu. На самом деле, конечно, не все так плохо. Мы по-прежнему помним, что в мире есть и другие цвета, кроме коричневого (поэтому припасли для вас яркую Mandriva), и посвятим Ubuntu только один выпуск. Благо, повод имеется более чем достойный: юбилейный десятый релиз Ubuntu 9.04, традиционную LXF-редакцию которого вы также найдете на DVD.

К Ubuntu, как и к любому дистрибутиву, можно относиться по-разному, но нельзя отрицать, что он стал синонимом слова «Linux» в глазах многих людей. Как ему это удалось? Кто-то скажет, что все дело в миллионах Шаттлворта и маркетинге (что лично я могу только приветствовать: в конце концов, Windows XP удерживает лидирующие позиции не из-за технических характеристик), другие вспомнят родословную Debian...

Мне кажется, что ключ к успеху скрыт в слогане Ubuntu: «Linux для людей». Здесь не больше супер-функций, чем в любом другом дистрибутиве, но чувствуется внимание к мелочам. Согласитесь, когда вы вводите команду и видите в ответ сообщение типа «команда недоступна, вам надо установить пакет XYZ» — это приятно.

Ну, а если вам кажется, что Ubuntu монополизирует Linux, не мне вас убеждать. Просто возьмите его исходники и соберите из них достойного конкурента, который будет отличаться от своего родителя не только кодами и цветом обоев.

Валентин Синицын, 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

Web-сайт: www.linuxformat.ru

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

» Телефон редакции: (812) 309-06-86. Дополнительная информация на стр. 104

Содержание

Весь номер — прямо как на ладони: приятного чтения!

Обзоры

Sacred Gold 8

Представьте себе *Diablo*. Потом представьте Linux. А теперь — закройте глаза на оценку в 6/10.

MonoDevelop 2.0 10

Разработчики на C# ликуют сквозь слезы, потому что отладка так до сих пор и не работает.

Calculate Desktop 9.6 11

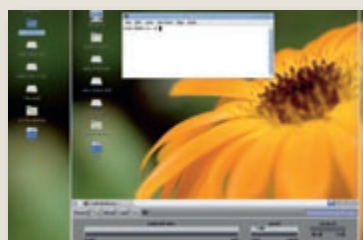
Настольный дистрибутив на базе Gentoo перестает быть оксюмороном, если за дело берутся русские умельцы.



➤ В который раз спрашиваем: сможет ли этот релиз Mandriva потеснить на пьедестале Ubuntu?

Mandriva 2009.1 12

Один из лучших виденных нами настольных дистрибутивов на базе KDE 4 уверенно шагнул из весны в лето.



➤ Мультимедиа-инструментарий Linux — один из лучших в мире.

Сравнение: Скринкастинг

RecordMyDesktop 15

DemoRecorder 16

Pyvnc2swf 17

Xvidcap 17

Istanbul 18

Byzanz 18

Секреты Ubuntu

Узнайте,
как создается
самый популярный
дистрибутив Linux
в мире
с. 20

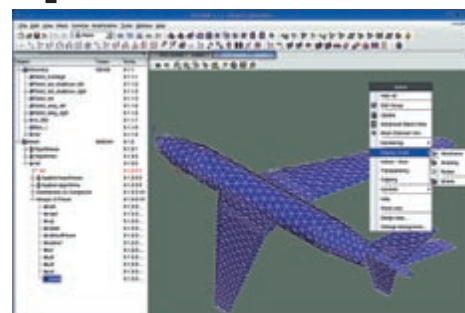
Что за штука...

Chromium

Теперь становится понятно, зачем Google нанял Бена Гуджера с. 42



Open CASCADE с. 30



В Linux нет AutoCAD, но его можно сделать самому.

Люди говорят



“Готов поспорить, что в Сети отыщется даже точная дата крещения «Наболиона Буонапарте».”

Ник Вейч вернулся, чтобы сделать мир чуточку лучше с. 66



Mandriva 2009.1

Для вас, Ubuntu-ненавистники!

- » **Ubuntu 9.04** Традиционная LXF-редакция с расширенным набором пакетов
- » **MonoDevelop 2.0** Если вы – разработчик на платформе .NET
- » **Подшивка Linux Format** Выпуски 107–112 в формате PDF ...и много чего еще! **с. 96**

Ищите в этом номере...

Контроль версий 34

Мы взяли три наиболее популярных системы контроля версий и посмотрели, что отличает их друг от друга.

Intel + Linux = удача! 38

Почему вам следует благодарить движение Free Software за поддержку Core i7 в ядре?

Google Data API 58

По вашим многочисленным просьбам: программные интерфейсы к службам Google.

ПОДПИШИСЬ

на Linux Format сегодня и получи PDF-версию журнала в подарок!

www.linuxformat.ru/subscribe/



Постоянные рубрики

Новости 4

События мира Linux глазами наших экспертов.

Интервью LXF 28

Хотите знать, как пишутся компиляторы?

Что за штука 42

Google все же выпустил операционную систему, но замаскировал ее под web-браузер.

Рубрика сисадмина 44

DHCP и мониторинг нагрузки на сервер.

Ответы 84

Проблемы Linux решены: от создания новых пользователей до авторских прав.

Hotpicks 90

Лучшие новинки открытого ПО на планете.

Диск Linux Format 96

Ubuntu или Mandriva, вот в чем вопрос!

Пропустили номер? 103

Желаете знать, с чего началась заинтересовавшая вас серия статей? Вам сюда!

Через месяц 104

Что ждет вас в LXF121?

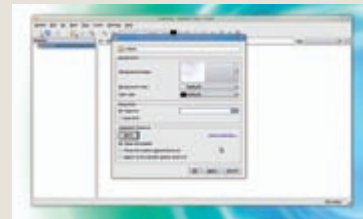


» Наша команда всегда рада подписать вас на LXF: электронная версия в подарок!

Учебники

Начинающим Basket и 00o 50

Храните свои соображения на компьютере, и делайте это красиво.



» **Basket** превратит ваш поток сознания в специальный образ упорядоченные электромагнитные поля.

GIMP Армагеддон 54

Наш Майкл побрился налысо, чтобы сняться в эпизоде с разрушением современного города.

Программирование Google Data API 58

У нас есть все средства для управления данными, хранящимися на серверах Google – поучимся ими пользоваться.

Сети Развернем web-приложение ... 62

Возьмите *Apache*, *PHP* и *MySQL*, настройте их и запускайте, что душе угодно.

Python Мэшап для погоды 66

Если вам нравились наши прошлогодние уроки программирования, вас порадует весть, что мы запускаем новую серию.

VirtualBox Управляй машинами 70

Это не про парковку задом или езду по песку – мы покажем, как автоматизировать жизненный цикл виртуальной машины.

Отладка Знакомьтесь: GNU Debugger ... 72

Компьютеры делают то, что вы сказали, а не то, что вы хотели. Но наши инструменты помогут превратить желаемое в действительное.

JavaFX Теперь графический 76

Применим изученный нами синтаксис к реальной задаче – создадим приложение с графическим пользовательским интерфейсом.

Hardcore Linux Asterisk 80

Подключите свои телефоны к Linux и навсегда избавьтесь от нежелательных звонков.

ГЛАВНОЕ: СПО на уровне государства » Nokia и Qt: плоды сотрудничества » Ubuntu и мобильный Linux » Первый Linux Install Fest в Ставрополе

Федеральные инициативы

» Рубрику ведет
ЕВГЕНИЙ КРЕСТНИКОВ



В последних выпусках **LXF** мы регулярно освещали проблему возврата денег за предустановленную Windows; при этом инициативы властных структур в области свободного ПО как-то отошли на второй план.

В текущем номере мы решили исправить эту оплошность и опубликовать небольшой дайджест последних событий. Федеральное агентство по образованию объявило открытый конкурс, победитель которого будет обучать российских чиновников работе со свободными настольными приложениями (в конкурсной документации упомянуты web-браузер *Mozilla Firefox*, почтовый клиент *Mozilla Thunderbird* и календарь *Sunbird*, офисный пакет *OpenOffice.org*, а также сервис *Google Docs*, хотя не вполне ясно, каким образом он попал в перечень открытого ПО). Цена государственного контракта — девять миллионов рублей. Обучение начнется в августе и завершится в ноябре 2009 года. Предполагается, что через эту программу пройдет около полутора тысяч государственных служащих.

Новость, безусловно, положительная. Однако не все восприняли ее с энтузиазмом. Главный редактор Яндекса Елена Колмановская считает выбор продуктов Google неоправданным: «Если не хочется выбирать из российских сервисов, то почему не остановиться на Hotmail, например?» — заявила она. Лично у меня подобная точка зрения вызывает некоторое недоумение, поскольку российские сервис-провайдеры пока не предлагают решений уровня *Google Docs*. И если в Яндексе недовольны выбором ФАО — пусть для начала задумаются о развитии собственных сервисов. Правда, второй вопрос к составителям программы — открытость и безопасность решений типа *Google Docs* для применения на государственном уровне — от этого не снимается.



» Чиновников обучат работать с популярным свободным ПО.

Интерес властных структур РФ к свободному ПО очевиден, об этом идет много разговоров, но активных действий пока никто не предпринимает. Единичные инициативы, наподобие обучения чиновников или создания «школьного» дистрибутива, не в счет — это достойно и хорошо, но как-то бессистемно. Полагаю, наверху сейчас идет активная борьба, и кто возьмет верх — пока непонятно. СПО в данном случае используется как «козырь» против Microsoft. Кстати, при таком раскладе, логотипом будущего государственного дистрибутива вся Русь вместо пингвина стоит сделать карточный джокер.

Но вернемся к теме возврата денег, потраченных на «налог» Microsoft (уйти от нее никак не удастся):

» ФАС предлагает ввести раздельное таможенное декларирование оборудования и программного обеспечения. По мнению антимонопольной службы, это позволит создать процедуру возврата денежных средств за предустановленные ОС. Соответствующий запрос направлен в таможенную службу. Владимир Кудрявцев зая-

вил: при любом ответе таможни ФАС решит проблему возвратов.

» 4 июня 2009 года ФАС возбудила дело в отношении корпорации Microsoft по признакам нарушения пп. 4.6 ч. 1 ст. 10 ФЗ «О защите конкуренции». Дело рассмотрят 4 июля 2009 года — речь будет идти об установленном доминирующем положении корпорации на исследуемом рынке, а также о маркетинговой политике Microsoft в отношении Windows XP. Одновременно с этим было возбуждено дело в отношении компаний Acer Inc., ASUSTeK Computer Inc., Toshiba Corporation, Hewlett-Packard Company, Samsung Electronics Co., Ltd., Dell Inc. по признакам нарушения п. 5 ч. 1 ст. 11 ФЗ «О защите конкуренции». В публикации на официальном сайте ФАС сообщается, что «по итогам проведенных ФАС России в феврале и марте 2009 года контрольных мероприятий установлено отсутствие у представительств компаний-производителей ноутбуков процедуры возврата конечными пользователями предустановленной неактивированной операционной системы корпорации «Майкрософт» (Windows XP, Windows Vista) с последующим возмещением конечным пользователям ее стоимости». Одним из оснований для возбуждения дела послужило заявление председателя правления РОО «ЦеСТ» В.И. Алксниса. Рассмотрение дела назначено на 15 июля 2009 года.

Тем не менее, чиновники не торопятся продвигать СПО — по крайней мере, поток федеральных новостей на эту тему сильно обмелел. Значит ли это, что в борьбе за государственные контракты наступил перелом, и очередной этап гонки выиграла компания из Редмонда? Поживем — увидим.

Nokia: все идет по плану

После приобретения фирмы Trolltech, компания Nokia часто упоминается в тематических Linux-изданиях. Мы тоже не остались в стороне и уже писали о развитии ситуации вокруг Qt в предыдущих номерах LXF.

Продолжим эту добрую традицию — тем более, что финский концерн приготовил поклонникам популярного инструментария несколько приятных подарков, главным из которых, на мой взгляд, стал общедоступный *Git*-репозиторий *Qt*.

Разумеется, исходные тексты *Qt* были открыты всегда, однако сторонним разработчикам приходилось отправлять исправления по электронной почте или через сервис отслеживания ошибок, а также пересылать факсом подписанный договор о передаче авторских прав на код. Сейчас процедура максимально упрощена — для внесения изменений достаточно зайти на сайт <http://qt.gitorious.org/> и там передать неэксклюзивные права на код в Qt Software. Правда, для формирования заявки автору придется создать клон *Git*-репозитория и сделать *commit* с изменениями.

В Qt Software полагают, что открытие репозитория позволит улучшить обратную связь с независимым сообществом программистов и существенно упростит процедуру включения сторонних исправлений в основную ветвь разработки.

Кроме того, компания Qt Software (напомним, так стала называться Trolltech после приобретения) анонсировала новый проект — Qt Mobility. Из названия понятно, что он ориентирован на разработку ПО для мобильных устройств. В рамках Qt Mobility будут реализованы пакеты кросс-платформенных API для создания мобильных приложений. Шаг вполне логичный — сейчас для каждой платформы, поддерживающей Qt, требуется своя собственная компоновка приложений. Ре-

шить проблему разработчикам поможет новый инструментарий. Первый API, который появится в общедоступном репозитории Qt — Qt Service Framework. Разумеется, проект использует последнюю стабильную версию Qt 4.5.

Есть новости и от самой компании Nokia:

» 27 мая финский гигант сообщил об открытии исходных кодов 3D-видеокодека H.264/MPEG-4 MVC. Кодек оптимизирован для работы на мобильных устройствах, его возможности были продемонстрированы на выставке Nokia World 2008. Специально для показа был создан интернет-планшет

«На рабочих столах Qt прописался уже давно.»

Nokia N800 со стереоскопическим дисплеем. На странице Nokia Research Center доступны: *MVC Encoder* — утилита для создания 3D-видеофайлов и *MVC Decoder* — плеер для платформы Маето.

» Новая модель интернет-планшета Nokia N900 на платформе Маето станет телефоном. Устройство будет работать в сетях GSM 850, 900, 1800 и 1900 Мгц, а также WCDMA 900, 1700 и 2100 Мгц. Кроме того, новинку оснастят модулем Wi-Fi. Начало продаж запланировано на октябрь 2009 года.

» Стартовал oFono — совместный проект корпораций Intel и Nokia. В рамках oFono планируется создать открытую платформу для мобильной GSM/UTMS-телефонии, позволяющую разрабатывать приложения для организации телефонной связи. Программное обеспечение будет распространяться под лицензией GNU GPLv2 и предоставит разработчикам D-BUS API для создания телекоммуникационных приложений.

Что же касается новостей, касающихся применения Qt в настольных системах, здесь наш «улов» будет поскромнее. Стоит отметить только выход *Qt Creator 1.1* в соответствии с ранее опубликованным графиком. Тем не менее — платформа Qt для настольных систем активно развивается, и сейчас главные новости о ней связаны

» Штаб-квартира Nokia: здесь определяют будущее Qt.



Новости короткой строкой

» Иск Red Hat к правительству Швейцарии удовлетворен. Федеральный суд по административным вопросам вынес решение, запрещающее размещать прямые федеральные заказы на программное обеспечение у компании Microsoft.

» Корпорация Intel была оштрафована антимонопольной комиссией Евросоюза на 1 миллиард евро за незаконные методы ведения бизнеса.

» На сайте Минкомсвязи РФ прошло голосование на тему «Что, по вашему мнению, мешает внедрению свободного программного обеспечения (СПО) в России?»

» Benq планирует поставлять настольные моноблочные машины с предустановленным Linpus Linux Lite OS.

» Sun Microsystems и Fujitsu представили новый процессор Venus SPARC64 VIIIx. По приблизительным оценкам, он в 2,5 раза быстрее Nehalem и гораздо экономичнее разработок Intel.

» SourceForge купил социальную сеть Ohloh (www.ohloh.net).

» Исходные тексты системы управления проектами xProcess стали доступны под лицензией GNU Affero Public License.

» Вышла Fedora 11 и Russian Fedora Remix 11. Разработчики планируют удалить Mono из следующего релиза — Fedora 12.

» Проект FreeBSD сообщает о планах по замене компилятора для базовой системы с GCC на LLVM+Clang. Кроме того, во FreeBSD 8-CURRENT значительно переработана звуковая подсистема.

не с Nokia или Qt Software, а скорее со сторонними разработчиками. Речь в основном идет о развитии среды KDE 4 и выпуске новых версий ПО, собранного с использованием Qt 4.5. Оно и понятно — если рынок мобильных устройств Nokia и Qt Software только осваивают, то на рабочих столах Qt «прописался» уже давно, и ждать сюрпризов здесь не стоит.

Ubuntu: еще мобильнее

Выбирая тему для последней заметки, я долго колебался. Хотелось рассказать о новостях мобильного Linux, о новых сервисах Google (впрочем, о них речь пойдет на стр. 58), а также о последних событиях вокруг компании Canonical и дистрибутива Ubuntu.

Учитывая тему номера, пришлось остановиться на последнем варианте — тем более, что многие инициативы Canonical так или иначе связаны с продвижением Ubuntu на рынок мобильных устройств (к которым я отношу и нетбуки).

Традиционно выделим главную новость: компания Canonical представила среду исполнения Android-приложений в Ubuntu. Разумеется, пока что речь идет о прототипе, в котором реализованы далеко не все особенности платформы Google. Я не буду заостряться на технических подробностях (о них уже немало говорилось на тематических порталах). Интересно другое — зачем понадобилось запускать приложения Android в настольном дистрибутиве, в репозитории которого достаточно других свободных программ? По официальной версии, подобная функция пригодится владельцам нетбуков. Согласен; но стоит учесть тот факт, что Canonical объявила о своей инициативе вскоре после того, как корпорация Google озвучила планы продвижения платформы Android на рынок нетбуков. Видимо, сейчас мы имеем дело с банальным перделом рынка. Разработчики ПО активно

продвигают Linux на мобильные устройства, и каждый пытается откусить кусок побольше от этого «пирога». Вспомним Nokia и Qt Software (Maemo и Qt), о которых говорилось выше; кроме них, есть еще Intel с платформой Moblin, а также Google Android и целая куча проектов помельче. Android имеет все шансы занять лидирующее положение на рынке мобильных устройств под управлением Linux. А значит, для платформы Google будет разработано огромное число приложений, и возможность запустить их в Ubuntu поможет Canonical претендовать на свою долю рынка.

Прочих новостей о популярном дистрибутиве GNU/Linux не так много:

» Вышел Linux Mint 7, созданный на основе Ubuntu 9.04.

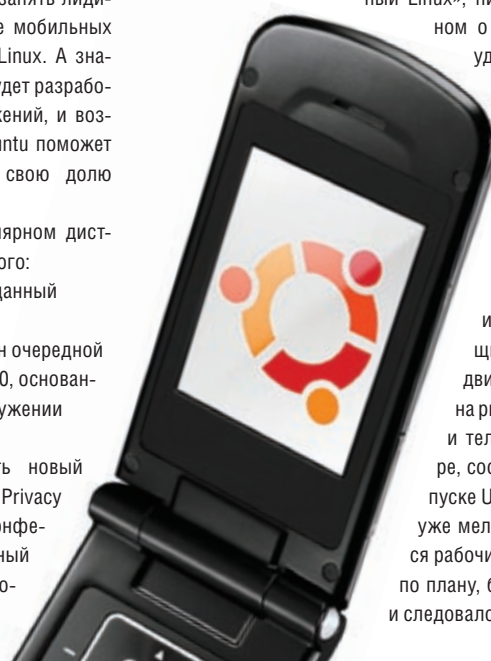
» Пользователям доступен очередной релиз Nexenta Core Platform 2.0, основанный на ядре OpenSolaris и окружении Ubuntu 8.04.

Впрочем, стоит отметить новый проект Canonical — Ubuntu Privacy Remix, представленный на конференции SIGINT 09. Это созданный на базе Ubuntu 9.04 LiveCD, который не предназначен для

установки на жесткий диск и позволяет максимально защитить данные пользователя от злоумышленников. К моменту написания заметки доступна только бета-версия Ubuntu Privacy Remix.

Как можно видеть, несмотря на отсутствие в новостном блоке раздела «Мобильный Linux», писать пришлось в основном о нем. В этом нет ничего удивительного — открытая ОС сейчас активно продвигается на рынок мобильных устройств, и многие компании сосредоточили свои усилия именно в данном направлении. Canonical не стала исключением, и мне кажется, что следующим ее шагом будет продвижение своих продуктов на рынок интернет-планшетов и телефонов. По крайней мере, сообщения об успешном запуске Ubuntu на планшетах Nokia уже мелькали в сети. Что касается рабочих столов — здесь все идет по плану, без сильных рывков, чего и следовало ожидать.

» Ubuntu уже свил гнездо на мобильных телефонах.



Что было: Linux Install Fest / Ставрополь

» Когда 24 мая 2009 года

» Где г. Ставрополь

24 мая 2009 года, впервые в Ставропольском крае, состоялся Linux Install Fest / Ставрополь, целью которого было показать возможности свободного программного обеспечения вообще и операционной системы GNU/Linux в частности. Фестиваль был организован самими пользователями Linux, при поддержке Северо-Кавказской Группы Пользователей Linux (NC LUG) на базе колледжа связи им. В.А. Петрова. Для всех участников мероприятия это стало настоящим событием: организаторы рассчитывали на 100, максимум — 150 человек, но в общем счете посетителей было более 500. В том числе, помимо жителей Ставрополя, приехали участники из других городов: региона Кавказских Минеральных вод, Ростовской области,

Краснодарского края и даже из Дагестана. Среди них были как профессионалы — программисты, системные администраторы, IT-специалисты, технические директора предприятий — так и обычные пользователи ПК, желающие получить основные навыки работы в Linux.

В рамках фестиваля у участников была возможность ознакомиться с Linux, задать вопросы специалистам и бесплатно установить ее себе — для этих целей многие принесли с собой ноутбук или компьютер. Также проводилась бесплатная раздача дисков с заранее записанными дистрибутивами. Кроме того, в программе были предусмотрены мастер-классы по установке и настройке Linux, демонстрации работающих систем и презентации с подробными докладами.

Репортаж с мероприятия был опубликован в газете «Ставропольская правда» и продемон-

стрирован телеканалом «Симпекс-СТС» в программе «Детали КМВ». **LXF**

Репортаж: Виталий Бахтинов (Dumus)

Фото: Дмитрий Карлов



» Мероприятие уровня Linux Install Fest проходило в Ставропольском крае впервые.

LINUX FORMAT Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов



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

Будут ли машины большими?

Сентенцию о том, что некогда машины, как и деревья, были большими, за последние десятилетия не повторял только суперленивый. А давайте спросим себя: нужны ли нам большие машины?

Правда, чтобы ответить на этот вопрос, нам придется заодно спросить себя: а зачем нам вообще машины? Когда-то все мы чего-нибудь да писали. Некоторые, настоящие мужчины, писали даже драйвера для своих устройств. А теперь? Все уже написано до нас.

Мы имеем право быть простыми пользователями. Не заморачиваясь себе голову локалями, драйверами и прочими материями.

И возникает вопрос — а нужны ли нам гигагерцы и гигабайты? Да, нужны. Один раз в жизни — когда мы собираем ядро (если, конечно, это вообще понадобится). Тогда какова потребная для этого мощность? (Не будем забывать, что какой же русский не любит быстрой езды, то есть, пардон, компиляции?)

Отвечаю: ядро Linux в конфигурации по умолчанию на машине с 3-ГГц процессором и 4-ГБ ОЗУ собирается где-то минут 7. На нетбуке с 800-Мгц процессором и 512 МБ памяти — раза в четыре дольше. Принципиально?

А учитывая, что наш нетбук еще и электроэнергию потребляет куда меньше... Как говорили классики, миллиард лет — это очень много. Если понимать и не сдаваться. Но этого миллиарда у нас уже нет. Давайте попробуем сделать, что можем.

alv@posix.ru

Сегодня мы рассматриваем:

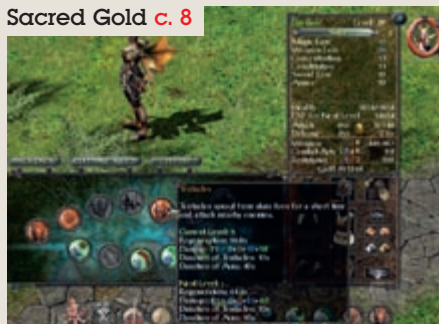
Sacred Gold 8
Авторам этой игры явно не дают покоя лавры *Diablo*. Тот ли это случай, когда копия оказывается лучше (или хотя бы сравнимой) с оригиналом? Все зависит от того, что именно вы цените в развлечениях подобного рода.

MonoDevelop 2.0 10
Объект вождения Пола наконец-то достиг версии 2.0. Если вы никак не могли найти повод, чтобы начать программировать с Mono, это как раз то, что нужно — если заработает. А если нет — попробуйте *Qt Creator* и настоящие языки.

Calculate Desktop 9.6 11
Схема нумерации версий как у Ubuntu, изнутри как Gentoo, по предназначению как Mandriva. Нет, это не чудовище Франкенштейна — это отечественный дистрибутив Linux, и причем весьма удачный.

Mandriva 2009.1 12
Если коричневый, пусть даже с апельсиновыми нотками, все равно не ваш цвет, имеет смысл взглянуть на Mandriva: она сумела совершить с KDE 4 то же самое, что Ubuntu когда-то проделал с Gnome.

Sacred Gold с. 8



➤ Надоело воевать с демонами в консоли? Воюйте с ними на зеленых лугах в мире фэнтези.

Calculate Desktop 9.6 с. 11



➤ С недавнего времени Calculate Linux Desktop обзавелся (в дополнение к рабочему столу KDE) самостоятельной Xfce-версией.

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 «Top 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.



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

Sacred Gold

Подчиняясь логике приключений, **Алекс Кокс** месит бесконечную «зеленку», подбирает тупые топоры и гоняет скрытых в его душе демонов.

Вкратце

» Рабское под-
ражание под-
земельным
бродилкам,
с 3D-надстройкой
над двумерной
картой. См. также:
Diablo под Wine.

Повальным увлечением нынче является *Diablo*; и не хочется, а надо укротить сей бич офисных помещений. *Sacred Gold* [Священное золото] многим обязана этой основополагающей бродилке от Blizzard. До такой степени многим, что Blizzard так и видится со взыскующим долгов паяльником в руках. *Sacred* устроена, как *Diablo*; выглядит, как *Diablo*; уши *Diablo* торчат отовсюду. Чтобы превзойти оригинал, надо свершить чудеса...

Sacred не будет загадкой даже для незнакомых с *Diablo*: это ролевая игра, почти полностью управляемая мышью. Щелчок — и вы продвигаетесь вперед. Щелчок



» Таланты персонажей *Sacred* не сильно различаются. Впрочем, щупальца — это нечто новенькое.

**«Вы спасаете мир,
круша все живое
на своем пути.»**

на монстре — и монстр поражен. Щелкаете правой кнопкой, и получаете особый ход. Со временем вы набираетесь опыта и всяких примочек для своего персонажа — и это почти все, на что способна игра. Всякий раз, входя в обширный, приятный взгляду игровой мир, вы будете делать одно и то же, поэтому *Sacred* — на любителя.

Богатством выбора игра тоже не блещет. Вероятно, сложнее всего приходит-

ся в начале, при выборе персонажа. Вам нравится Гладиатор — гора мышц, которая смачно крушит все на своем пути? А может быть, Гном, бородатый тип чуть поменьше ростом, предпочитающий палубу? Или Серафим — сплошные крылья и когти? Ну, а для разнообразия можно взять Магистра Битвы, волшебника (он, впрочем, тоже чаще колошматит, чем закликает). У всех явная проблема с немотивированной яростью...

Кем вы хотите стать?

Прочие персонажи — опять-таки с разрушительными наклонностями. Кроме традиционной перемены статистики, есть и другие различия: все начинают игру из разных мест (Гладиатор, например, в подземельях под ареной). Однако после краткого «вводного инструктажа» их пути сливаются. Начинается самый обычный квест, и мы не сумели найти каких-либо отклонений от общего принципа роста способностей.



Шаг за шагом: Гоблины для новичков



» Атака

Гоблины весьма назойливы. Они не преминут напасть на вас кучей, пусть вы даже Демон высшего уровня с огромным мечом.



» Отпор

Нет проблем, у вас ведь козырь в рукаве. Достаете сверкающий шар смерти, и наблюдаете массовое побоище.



» Последствия

Не по нутру залитый кровью пейзаж? Расслабьтесь: через пару секунд трупы испарятся сами собой.



» Многие пейзажи *Sacred* поражают красотой — отменная работа художников.

А что же сюжет? Насколько нам удалось разобраться, речь идет о каком-то демоне. Из-за неверно составленной пентграммы он вырвался на волю. И вот, по велению принца Валора, вы спасаете мир Аскарии, перебираясь из точки А в точку В и круша по пути все живое. Заодно вам попадется несколько необязательных квестов (что-нибудь добыть, кого-то успокоить, или все это комплектом). Кроме того, есть расширение *Underworld*, оно вступает в действие по исчерпанию основного сюжета.

Стоит упомянуть систему сбора трофеев, несуразная до смешного. Убиваете трех монстров — двое из них роняют бонусы. Весьма щедрое соотношение, но при таком изобилии вам придется каждые пять минут останавливаться и вытряхивать лишнее «добро» из заплечного мешка. Что еще хуже, название трофея

» Совет: если бежать по прямой, досажая злобным монстрам, они сколотят банду и вас зашибут.



» Дополнительный сюжет начинается прямо у врат преисподней, перед входом в страну гигантских монстров.

и его воплощение практически не связаны между собой: Волшебный боевой топор может быть двуручным или миниатюрным; обоюдоострым или безнадежно тупым.

Ах, эта графика

Надо отдать должное *Sacred*: графика везде превосходная. Двумерные ландшафты прорисованы отменно. Мир Аскарии настолько велик, что и после нескольких ча-

даже после того, как вы специально закрываете их в клетке на ключ и убегаете.

Короче, игра странная, неоригинальная, однообразная и скучная. Ну, или замечательная — кому как. Если вы обожаете расправляться с гоблинами одной левой (кнопкой мыши), то несколько недель беззаботного веселья вам обеспечены. *Sacred*, конечно, не *Diablo*, но чертовски похожая имитация. **1XP**



» В *Sacred* есть чародеи — бродячие торговцы информацией. Этот раскрывает тайны сложных атак.



сов игры на карте остается немало «белых пятен». Похоже, места там даже слишком много: сказать что-нибудь интересное могут только персонажи квеста, поэтому разные локации плавно переходят одна в другую.

А тут еще искажения, возникающие из-за временного зазора между первоначальным релизом *Sacred* и появлением игры для Linux. Системы репутации нет, и можно нагло вторгаться в чужие дома и выносить оттуда столько золота, сколько душе угодно. Можно купить лошадь и ездить на ней, хотя мы так и не поняли, зачем бы это было надо. Иногда к вам привязываются люди, втиснутые в маловразумительный и плохо озвученный сюжет. Они следуют за вами по пятам, и телепортируются

Коллективное безумие

Sacred не чисто одиночная игра. Это, конечно, не *World of Warcraft*, но у вас, по крайней мере, есть шанс выйти в Сеть и разделить игру с друзьями. Linux Game Publishing организовала для этого собственную серверную систему, PenguinPlay (www.penguinplay.com). Платить ничего не нуж-

но; теоретически PenguinPlay обеспечит вам онлайн-игру, например, в *Ballistics* или *Knights and Merchants*. Мы говорим — теоретически, так как на практике пока никто особо не подключается. Поэтому сперва найдите единомышленников, чтобы извлечь из сервиса максимум пользы.

LINUX
 FORMAT

Вердикт

Sacred Gold

Разработчик: Linux Game Publishing
 Сайт: www.linuxgamepublishing.com
 Цена: около 1500 рублей

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

» Тягучий однообразный сюжет с минимальными вариациями. Впрочем, у некоторых любителей ролевых может быть другое мнение...

Рейтинг

6/10

MonoDevelop 2.0



Пол Хадсон считает, что программисты Mono, наконец, получают IDE, достойную их любимой платформы.

Вкратце

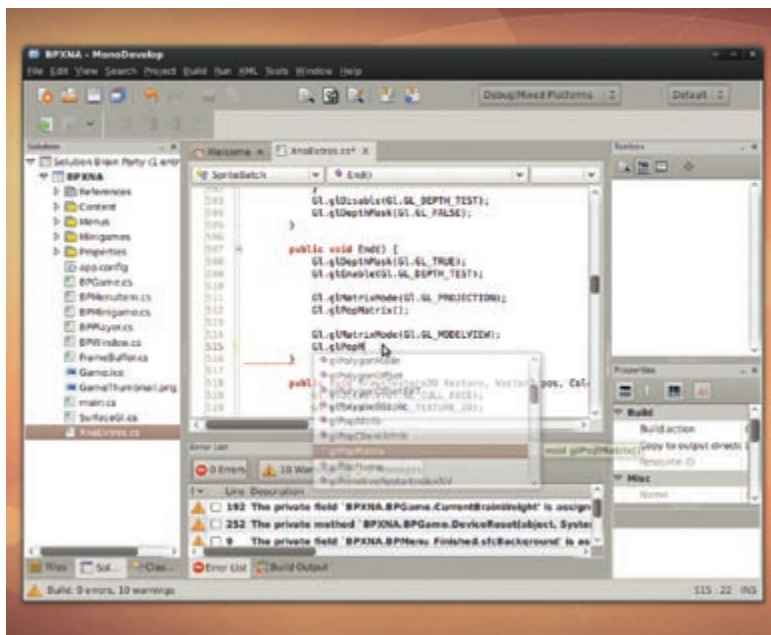
» Лучшая (единственная!) IDE для Mono: стала удобнее, но отладка по-прежнему скверная. См. также: Eclipse и Java.

Отладки для Mono никогда не было (дюжина вызовов `Console.WriteLine()` не в счет).

MonoDevelop 2.0 – попытка устранить эту проблему. Попутно улучшены процессы редактирования и создания GUI и укреплена насыщенная поддержка (ха-ха) цветовых схем.

Приступив к использованию *MonoDevelop 2.0*, вы сразу же заметите массу изменений. Больше нет старых форматов файлов *MonoDevelop* (.mds и .mdp): теперь используется тот же формат, который Microsoft применяет в среде *Visual Studio*, а значит, можно брать Windows-проекты и спокойно работать над ними в *MonoDevelop*.

Но даже утрата обратной совместимости покажется не столь впечатляющей, когда вы начнете кодировать: свертывание кода (можно отметить участки программы и визуально спрятать их за одной строкой), отдельные панели, маркеры изменений в духе *Eclipse* – все это есть, но отдельной похвалы заслуживает завершение кода C# 3.0. Эта функция в *MonoDevelop* действует замечательно: она безошибоч-



» Интерфейс пользователя *MonoDevelop* богат функциями редактирования и вовсе использует композитинг.

Поддержка последних по умолчанию не активируется, что подчеркивает еще одну функцию *MonoDevelop* – выбор целевой версии среды времени выполнения. Это было возможно и прежде (в смысле, вы могли компилировать для .NET 2.0 вместо 1.1). *MonoDevelop 2.0* идет дальше, предлагая уже 3.5, и, самое важное, автоматически определяет несоответствие, когда, заявив поддержку .NET 2.0, вы случайно начинаете использовать конструкции из 3.5.

А это работает?

Новшества радуют: они подводят *MonoDevelop* ближе к уровню, которого он по праву заслуживает. Но вот отладка... хуже некуда. Часто *MonoDevelop* бездействует, пока вы не закроете свое приложение. А если и реагирует, то не поддерживает условные точки останова.

Если отладочный модуль все же соизволит заработать, то все отлично: вы можете пройти код по шагам; при наведении курсора на переменную (даже с неопределенным типом) выводятся подробные сведения; можно также проникать в структуры данных с помощью «разворачивающих» стрелок. Новый режим редактирования хорош. Поддержка C# 3.0 хороша. Тесная взаимосвязь

с *Visual Studio* – тоже хорошо. А то, чего так ждали – отладка – не годится. Может быть, достаточно наложить пару заплат, и все наладится – но куда смотрело бета-тестирование?

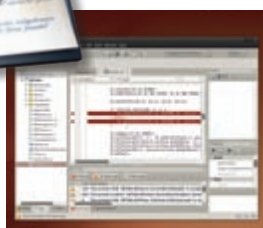
В общем, приобретите *MonoDevelop 2.0* – ведь это ценное обновление единственного нашего Mono-редактора. Найти его нетрудно (он просочился в Ubuntu 9.04 и, видимо, будет в Fedora 11), но отвратительная отладка вас разочарует. **LXF**

«Новшества подводят MonoDevelop ближе к должному уровню».

но сработала во всех наших тестах, даже при нечетком определении переменных ключевым словом `var` и использовании методов-расширений.

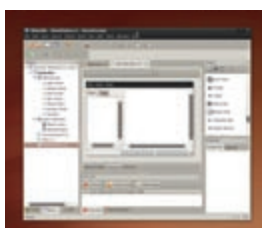


Свойства навскидку



Хромая отладка

Расставьте точки останова, клавишей F5 запустите отладку, и... любуйтесь, как *MonoDevelop* их игнорирует.



Компоновка GUI

Дизайнер пользовательских интерфейсов *MonoDevelop* неплох, но *Glade* все-таки лучше.

LINUX FORMAT Вердикт

MonoDevelop 2.0

Разработчик: Novell
Сайт: www.monodevelop.com
Цена: бесплатно под GPL

Функциональность	8/10
Производительность	7/10
Простота использования	6/10
Оправданность цены	8/10

» Лучшая среда разработки Mono – потому как единственная. Но не без дефектов.

Рейтинг **7/10**

Calculate Linux Desktop

Григорий Рудницкий откопал в закромах родины отечественную альтернативу Sabayon. Установит ли он ее себе на нетбук?

Вкратце

» Отечественный настольный дистрибутив со специализированной серверной версией. См. также: ALT, Mandriva, Runtu.

Дружелюбные дистрибутивы часто делают на базе Ubuntu и Fedora, реже – Mandriva; на основе Gentoo проектов немного, и среди них – Calculate Linux Desktop. Согласно названию, он нацелен на настольные ПК и ноутбуки, и доступен в двух вариантах: с KDE и, с недавних пор, с Xfce. Это самостоятельные сборки, распространяемые как отдельные образы LiveDVD; впрочем, Xfce-версия умещается и на CD.

Готовьте ваши кабели

Загрузка с Live DVD несложная: надо лишь указать в начальном меню язык и разрешение экрана, причем последнее – только если нет уверенности, что видеоадаптер будет распознан корректно.

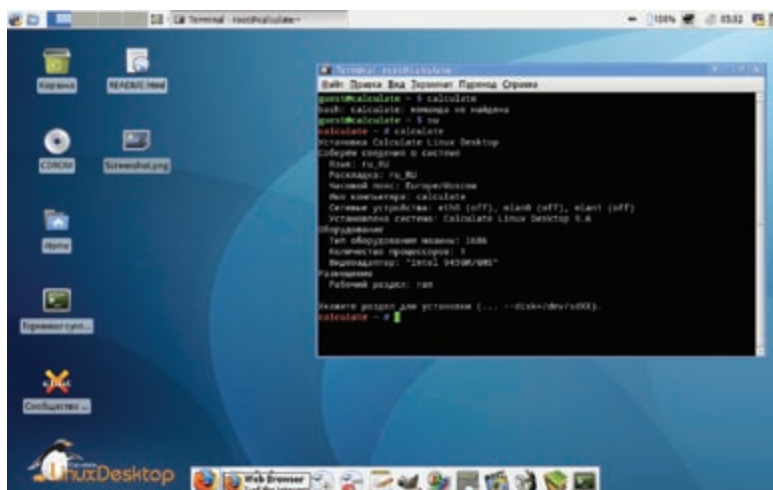
Набор ПО, предлагаемый «из коробки», весьма невелик и покрывает лишь базовый диапазон задач: для Интернета предлагается браузер Firefox, для электронной почты – клиент Claws Mail, для офисных документов – OpenOffice.org 3.0.1. Разумеется, потом можно добрать все необходимое.

«Интересна реализация механизма обновления CLD.»

Заметно отсутствие поддержки ряда беспроводных сетевых карт: так, не обнаружилось внешнее устройство от Netgear и встроенное от Broadcom. На официальном сайте разработчики заявляют, что протестировали адаптеры Intel PRO/Wireless и Realtek RTL8187, а также точки доступа Zyxel P-330W и D-Link DI-824VUP+ – возможно, обладателям этих устройств удастся подключиться к беспроводной сети. Зато повезло владельцам софт-модемов: они поддерживаются ядром CLD.

Получите прописку

Установить CLD можно прямо с Live DVD, но не ищите в меню инсталляторов с красивой графической оболочкой, какие есть в Ubuntu или Mandriva: все придется делать в консоли. «Фирменной» программы для разбиения диска нет. Впрочем, fdisk, который разработчики для этого рекомендуют, знаком многим. После разметки диска установите систему, выполнив



» Все необходимые операции, в том числе и установку на жесткий диск, приходится выполнять из консоли.

в терминале команду **calculate** с нужными параметрами.

Интересна реализация механизма обновления. Новая версия выходит каждый месяц. Текущий номер – 9.6, что соответствует году и месяцу выпуска. Если вы установили CLD, а через месяц вышла его новая версия, поместите ее ISO-образ в директорию **/usr/Calculate/share/linux** и скомандуйте **calculate**: все пользовательские настройки уцелеют, а система обновится. И это можно делать сколько угодно раз – опять же командами из консоли. Подробная информация – в прилагаемом Readme-файле. Документация отличная, и есть русская версия: без лишнего технических подробностей, зато с массой полезных практических примеров.

Кому не нравится CLD, но симпатична модель сборки дистрибутива из портежей, может изготовить собственный вариант на его основе. Для этого понадобится уже установленная самая свежая версия CLD, а также два пакета из дистрибутива Gentoo – *portage* и *stage3*. Подробная инструкция по сборке заняла бы отдельную статью; но данный процесс хорошо документирован разработчиками.

На работе

CLD рассчитан не только на домашние компьютеры, но и на корпоративное применение. Его серверный вариант, Calculate Directory Server, способен выступать в качестве главного контроллера доме-

на в Windows-сетях – в таких сетях смогут работать компьютеры под управлением Windows и Linux. Правда, отсутствует специализированная версия Wine@Etersoft (а значит, безболезненный запуск российских бизнес-приложений), но ее можно приобрести и установить отдельно. Как это сделать, тоже сказано в документации.

В целом, CLD вполне стабилен и удобен в работе. Но мы бы не рекомендовали его новичкам в Linux, даже несмотря на неплохую документацию. Его аудитория – опытные пользователи, которые не боятся и любят работать в консоли, и при этом эстетика и удобство для них играют не последнюю роль. **LXF**

LINUX
FORMAT

Вердикт

Calculate Linux Desktop 9.6

Разработчик: Компания Calculate Pack
 Сайт: www.calculate-linux.ru
 Цена: бесплатно

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

» По-своему красивое и функциональное решение, но отнюдь не для начинающих.

Рейтинг

8/10



Mandriva 2009.1

Первая вылазка Mandriva после прошлогоднего тусклого релиза. Маянк Шарма гадает: это скачок на новую ступень или прыжок в никуда?

Вкратце

» Один из старейших дистрибутивов общего назначения, имеющий свободную и проприетарную ветви. См. также Fedora, OpenSUSE и Ubuntu.

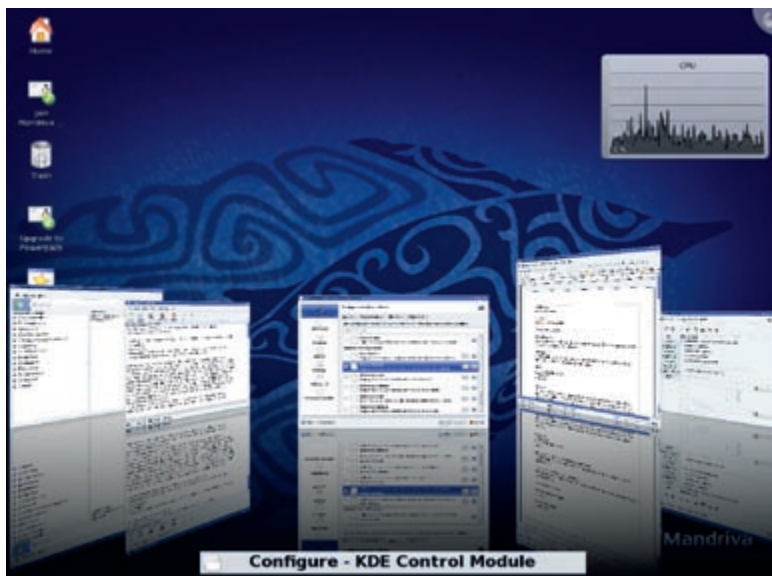
Ключевое ПО

» Ядро 2.6.29
 » X.org 7.4
 » KDE 4.2.2
 » Gnome 2.26
 » Xfce 4.6
 » Установщик кодеков Codeina
 » Qt Creator 1

После выхода предыдущей версии Mandriva подвергнулся изменениям, многие из которых пришлись сообществу против шерсти. Но по части выпуска дистрибутивов общего назначения Mandriva всегда держит марку. Так произошло и на этот раз: Mandriva Spring 2009.1 снова задает тон.

По обычаю, Mandriva подготовила три варианта: One с проприетарными драйверами; Powerpack с проприетарными приложениями; и Free – ничего проприетарного. Free размещается на 4-ГБ ISO-образе, а для One имеется Live CD с возможностью инсталляции. Кроме того, ISO One можно записать на USB и загружать уже с него.

Обновленный установщик научился выводить встроенную документацию с помощью WebKit. Гордость Spring – скорость загрузки, значительно повышенная за счет применения системы Speedboot (первым делом выводится экран входа в систему, остальные службы разворачиваются в фоновом режиме). На нашей тестовой машине Spring загрузился за 26,6 с, тогда как Debian Lenny понадобилось 42,6 с. Ха-



» Mandriva Linux Spring 2009.1 теперь тайно поддерживает файловую систему ext4 – но до чего же эффектно выглядит KDE 4!

равления Mandriva еще более обогатился. Каркас безопасности Mandriva, Msec, обновлен полностью, от графического интерфейса до поддержки подключаемых модулей.

Почувствуйте разницу

Mandriva Spring 2009.1 одним из первых крупных дистрибутивов взяла за основу рабочий стол KDE 4.2. Весь дистрибутив построен вокруг текущей стабильной версии KDE, и нигде ни сучка ни задоринки (а ведь в ходе подготовки обзора мы специально наперебой нажимали все кнопки подряд). Кроме того, в комплекте есть Gnome 2.26, Xfce 4.6 и LXDE – для нетбуков и «пожилых» машин.

Если вы ранее были не знакомы с KDE 4.2, вас поразит стильный оконный менеджер KWin. Он почти так же крут, как Compiz, только не включается по умолчанию. Плазмой Folder View позволяет помещать значки на рабочий стол (стандартное поведение для Spring). Правда, если выйти из этого режима и войти снова, то значки блекнут.

Есть перемены и в рабочей среде Xfce (в частности, возможность выбирать несколько значков на рабочем столе сразу), но есть и связанные с этим проблемы. Например, после включения композитинга рабочий стол становится как бы и не ра-

бочим – пропадает даже переключение между окнами. Новый менеджер сеансов обзавелся кнопками ждущего и спящего режимов – работают безупречно; а вот кнопки выключения и перезагрузки позволяют лишь выйти из KDM, а затем войти назад.

Mandriva всегда извлекает из KDE максимум возможного – впрочем, иного просто не дано. В любом случае, если вы хотите получить солидный дистрибутив Linux со всеми прелестями KDE 4.2, пора действовать. **LXF**

«Весь дистрибутив выстроен вокруг новой версии KDE 4.»

рактические характеристики тестового компьютера: Intel E400 Core 2 Duo, 2 ГГц и 2 ГБ ОЗУ.

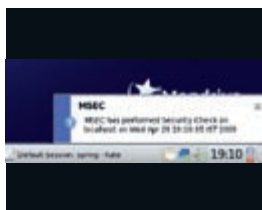
И без того внушительный набор административных инструментов Центра уп-

Свойства навскидку



Speedboot

Благодаря некоторым ухищрениям Spring загружается гораздо быстрее конкурентов.



Упрочение безопасности

Система безопасности Spring значительно переработана: управлять ею проще, а уровень защищенности возрос.

LINUX FORMAT Вердикт

Mandriva Linux Spring 2009.1

Разработчик: Mandriva
 Сайт: www.mandriva.ru
 Цена: бесплатно/1300 руб.

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

» Добротный, стабильный дистрибутив для поклонников KDE. Вот если бы еще Xfce подтянуть...

Рейтинг 8/10

HOSTING NEXT LEVEL

НОВИНКА!

ВЫДЕЛЕННЫЙ СЕРВЕР
HETZNER С ПОЛНЫМ
ДОСТУПОМ ОТ

1900

РУБЛЕЙ В МЕСЯЦ



Сэкономьте 390 рублей как
новый клиент! Укажите в своем
заказе код ваучера: **011107**
(Предложение действительно
до 15 августа 2009)

ВЫДЕЛЕННЫЙ СЕРВЕР
HETZNER DEDICATED EQ 4
С ПОЛНЫМ ДОСТУПОМ

- Intel®Core™ i7-920 Quad-core с поддержкой технологии Hyper-Threading
- 8 GB DDR3 RAM
- 2 x 750 GB SATA-II HDD (Software-RAID 1)
- Различные операционные системы
- Неограниченный трафик*
- Восстановление системы
- Установка из образов
- 100 GB пространства для резервных копий
- Без минимального контракта
- Стоимость установки 5900 рублей

1900

рублей в месяц

ВЫДЕЛЕННЫЙ СЕРВЕР
HETZNER DEDICATED EQ 6
С ПОЛНЫМ ДОСТУПОМ

- Intel®Core™ i7-920 Quad-core с поддержкой технологии Hyper-Threading
- 12 GB DDR3 RAM
- 2 x 1500 GB SATA-II HDD (Software-RAID 1)
- Различные операционные системы
- Неограниченный трафик*
- Восстановление системы
- Установка из образов
- 100 GB пространства для резервных копий
- Без минимального контракта
- Стоимость установки 5900 рублей

2700

рублей в месяц

ВЫДЕЛЕННЫЙ СЕРВЕР
HETZNER DEDICATED EQ 9
С ПОЛНЫМ ДОСТУПОМ

- Intel®Core™ i7-965 Quad-core с поддержкой технологии Hyper-Threading
- 12 GB DDR3 RAM
- 3 x 1500 GB SATA-II HDD (Software-RAID 5)
- Различные операционные системы
- Неограниченный трафик*
- Восстановление системы
- Установка из образов
- 100 GB пространства для резервных копий
- Без минимального контракта
- Стоимость установки 5900 рублей

3900

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

* Трафик предоставляется бесплатно. При превышении порога 2000 GB/месяц скорость соединения будет ограничена 10 MBit/s. Стоимость аренды постоянного канала с пропускной способностью 100 MBit/s составляет 1100 рублей за каждый дополнительный TB.

Сравнение

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

Инструменты скринкастинга

Если картина стоит тысячи слов, то скринкасты отправят на покой текстовые редакторы. В это верит заработавшийся **Маянк Шарма...**



Про наш тест...

Основные факторы оценки приложений скринкастинга — поддержка записи аудио и формат выходного файла. Хотя свободные от патентов кодеки и хороши, они не слишком подходят для загрузки роликов на сайт видеохостинга. Размер выходного файла мы не учитывали, поскольку для разных программ довольно трудно повторить записываемые действия один в один и получить одинаковую продолжительность видео и аудиокomentarиев. По нашим ненаучным наблюдениям, если использовать один кодек для всех приложений, то большой разницы между выходными файлами не наблюдается.

Все программы тестировались на двух компьютерах: на ноутбуке с 1,4-ГГц процессором Celeron и 1-ГБ памятью и на настольной машине с двухъядерным Intel. Приложения запускались и из их графического интерфейса, и из командной строки, чтобы изготовить скринкасты в рабочих средах Gnome и в KDE, а также в более сложных случаях, вроде *Compiz*.

Наш выбор

RecordMy Desktop	c. 15
Demo Recorder	c. 16
Pvnc2swf	c. 17
Xvidcap	c. 17
Istanbul	c. 18
Byzanz	c. 18

Скринкасты — это цифровые ролики-клипы; их сюжет развивается в декорациях рабочего стола, в роли главного героя — мышшь, а голос за кадром поясняет происходящее. Они стали неотъемлемой составляющей процесса электронного обучения и являются основой индустрии компьютерных тренингов. По мере увеличения доступности широкополосных каналов и количества сай-

тов с разделяемым видеоконтентом, пополняется и соответствующий свободный и открытый инструментарий.

Не у всех этих утилит один и тот же принцип фиксации активности на вашем экране. Одни основаны на службах удаленного рабочего стола типа VNC; другие делают с большой частотой ряд экранных снимков и сшивают их в видеопоток; третьи предлагают выбор формата вывода; а четвертые выдают видео только в форматах, свободных от патентов. С их помощью можно заснять всю поверхность рабочего стола или одно конкретное окно. Одни позволяют записывать аудиокomentarии, другие — нет. Но мультимедиа-инст-

рументарий Linux является одним из лучших в мире, и некоторые пользователи предпочитают записывать аудио отдельно с помощью специально предназначенных для этого программ, например, *Audacity*, а затем просто добавив звук к скринкасту с помощью таких чудо-видеоконбайнов, как *FFmpeg*, *MEncoder* или *Transcode*.

Благодаря этим простым в использовании инструментам, почти у каждого популярного проекта с открытым кодом теперь есть и обучающие скринкасты. А еще есть и сайты вроде showmedo.com, где собираются видеоучебники по различным Linux-тематикам.

«Скринкасты стали неотъемлемой частью электронного обучения.»

RecordMyDesktop

Лучшая программа для скринкастинга; ее и за деньги не купишь.

RecordMyDesktop — свободная программа для создания скринкастов, изготавливающая видео в открытом формате OGG. Она написана на C и предоставляет гибкость и контроль как начинающим, так и опытным в деле скринкастинга пользователям.

У RecordMyDesktop есть оболочки, написанные на Python, для Gnome и для KDE, и обе предоставляют предварительный просмотр рабочего стола, где можно выбрать область, которую вы желаете запечатлеть. Также предлагается набор параметров настройки записи.

Скажем, при подготовке демо-ролика графического приложения, содержащего массу виджетов со всплывающими подсказками, можно не включать эти подсказки в ролик. При выборе малой области можно попросить RecordMyDesktop следовать за мышью, что сохраняет размер скринкаста, но отслеживает движение курсора по экрану. А можно вообще исключить изображение курсора из записи.

Настраиваемый захват

По умолчанию запись делается со скоростью 15 кадров в секунду, что подходит для большинства скринкастов, но при желании не устраивающие вас настройки можно изменить.

Также по умолчанию RecordMyDesktop кодирует видео по завершении записи, что требует места для хранения временных файлов. При отсутствии места можно делать запись на лету, что, в свою очередь, требует приличной мощности процессора.

» Если графика KDE и Gnome — не для вас, то есть еще и богатая командная строка.



Еще одна опция RecordMyDesktop, предназначенная для запуска программы на маломощных машинах — отключение сжатия: ценой увеличения затрат дискового пространства уменьшится нагрузка на процессор.

И наконец, еще одна функция — «быстрая» субдискретизация, опять же помогающая разгрузить процессор; но обращаясь к ней как к последнему прибежищу, поскольку видео может получиться размытым. Мы испытывали RecordMyDesktop на ноутбуке с 1,4-Гц процессором Celeron и 1-ГБ памятью и на настольной машине с двухъядерным Intel, и на обоих компьютерах программа работала одинаково хорошо.

RecordMyDesktop может обращаться напрямую к ALSA или звуковой подсистеме OSS, или подключиться к порту записи через звуковой сервер Jack. Можно ука-

зать количество каналов и частоту для кодирования аудиозаписи. По умолчанию, настройки как аудио-, так и видеозаписи установлены на получение наилучшего возможного качества, но планку можно снизить ради сокращения размера получаемого кодированного видео.

LINUX FORMAT Вердикт

RecordMyDesktop

Версия: 0.3.8.1

Сайт: www.recordmydesktop.sourceforge.net

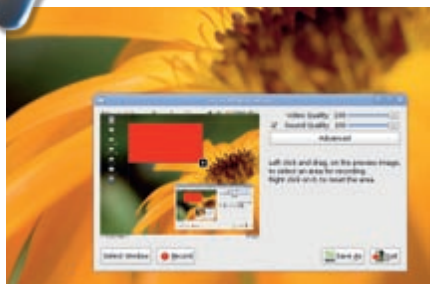
Цена: бесплатно под GPL

» Устойчивое и хорошо управляемое приложение с понятным интерфейсом. Формат получаемых файлов — только OGG.

Рейтинг 8/10



Шаг за шагом: Записываем скринкаст



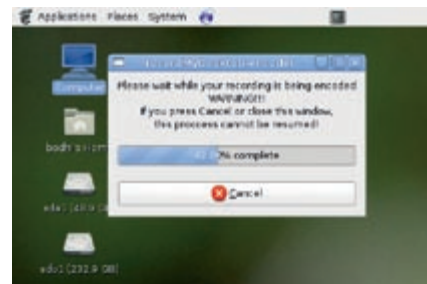
» Выбор области экрана

После запуска программы начните с выбора места действия. Это может быть отдельное окно, или выделенная область, или весь рабочий стол.



» Настройка параметров

Теперь пройдитесь по настройкам, назначая их согласно вашим needs. Отключите запись звука, если планируете добавить аудиокментарий позже. До начала записи надо будет также выбрать имя для вашего файла.



» Начало записи

Жмем кнопку Запись — и поехали! Не забывайте пользоваться горячими клавишами для паузы, если хотите опустить этапы с большим временем выполнения. Записав все нужное, жмем Стоп — и наш скринкаст готов.

DemoRecorder

Отменно быстр, но разве творчество не должно быть свободным?

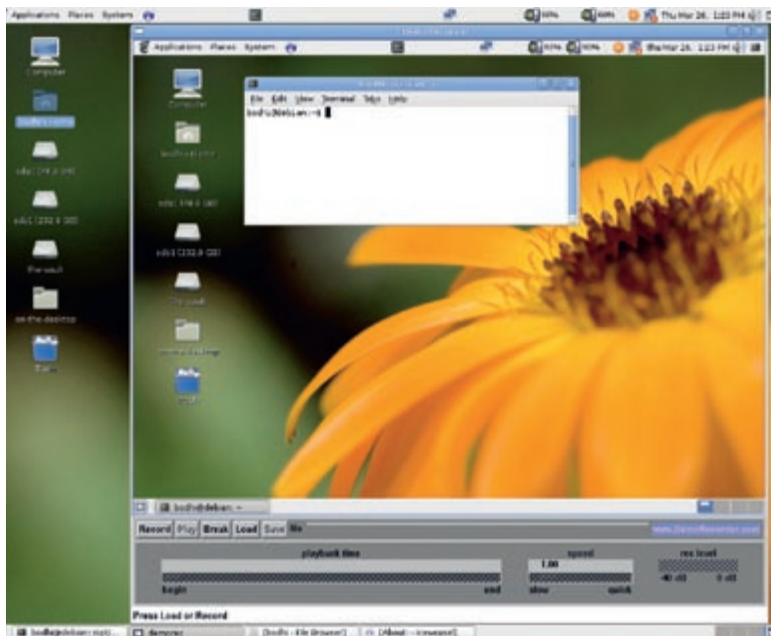
DemoRecorder — проприетарная программа для записи скринкастов, кодирующая видео в проприетарном формате. Возможно, кое-кто неодобрительно качает головой; но, запустив эксперимент с демо-версией, вы подпрыгнете от изумления.

По умолчанию DemoRecorder запускается во вложенном режиме: как виртуальный рабочий стол внутри графического интерфейса плюс кнопки для контроля за процессом. Это отлично подходит тем, кто предпочитает навести порядок на своем рабочем столе перед началом записи. Другой плюс — виртуальный рабочий стол во вложенном режиме умеет работать с меньшим разрешением (по умолчанию — 800 × 600). Как правило, скринкасты и делают при уменьшенном разрешении, поскольку действие сосредоточено в окне или в ограниченной области рабочего стола. Обычно потом приходится подгонять масштаб под стандартное разрешение, искажая видео. Вложенный режим DemoRecorder помогает избежать масштабирования. Если разрешение виртуального рабочего стола по умолчанию слишком мало, можно его увеличить.

Далее, есть полноэкранный режим, где можно выбрать записываемую область на основном рабочем столе. Этот режим предназначен специально для записи трехмерных приложений OpenGL, и в нем без труда сохраняются игры и эффекты Compiz.

Интерфейс вложенного режима также включает воспроизведение видео, записанного в проприетарном формате

» Интерфейс для записи, он же и плеер для воспроизведения готовых скринкастов.



DemoRecorder. Разработчик программы говорит, что ни один из существующих форматов его не устраивал, и он создал собственный, без потерь, который записывает кадры с высокой частотой, не пожирая при этом все циклы процессора и все пространство на диске.

Записывая видео, также можно указать длительность постепенного усиления и затухания звука, чтобы придать скринкасту профессиональный штрих. На медленной машине можно также ограничить ресурсы CPU, доступные программе. Значение по умолчанию — 50 %, что должно оставить достаточно ресурсов для других задач. На двухъядерной машине значение можно смело поднять до 100 %, поскольку DemoRecorder умеет работать только с одним ядром.

Пиршество форматов

В зависимости от версии, DemoRecorder располагает всекими скриптами, которые экспортируют ваши скринкасты не только в популярные форматы FLV, OGG и AVI, но также и в NTSC, PAL, DVD, VOB и т. д. Скрипты запускаются из консоли и имеют больше опций, чем оба режима записи. Ролики можно обработать, уменьшив частоту кадров по умолчанию (25 в секунду) и изменив масштаб видео с 1024 × 768 до 800 × 600 без заметной потери качества. У скриптов экспорта также имеются опции для оптимизации видео под YouTube и Google Video.

На сайте проекта — большое количество документации, где подробно описываются различные функции программы; там же имеются разнообразные настраиваемые сценарии, помогающие поставить производство видео на поток. Разработчик ежемесячно рассылает сообщения, где обсуждаются сложные темы диагностирования и решения проблем узких мест процесса съемки.

Цена за лицензию для DemoRecorder начинается от \$47 за два года использования. Сюда включены техподдержка и исправление ошибок в течение этого периода. Также есть полноценные демо-версии — чтобы разжигать аппетит пользователей, а пре-релиз в статусе бета с новейшими экспериментальными возможностями и исправлением ошибок можно попробовать бесплатно.

Сделай сам

Почти все графические приложения в Linux являются просто оберткой вокруг каких-нибудь мощных консольных утилит. Инструменты для записи скринкастов — не исключение.

Многие из них созданы на базе Ffmpeg — мощной программы, конвертирующей аудио и видео в различные форматы. Она также может «смотреть» потоки X11, что делает ее идеальной для записи скринкастов и преобразования их в распространенные форматы. Команда

```
ffmpeg -f x11grab -r 25 -s 800x600 -i 0.0 /tmp/fftest.mpg
```

захватит ввод X11 и создаст файл MPEG с частотой 25 кадров в секунду и с указанным разрешением.

Другая возможность — использовать import и convert, инструменты в составе ImageMagick. Включенный в скрипт, import может записать ряд экранных снимков, которые затем можно «сшить» воедино с помощью convert или Ffmpeg, получив в итоге скринкаст.

LINUX
FORMAT

Вердикт
DemoRecorder
 Версия: 2.5
 Сайт: www.demorecorder.com
 Цена: \$47—\$247 на два года
 » Квинтэссенция всего необходимого для скринкастинга. Цена — дешево краденого.
Рейтинг 9/10

Pyvnc2swf

Минус и минус не всегда дают плюс. Но они стараются.

Еще до того, как Linux обзавелся чудесным рабочим столом с плазموидами и крутящимися кубиками, которые так и просятся в скринкасты, существовал *Vnc2swf*. Версия этой программы на языке C более не поддерживается, зато поддерживается версия на Python, и она ничуть не хуже.

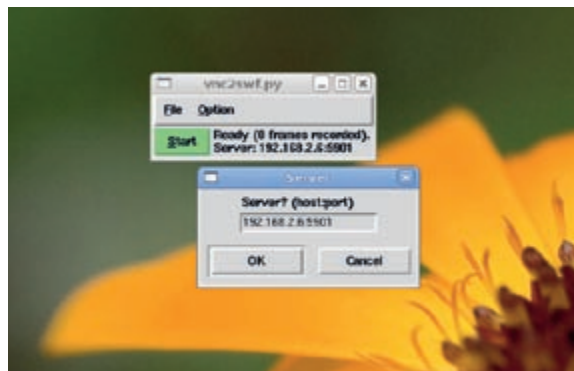
Vnc2swf никогда не гнался за внешним блеском, и у *Pyvnc2swf* тот же стиль. Базовый пользовательский интерфейс есть, но полностью программа контролируется только из командной строки. Скрипт *edit*, через который можно добавить аудиокомментарий в формате MP3 к готовому скринкасту, ужать его размеры или преобразовать в MPEG FLV, работает исключительно в консоли.

Pyvnc2swf умеет записывать экран целиком или ограничивается областью с указанными ему размерами. Размеры окна можно узнать с помощью команды *xwininfo -frame* (ищем числа в строке *-geometry* в ее выводе).

По умолчанию, скорость записи видео – 12 кадров в секунду, но это настраи-

вается. Закончив запись, *Pyvnc2swf* создает файл в формате SWF плюс HTML-файл со встроенным SWF, проигрываемым в бесконечном цикле. Это хорошая отправная точка, если нужно выложить видео на web-сайт. На медленном компьютере вместо создания сразу SWF можно сделать файл *.vncrec*: он быстрее кодируется.

Самый большой плюс программы вытекает из использования VNC для записи: сохранить можно события на любом рабочем столе, доступном через VNC. Чаще всего это будет локальный рабочий стол, но точно так же можно записать и удаленный. Именно поэтому *Pyvnc2swf* запустится и будет работать и в Gnome, и в KDE, и даже в Xfce. Также программа записывает все трехмерные навороты и украшения современного экрана, позволяя делать скринкасты *Compiz*.



» *Pyvnc2swf* вполне способен к записи на удаленном рабочем столе, но область действия выбрать довольно трудно.

«Полностью *Pyvnc2swf* контролируется только из консоли.»

LINUX
FORMAT

Вердикт

Pyvnc2swf

Версия: 0.9.5

Сайт: www.unixuser.org/~euske/vnc2swf/pyvnc2swf.html

Цена: бесплатно под GPL

» *Pyvnc2swf* удаленно работает с захватом кадров, но не звука.

Рейтинг **7/10**

Xvidcap

Падучий, но не упаднический.

Несмотря на неуклюжее название, *Xvidcap* – чудесная утилита, в графическом интерфейсе и консольных опциях которой умещается удивительное количество функций.

Утилита работает в двух режимах. В кадровом режиме отдельные кадры сохраняются как изображения, а затем сшиваются в ролик. Во многокадровом режиме видео кодируется автоматически со значениями по умолчанию. Для предпросмотра скринкаста используется инструмент *animate*, входящий в состав *ImageMagick*.

Xvidcap выводит видео в различные форматы, включая MPEG, AVI, ASF, FLV, MOV, DV и VOB. Также по умолчанию звук записывается в MP3, но можно закодировать его и в Vorbis. Программа обращается к *Ffmpeg*, *Transcode* или *Mencoder*, так что один из них должен быть установлен в вашей системе.

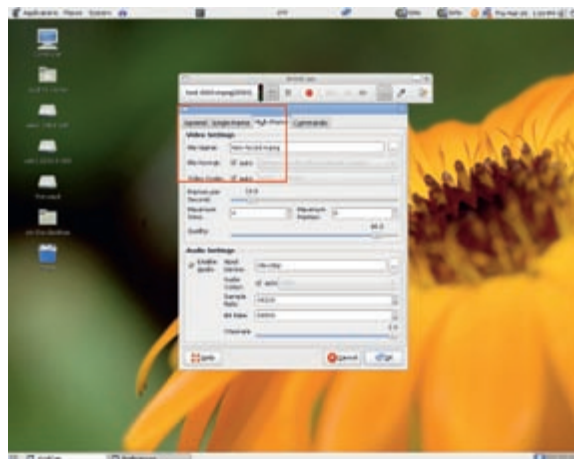
Можно делать полноэкранные ролики, или же выбрать область записи, будь то отдельное окно или любой участок ра-

бочего стола, с помощью мыши. Чтобы скринкаст запечатлел движения по обширной площади, нужно указать меньшую область захвата и заставить *Xvidcap* следовать за курсором.

Запись скринкаста допускает приостановку; это очень кстати, когда задача – например, инсталляция программы – отнимает много времени. Назначить конец скринкаста также можно, указав продолжительность ролика или количество кадров в нем.

Xvidcap работает как в Gnome, так и в KDE, а также без труда улавливает эффекты *Compiz*. Есть очень хороший FAQ, подробная map-страничка и пара видеоинструкций. Единственная проявившаяся проблема – *Xvidcap* случалось падать на обеих наших тестовых машинах.

«*Xvidcap* работает в KDE и в GNOME, и даже в *Compiz*.»



» *Xvidcap* умеет делать видео в кадровом режиме, так что кадры можно подправить в GIMP.

LINUX
FORMAT

Вердикт

Xvidcap

Версия: 1.1.7

Сайт: www.xvidcap.sourceforge.net

Цена: бесплатно под GPL

» Программа выводит видео в различные форматы и умеет записывать и звук, но на данный момент очень нестабильна.

Рейтинг **7/10**

Istanbul

Пора клубу Ливерпуль выиграть еще один Еврокубок.

Довольно долго синонимом скринкастинга в Linux была программа *Istanbul*, и для многих так остается и по сей день. Все потому, что *Istanbul* проста в использовании и выводит видео только в свободный от патентов Ogg-контейнер.

Istanbul унаследовал свои качества захвата экрана от расширений для *GStreamer*. Он сидит в панели задач и позволяет выбрать для записи область рабочего стола или отдельное окно. Звук также можно записывать вместе с видео, для обоих используется формат Ogg. Можно также отключить запись движений курсора, что удобно при съемке действий, управляемых с клавиатуры. Если вы собираетесь добавить аудиокomentarии позднее, запись звука тоже можно отключить. Есть переключатель для записи трехмерных эффектов, и с *Compiz* все работает отлично. После записи скринкаста *Istanbul* выводит диалоговое окно, где можно просмотреть видео перед сохранением.

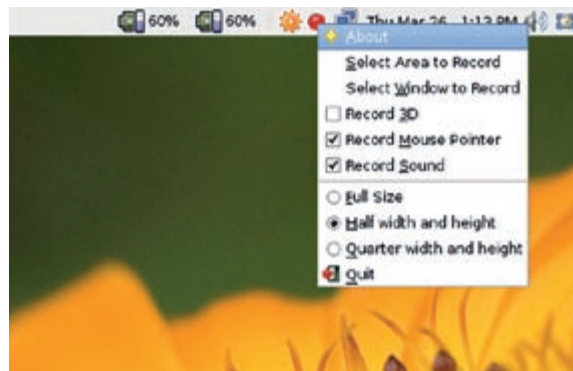
Также есть возможность быстрого масштабирования записанного видео. Можно уменьшить размер в половину

или на четверть, или просто оставить все как есть. Не думайте, что это тривиально: в других программах часто приходится делать такие вещи вручную, поскольку область записи варьируется от полноэкранной до маленьких окон *xterm*.

Возможности изменить частоту смены кадров мы не нашли, и победить «рывки» в наших видео тоже не получилось. Также бывает, что *Istanbul* внезапно виснет по окончании записи, особенно если съемка шла со звуком. Из консоли почти ничего дополнительно сделать нельзя, разве что настроить некоторые из опций *GStreamer*, в основном служащих для отслеживания ошибок, не влияя на саму запись.

Istanbul уже некоторое время не обновляется, хотя разработчик и сказал нам, что в скором времени надеется выпустить новый релиз.

**«Захват экрана
Istanbul унаследовал
от Gstreamer.»**



» *Istanbul* назван в честь пятой победы Ливерпуля в Еврокубке.

LINUX
FORMAT

Вердикт

Istanbul

Версия: 0.2.2
Сайт: <http://live.gnome.org/Istanbul>
Цена: бесплатно под GPL

» Вывод делается в OGG, а параметров настроек, типа выбора количества кадров в секунду, мало.

Рейтинг 6/10

Byzanz

Скринкасты в качестве аватар. Круто?

Кому нужен скринкаст, запускаемый в любом браузере на любой платформе, берите *Byzanz*. Он выдает изображения в формате GIF, где применяется алгоритм сжатия без потерь, так что мы получаем свеженькие скринкасты, пригодные для блога или сайта. Никакое другое приложение не делает меньшие по объему скринкасты для тех же размеров областей. И если вы не придумаете *Byzanz* никакого другого применения, сделайте себе в нем анимированную аватарку.

Параметры можно передавать через консоль или через апплет панели Gnome. Тем, кто не использует Gnome, доступна только консоль, но работать вполне можно, что мы и делали — и в KDE, и в Xfce. По умолчанию записываются также и движения мыши, но при съемке окон *xterm* можно отключить эту функцию.

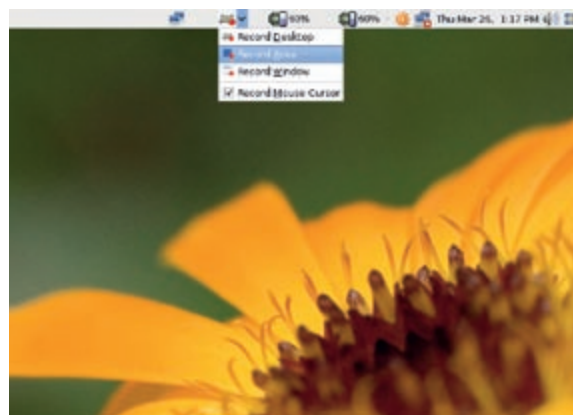
Byzanz записывает действия как на целом рабочем столе, так и в одном окне. Отдельную область также можно выбрать; при этом экран затемняется, а выбранная область окрашивается синим. Если вам

нужна не просто некая область, а нечто более конкретное, это не очень удобно, поскольку нельзя понять, где находятся окна или другие графические элементы на зачерненном экране.

Из консоли можно ограничить длительность записи. По умолчанию она равна 10 секунд. Запись начинается через секунду после старта; ее можно отсрочить, указав задержку в секундах из консоли.

Недостаток *Byzanz* — слабая работа с трехмерными эффектами, так что не ждите здесь хорошей записи эффектов *Compiz*. Кроме того, разработчик не рвется добавлять в программу новые функции, поскольку она уже делает все, что ему нужно. Тем не менее он согласен принимать сторонние дополнения и включать их в код.

«Нет приложений, делающих меньшее видео для данной области.»



» Если *Byzanz* не достигаает до ваших нужд в скринкастинге, сделайте в нем анимированную аватарку.

LINUX
FORMAT

Вердикт

Byzanz

Версия: 0.1.1
Сайт: <http://people.freedesktop.org/~company/byzanz>
Цена: бесплатно под GPL

» Неплохая штука для GIF-скринкастов, когда не нужно звуковое сопровождение.

Рейтинг 7/10

Инструменты скринкастинга

Вердикт

DemoRecorder 9/10

Если у вас случится нужда в скринкасте, то у *DemoRecorder* практически нет конкурентов.

Рассмотрим сперва альтернативы. *Istanbul* когда-то был образцом для подражания в семье утилит для получения скринкастов в Linux – он был легок в использовании и сохранял видео только в форматах, свободных от патентов. Но программа уже давно не обновлялась, и ей не хватает гибкости в настройке.

Дедушка всех инструментов скринкастинга, *Pyvnc2swf*, работает так, как и обещано, и предоставляет отличную гибкость, если только вы согласны записывать аудиокomentarии отдельно. Главное преимущество *Pyvnc2swf* – способность к съемке удаленных рабочих столов.

Две лучших программы с откры-

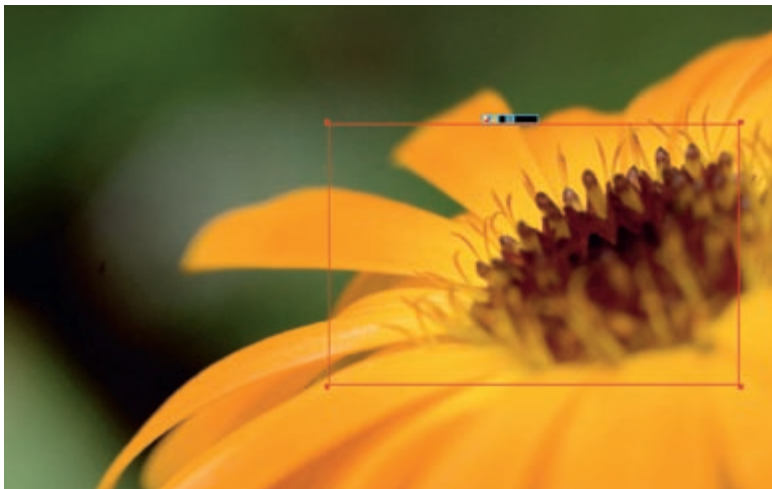
тыми исходными текстами – *Xvidcap* и *RecordMyDesktop*. *Xvidcap* – наименее стабильное приложение в нашем обзоре, а *RecordMyDesktop* – лучший выбор для скринкастинга за бесплатно. Программа обладает огромной гибкостью настроек, но единственный доступный формат файлов – OGG, который придется затем конвертировать в более подходящий для сети с помощью *Ffmpeg* и ему подобных. Кстати о сети: если вам нужен действительно интероперабельный скринкаст, воспроизводимый на всех платформах, то тут

непревзойденный кандидат – *Byzanz* с его GIF-анимацией без потерь.

Кроме *RecordMyDesktop* с интерфейсом, заточенным под KDE, все другие свободные инструменты скринкастинга в нашем обзоре предназначены в первую очередь для Gnome, хотя не хуже работают и в KDE. Но если вам нужно именно KDE-приложение, можно попробовать полузаброшенную *ScreenKast*, использующую VNC. Разработчик программы ищет людей, заинтересованных в ее дальнейшем развитии.

DemoRecorder делает все, что могут свободные программы, плюс много чего еще. Он умеет записывать с удаленных рабочих столов и предусматривает специальный режим для съемки приложений *OpenGL*. Скрипты для экспорта охватили практически все существующие в природе форматы. Большинство проприетарных программ от одиночек, как правило, ничуть не лучше своих свободных альтернатив; но *DemoRecorder* – исключение из правила. Если вы намерены делать скринкасты постоянно, а просматривать их будут с помощью самых различных программ, альтернативы практически нет. **LXF**

«Скрипты экспорта *DemoRecorder* охватили почти все форматы.»



➤ *DemoRecorder* пока не умеет проигрывать видео в режиме *OpenGL*.

Обратная связь

Что для вас важнее: работа независимо от платформы или же свободные от патентов форматы? Предпочитаете ли вы записывать аудиокomentarии отдельно? Присылайте ваши мнения о Сравнении на letters@linuxformat.ru

Таблица функций

Название	Лицензия	Командная строка	Звук	Паузы в записи	3D	Gnome	KDE	Режим полного экрана	Окно	Область	FLV	MPEG	OGG	Запись удаленного рабочего стола
DemoRecorder	Проприетарная	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RecordMyDesktop	GPL	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✗
Pyvnc2swf	GPL	✓	✗	✗	✓	✓	✓	✓	✗	✓	✓	✓	✗	✓
Xvidcap	GPL	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✗	✗
Istanbul	GPL	✗	✓	✗	✓	✓	✓	✗	✓	✓	✗	✗	✓	✗
Byzanz	GPL	✓	✗	✗	✗	✓	✓	✓	✓	✓	✗	✗	✗	✗

ubuntu стукнуло 10

Отмечая десятый релиз дистрибутива Ubuntu, **Грэм Моррисон** исследует все «как» и «почему» его успеха и наследия.



Своими ли достоинствами, грубой ли силой, Ubuntu изменил людские ожидания от дистрибутива Linux, в том числе у людей за пределами сообщества свободного ПО. В некоторых сферах Ubuntu вообще стал синонимом самой операционной системы Linux.

Однако и другие дистрибутивы не стоят истуканами, созирая перемены вокруг них. Они динамичны, гибки и адаптивны. Каждый своим путем, конкуренты среагировали на успех Ubuntu: Debian – увеличением уклона в сторону свободы, Fedora – графическим изяществом, OpenSUSE – активной интероперабельностью, тем самым изменив и наши ожидания. Удивительно то, что Ubuntu ведь не новее или умнее всех. Он просто сумел завладеть воображением людей и отлично справляется с некоторыми мелочами.

На последующих семи страницах мы рассмотрим то, что Ubuntu делает хорошо, и то, как это повлияло на ландшафт дистрибутивов Linux. Мы поговорим с разработчиками дистрибутивов-производных от Ubuntu – Mint, Mythubuntu и Ubuntu Studio, а также с основателем Ubuntu Марком Шаттлвортом [Mark Shuttleworth], и предложим наше собственное видение расцвета Ubuntu. В итоге мы представим анализ развития Linux за последние пять лет, а также посмотрим на него через призму (или куб?) коричневого рабочего стола.



Содержание

Вехи Ubuntu	с. 21
В чем Ubuntu был прав, а в чем нет	с. 22
Обзор Jaunty Jackalope (9.04)	с. 22
Изменение ландшафта дистрибутивов	с. 24
Будущее Ubuntu	с. 26

Дорога к Jaunty

Как мы здесь оказались?

Уместно будет вспомнить, что дистрибутив Ubuntu всего лет пять, как с нами. Стоящая за ним компания Canonical Ltd была основана в конце 2004 для подготовки к первому релизу Ubuntu в октябре того же года. Canonical начала, не имея центрального офиса, с горсткой сотрудников, работавших на дому; сейчас в компании 200 служащих, живущих в 23 странах, а ее штаб-квартира находится в Миллбэнк-Тауэр [престижный небоскреб неподалеку от Вестминстера, — прим. ред.] в Лондоне. В недавнем интервью *New York Times* Марк Шаттлворт заявил, что выручка приближается к \$30 миллион, и Ubuntu уверенно удерживает первую строчку на Distrowatch. Нечего и говорить, что позиции Canonical и Ubuntu сильны в мире Linux.

Дом для Gnome

Но жизнь Ubuntu не всегда была историей успеха. Первый его релиз особого восторга не вызвал: он был девятым в Сравнении дистрибутивов в LXF62, и сравнительно мало повлиял на расстановку чарта Distrowatch в 2004 году. Однако дух перемен носился в воздухе, и ко второй версии, Hoary Hedgehog [Едреный Еж], в апреле 2005, Ubuntu решительно утвердился на генеральной карте. Рабочий стол Gnome получил компаньона в лице официально поддерживаемого Kubuntu, основанного на KDE, что породило идею специализированных версий. Потом будут Edubuntu и Xubuntu, а также с десяток дистрибутивов, официально признанных Canonical. Однако цикл шестимесечных обновлений, синхронизированный с релизами Gnome, продолжался. Это окупилось в следующем релизе, Breezy Badger [Бодрый Барсук]. Он вышел в октябре 2005, со свежим рабочим столом Gnome 2.12, графическим загрузчиком и утилитой установки и удаления приложений, присутствующей в дистрибутиве и поныне.

Первый релиз с долгосрочной поддержкой (Long Term Support, LTS), под кодовым именем Dapper Drake [Сметливый Селезень], выбыл из этой закономерности, выйдя двумя месяцами позже в июне 2006 — вследствие того, что Canonical предложила сопровождение серверной версии до 2011 года, и, таким образом, желала лучше стабилизировать пакеты перед выпуском. Это также сигнализировало о намерениях Canonical вывести Ubuntu за рам-

ки простого дистрибутива с дружественным рабочим столом и выдавало прицел на бизнес-сегмент и, как следствие, прибыльные контракты по сопровождению и поддержке.

Но и настольные системы не были забыты. Dapper стал также первым релизом, где использовался установщик на базе Live CD, поднявший планку для всех остальных дистрибутивов Linux. На первый раз потенциальные пользователи Linux могли попробовать рабочий стол, чтобы увидеть, как он выглядит, а уж потом выделять ему уголок на жестком диске.

Тут компанию покинул Джефф Во [Jeff Waugh], третий сотрудник Canonical и влиятельный сторонник

Gnome: он завел собственный консалтинговый бизнес. Не прошло и месяца, как Джоно Бэкон [Jono Bacon], восходящая звезда LugRadio, учредил в Canonical новую должность, а именно Ubuntu Community Manager; он и занимает ее до сих пор. Теперь он отвечает за поддержку и рост широкого сообщества, которое так помогает пользователям Ubuntu.

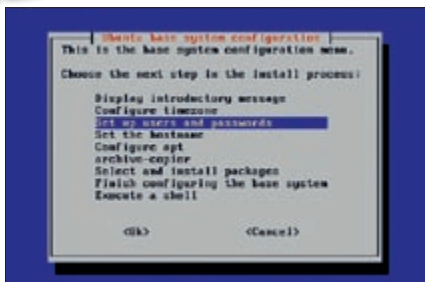
На марше...

В октябре 2006, задержавшись на два месяца после выхода 6.06, был выпущен Edgy Eft [Тропличный Тритон]. Как следует из названия, стабильность его была на грани, и в настольной версии не хватало реальных инноваций. К счастью, 7.04 исправил эти изъяны, добавив утилиту поиска проприетарных драйверов и возможность установить Flash, Java и MP3 через специальное приложение. Обе эти приманки для новичков вызвали конфликты в сообществе, но благодаря им Linux, несомненно, выиграл в простоте — многие ожидают, что их музыка заиграет, не требуя сперва вникнуть, что такое кодек и проприетарный драйвер. Быстрое переключение пользователей и первые проблески плавных переходов Compiz появились в релизе 7.10, а апрель 2008 увидел вторую LTS-версию Ubuntu. С тех пор вышли еще две солидные разработки; релиз этого года, Jaunty Jackalope [Задорный Заяц], оказался лучшим из всех Ubuntu.

«Первый Ubuntu
особого восторга
НЕ ВЫЗВАЛ.»

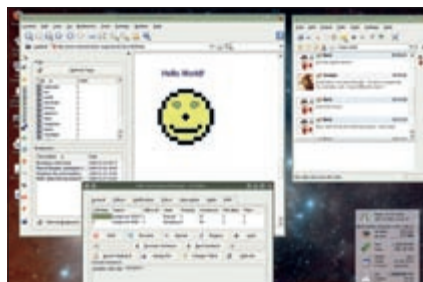


Перемены лица Ubuntu



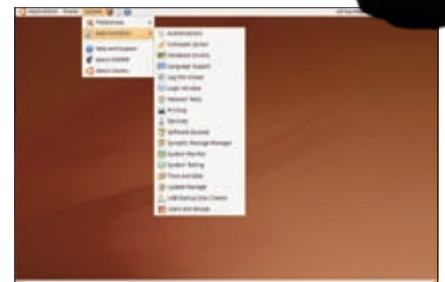
Текстовая установка

Легко забыть, что когда-то в Ubuntu был текстовый инсталлятор, совсем как в Debian. До Live CD было еще шагать и шагать.



KDE

Добавка Kubuntu в конюшню Ubuntu умерило критику за использование Gnome, заодно породив идею респинов.



Jaunty Jackalope

Каждый релиз отличается от предыдущего лишь на несколько шагов, однако после 10 выпусков прогресс по сравнению с первой версией нельзя не заметить.

В чем Ubuntu был прав...

...и в чем неправ. Что ж, дорога к мировому господству ухабиста.

Легко говорить, что Ubuntu преуспел лишь благодаря неограниченным вливаниям от супербогатой родительской компании Canonical. Но ведь Linux не есть обычный комплект приложений. Использовать его никто не обязан, и всегда найдется что-нибудь другое, столь же бесплатное. Ubuntu обречен был сделать верный ход.

Самым крупный и самый ранний успех Ubuntu был в маркетинге. Он стал узнаваемой маркой, и не только в Linux-сообществе, но и в более широком нетехническом мире. Для некоторых людей слово «Ubuntu» стало синонимом Linux, что крайне важно. До прихода Ubuntu этим людям и в голову бы не пришло, что Linux — жизнеспособная альтернатива проприетарным ОС. С самого начала Ubuntu позиционировался как «Linux для людей», а не для стереотипно-асоциальных бородатых технарей из голливудских фильмов. (Также неплохо иметь харизматического космонавта-заводину и средства на бесплатную рассылку CD.) Раньше эти люди, должно быть, представляли нашу ОС с загадочным интерфейсом и ядовито-желтой командной строкой на черном экране; теперь же они поняли, что Linux очень похож на привычные им системы, и даже превосходит их по производительности и безопасности.

Ubuntu Live

Продвижению товарного знака Ubuntu помогли два факта. Во-первых (независимо от нашей периодической критики), несомненно, что коричневая/оранжевая/красная цветовая схема, принятая и в рабочем столе Gnome, и в логотипе дистрибутива, помогла сделать Ubuntu узнаваемым брендом. Рабочий стол излуча-



Мы прошли долгий путь со времен Warty Warthog.

ет ауру заботы, способствуя солидарности новых пользователей и создателей дистрибутива.

Во-вторых, Ubuntu явился вовремя. Релиз Warty Warthog [Бугристый Бородавочник] в 2004 произошел после долгого периода прореживания зубов у Linux, когда многие ПК и ноутбуки страдали от плохой поддержки оборудования ядром. С точки зрения домашнего пользователя, почти все требовало экспериментов, от USB-модемов до видеокарт и принтеров. Но к 2004 году появился хороший шанс, что стандартная установка сразу же заработает, и Ubuntu был на гребне волны дистрибутивов, извлекавших преимущества из новообретенной стабильности. С первого же релиза разработчикам Ubuntu хва-



Присмотримся к Jaunty

LXF обозревает последний релиз

Каждый последующий релиз Ubuntu за последние 18 месяцев был твердым шагом

в правильном направлении. Возьмем, например, Jaunty. Теперь установка почти прозрачна и занимает примерно 20 минут от момента вставки диска до готового рабочего стола. Уж не знаем, почему так долго; зато прежний выбор местоположения наконец сослан на свалку истории, а щелчок по карте теперь выбирает точку, ближайшую к вашему предполагаемому дому, хотя остров Мэн и лезет вперед Лондона.

Блеск загрузки

Через несколько минут вы столкнетесь с хваленым улучшением времени загрузки Jaunty. Мы рады объявить, что оно есть! Раньше наша система добиралась до рабочего стола Gnome больше минуты, а теперь ей хватило около 35 секунд. Попад на рабочий стол, вы видите обычные космети-

ческие изменения. Фон теперь имеет утонченный эффект муара, напоминающий недоразмагнитенный CRT-экран, и шрифты выглядят немного побольше.

Благодаря Gnome 2.26, Jaunty приобрел несколько новых приложений. Главное из них — *Brasero*, программа для записи CD/DVD, наконец достигшая зрелости. Теперь можно создавать диски с данными, аудио и видео, и записывать ISO-образы дистрибутивов с помощью полноценного инструмента, а не устаревшим способом, через папки Burn. Gnome улучшил поддержку нескольких дисплеев, и появилась масса приятных мелочей, столь украсивших работу с Gnome 2.26. И теперь пользователи Ubuntu могут получить *OOo 3.0* и *GIMP 2.6*, не прибегая к неподдерживаемым репозиториям.

Свойства навскидку



Brasero

Наконец-то папки Burn, неизбежные в прошлых релизах, свергнуты более искусной программой записи.



Проприетарное — вам

Куча кодеков и драйверов теперь устанавливается без обязательного добавления новых репозиториях.

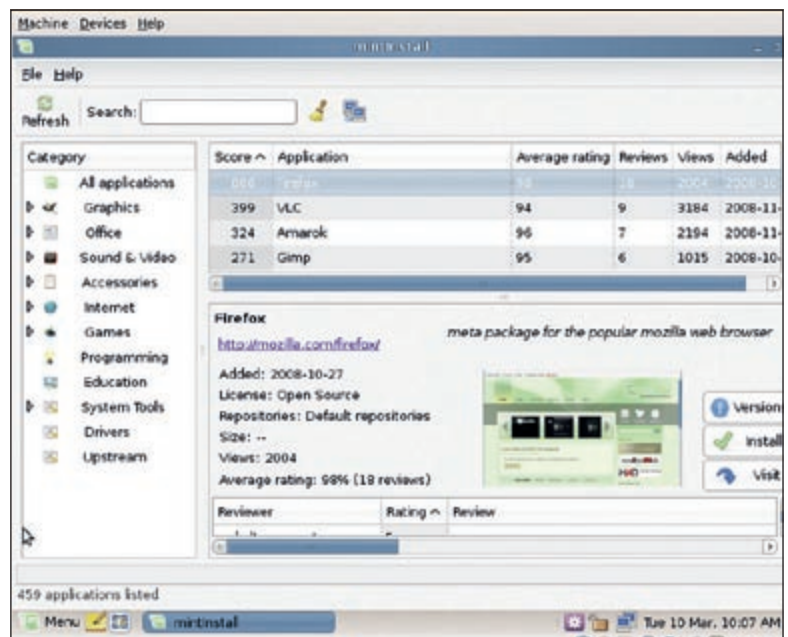
тило ума приналеж на вещи, ключевые с точки зрения обычного человека: прежде всего это поддержка оборудования и простота установки. Ubuntu победил всех одной левой, предложив инсталляцию с LiveCD, которая дала пользователям возможность попробовать водичку перед нырком в полную установку, и сейчас дистрибутивы, не имеющие такой возможности, отходят на обочину. В Ubuntu одними из первых осознали значимость, например, менеджера сети и нового окна «обратного отсчета до выключения», в Ubuntu и Kubuntu, а также системы оповещений в Gnome. Непрерывная череда мелких улучшений приводит к серьезному сдвигу. Не сваливая новшества в общий котел, разработчики Ubuntu оценивают, что важнее, и сначала добиваются, чтобы все работало, даже если при этом приходится опаздывать на новоселье к *ООо 3.0*.

За последние пять лет возникли еще два важных фактора, сработавших на успех Ubuntu. Во-первых, *Mozilla Firefox* принес открытое ПО — или, как минимум, знание о нем — на каждый компьютер. Теперь люди смотрят на Open Source с меньшей подозрительностью и понимают роль Linux как свободной, открытой и безопасной замены их операционных систем. А второе связано с первым: это неудавшийся переход на Vista и дороговизна членства в партии Microsoft, заставившие пользователей компьютеров искать альтернативы. Поскольку Ubuntu — дистрибутив Linux с самым широким «ареалом обитания приложений», оба этих фактора сыграли ему на руку.

Ложь из роз

Но у Ubuntu не все получалось гладко. Главной проблемой стала негативная реакция определенных кругов сообщества Linux, считавших, что Ubuntu не предлагает ничего нового, кроме глянцевого издания Debian и урока маркетинга. Среднему дистрибутиву, живущему за счет сообщества, сложно конкурировать с компаниями, способной позволить себе оплачиваемый персонал по связям с общественностью и даровые отправки CD всем

«Разработчики Ubuntu оценивают, что важнее.»



» Готовность адаптировать своих отпрысков и содействовать им помогла Ubuntu вырасти в глазах сообщества Linux.

желающим. Это выглядело нечестной игрой, и оттолкнуло старую гвардию и широкую часть публики.

Сложно сказать, остались ли позади те дни перепалок и снобизма. За пять лет с момента первого релиза шум из лагеря Debian поутих, но кое-кто продолжает считать, что Canonical доит энтузиазм разработчиков свободного ПО, не внося своего весомого вклада. Конечно, это не идет вразрез с миром GPL, который и разрешает, и даже поощряет такое использование, но Canonical никак не купит хороший пиар в Linux-сообществе.

Все репозитории *APT* сейчас включены по умолчанию, в том числе и Multiverse, и источник проприетарных драйверов. А значит, вы можете скачать несвободное ПО через менеджер пакетов, не преодолевая полосу препятствий, и это сразу же станет очевидным, если у вас графические карты от Nvidia или ATI. Через 30 секунд после первого входа в графический рабочий стол Ubuntu определит, какие проприетарные драйверы способны улучшить производительность вашей системы, и предложит вам добавить их. Для изменений нужны только щелчок мышью и ввод пароля, и это хороший компромисс между распространением несвободного ПО на дисковых образах и запросом разрешения пользователя на установку. То же касается Flash, MP3 и Java: предусмотре-

ны приложения для вызова их инсталлятора. *Firefox*, например, предложит выбрать между установкой *Gnash*, SWF от Gnome или Flash от Adobe. Попробуйте открыть MP3 из *Totem* или *Rhythmbox*, и Ubuntu предложит вам найти и установить подходящий модуль расширения автоматически. Не на 100 % — например, *Brasero* такой диалог не показывает — однако это очень близко к давней мечте многих из нас.

KDE

Для пользователей KDE в состав Kubuntu 9.04 вошла лучшая из виденных нами версий KDE 4.2. За несколько недель рабочий стол доказал, что он совершенно стабилен, пусть даже некоторые плазмоиды ведут себя плоховато. Как ни странно, по умолчанию установлен не *Firefox*, хотя

web-браузер *Konqueror* уже ощущает свой преклонный возраст; да и установщик *KPackage* не столь интуитивен, как *Synaptic* или даже *Adept*.

Ubuntu, быть может, полнится изменениями постепенно, но с широкой точки зрения он все более становится мерилом для дистрибутивов Linux, с которым сравнивают остальных.

Вердикт

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

» Лучший Ubuntu всех времен.

Рейтинг 9/10

Смена ландшафта

До Ubuntu не было ни установок с Live CD, ни Wubi, ни коричневого цвета.

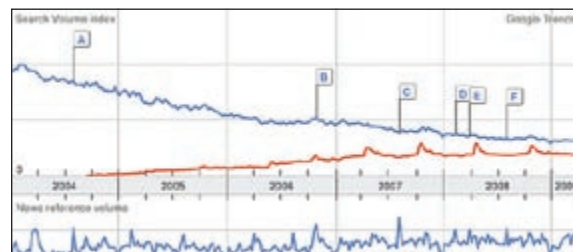
Благодаря уверенной и четкой рекламной кампании и узнаваемому харизматичному главе, Ubuntu легко вписался в модель операционной системы, применяемую Windows и OS X, которые также имеют ярких харизматичных лидеров. Это льет воду на мельницу популярных СМИ, далеких от технологий, и есть немало доказательств тому, что взлет Ubuntu привел к взлету Linux в целом.

Чтобы осознать место Ubuntu на ландшафте Linux, представим, как этот ландшафт выглядел до 20 октября 2004 года — дня, когда Ubuntu впервые попал на сайт Distrowatch. Как показывает чарт Distrowatch, в тот день на вершине горы находился Mandrake Linux.

Закат Mandrake

Mandrake, предшественник Mandriva Linux, был сильным соперником, и у него было немало общего с Ubuntu. Он был и остается дистрибутивом, разработанным для типового пользователя компьютера. Именно Mandrake привел многих в мир Linux, благодаря простой процедуре установки, почти что лучшей поддержке оборудования и периферии и фантастическому менеджеру пакетов, позаимствованному многое из RPM от Red Hat, но обладающему собственными средствами для разрешения зависимостей. В результате получалась колоссальная библиотека доступных пакетов и стабильный рабочий стол Linux, который мог установить почти кто угодно на самом разнообразном оборудовании.

«У Mandrake Linux было немало общего с Ubuntu.»



➤ Согласно тенденциям в поисковых запросах Google, первые несколько лет Ubuntu перекачивал пользователей от Mandriva.

Через месяц после релиза Ubuntu поднялся до 12-го места в чарте Distrowatch. На новый год он уже был девятым. А когда в наши офисы ворвался Hoary Hedgehog (апрель 2005), Ubuntu уже достиг вершины горы, опустив Mandrake Linux на второе место. Это не простое совпадение — Ubuntu победил за счет Mandrake, так как оба дистрибутива построены по одному и тому же рецепту. Комбинация легкой установки и нетехнарской среды рабочего стола, обогащенная беспрецедентным количеством пакетов, сделала как Ubuntu, так и Mandrake первым портом приписки и для новых пользователей Linux, и для тех, кто устал менять дистрибу-



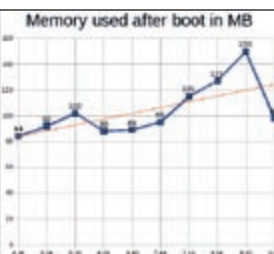
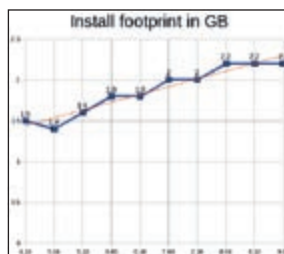
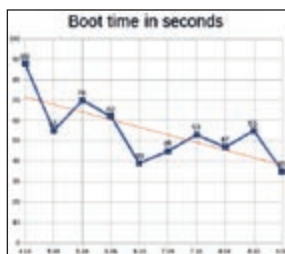
Производительность

Бесплатное обновление вашего компьютера — просто за счет ОС

Многие из нас помешаны на производительности своих систем. Но ведь компьютерам теперь хватает скорости для просмотра клипов на YouTube и чтения Slashdot, чего же еще надо?

Ну, конкуренция-то поджигает, и преимущества быстрого, простого компьютера — например, нетбука — показывают, что скорость и эффективность никогда еще не были так важны. Последний релиз Ubuntu серьезно повысил ставки: сейчас это дистрибутив с самой быстрой загрузкой из всех опробованных нами. Время старта на нашем тестовом Intel Core Duo 2 снизилось примерно на 43 % и занимает около 35 секунд, от нажатия кнопки Power до графического рабочего стола. Кроме оптимизации загрузки, очевидно облегчение ряда приложений, так как по сравнению с предыдущим релизом Ubuntu в стандартном рабочем столе Gnome высвободилось около 50 МБ оперативной памяти.

Внедрение новейшей файловой системы ext4 кое-кто принял скептически: ведь она может привести к потере данных, если во время записи файла



➤ Количество пакетов стало больше, а свободная память и время загрузки — лучше.

на диск вдруг отключится питание. Мы протестировали ее метода с помощью виртуальной машины, и Jaunty не испытывал никаких проблем. Однако мы пока не взялись бы использовать ext4 на производстве; потому-то в стандартной установке Jaunty ext4 и не задействована. Кто захочет перейти на ext4, должен будет указать это самостоятельно во время установки, на странице разбиения диска вручную. Если вы переходите на Jaunty с предыдущей версии Ubuntu, заодно позаботьтесь об обновлении

загрузчика Grub, так как с разделами ext4 будет работать только самая последняя версия.

На некритичных системах, на наш взгляд, стоит рискнуть и использовать новую технологию, так как ext4, похоже, дает 25 % улучшения производительности файловой системы. Даже если вы опустите это в нынешнем релизе, после улаживания проблем со стабильностью внедрение ext4 даст эффект бесплатной модернизации вашего компьютера.

тивы. При этом обе системы имели завидное сообщество, способное решить большинство проблем новичков, не дожидаясь официальной службы поддержки. Каждое из этих сообществ было относительно доступным и дружелюбным, по сравнению с другими дистрибутивами.

С трудом избежав банкротства по Статье 11 в 2004 году, Mandrake ударился в метаморфозы. За шестимесячный период между релизами Warty и Hedgehog компания Mandrake слилась с бразильской фирмой Conectiva, производителем одноименного дистрибутива. Вполне извинительно, что ради избавления от финансовых неурядиц новая Mandriva встала на коммерческие рельсы. Это помогло протянуть следующие нелегкие пять лет релизов, но до сих пор неясно, хватит ли Mandriva сил на следующие пять лет, и в компании недавно произошли сокращения. Ubuntu, выживание которого напрямую не связано с прибыльностью, отвоевал себе часть пространства, некогда занято Mandriva.

Клеман Лефевр [Clement Lefebvre], разработчик Linux Mint, считает, что есть другая веская причина успеха Ubuntu. «Он пришел с лозунгом “Linux для людей”, — сказал он нам. — Земные цвета, идея обмена, африканская родина, вот концепция Ubuntu. Я считаю, что это важно само по себе. Философия проекта привлекала многих».

Луис де Бетанкур-Гимера [Luis de Bethencourt Guimerá], ведущий разработчик Ubuntu Studio, соглашается: «Ubuntu распространил слово свободного ПО на новую пользовательскую базу. Более типовую толпу, не из одних технарей». Он продолжает: «Ubuntu должен быть символом Linux, но не более того. Одна из лучших сторон свободного ПО — свобода выбора».

Пример заразителен

Взглянув на Google Trends, можно четко увидеть, что в 2004 году подъем Ubuntu произошел за счет Mandriva. Очевидным спадом на графике Mandriva отвечают подъемы графика Ubuntu. Конечно, это графики количества людей, которые искали данные термины, а не использовали операционные системы, но все-таки по ним видно, куда ветер дует.

График показывает, что Mandriva притормозилась в середине 2005 года, после чего на протяжении последних четырех лет наблюдается медленный, но постоянный спад. Эта часть графика имеет гораздо больше общего с графиками для Fedora и SUSE Linux, которые демонстрируют одинаково медленный и растянутый спад в использовании в тот же период, а не первоначальное



➤ Mandriva (урожденный Mandrake) — это дистрибутив Linux примерно с той же пользовательской базой, что и Ubuntu..

Персональные архивы пакетов

Если у вас есть склонность к экспериментальному ПО, но не хватает времени или мастерства поддерживать собственную ветку разработки вашего любимого приложения, вам нужен Personal Package Archive (PPA). Для пользователей Ubuntu это способ получения новых пакетов, в стабильности которых разработчики еще не уверены настолько, чтобы влить их в основные репозитории. Пакеты собираются автоматически на сервере, и после запуска проекта разработчику нужно только передать свои изменения в исходном коде, а остальное проделает PPA. Добавив соответствующие PPA в ваш менеджер пакетов, вы сможете скачивать обновления тем же способом, как прочие обновления Ubuntu.

Хорошим примером того, что можно сделать с помощью PPA, является ветка KDE 4.2 (неон), поддерживаемая в PPA в ходе разработки Jaunty. Это значит, что любители KDE, использовавшие более старую версию Ubuntu, могли обновиться до нового рабочего стола че-



➤ Экспериментальные приложения типа Chromium можно установить, добавив репозиторий PPA в менеджер пакетов.

рез менеджера пакетов, одновременно обеспечивая обратную связь для разработчиков. Проблемы с пакетами и стабильностью случались, но конечный результат вылился в лучшее качество KDE 4.2 в релизе 9.04, то есть выиграли и пользователи, и разработчики.

резкое падение, вызванное взлетом Ubuntu. Похоже, отсюда следует, что хотя первичный рост Ubuntu мог быть результатом перетягивания пользователей от других дистрибутивов, для более долгого, более растянутого роста, наблюдаемого у Ubuntu после 2005 года, это уже не так. То есть должно существовать ощутимое число новых пользователей Linux, пробующих открытую ОС в первый раз.

Ubuntu удалось сделать что-то намного лучше, чем всем остальным дистрибутивам. Он сумел извлечь выгоду из растущей стабильности и популярности свободного ПО и достичь типовой компьютерной аудитории способом, оказавшимся недостижимым для других дистрибутивов Linux.

Комментарий сообщества

Марио Лимончелло [Mario Limonciello], главный герой Mythbuntu, считает, что доминирование Ubuntu может вызвать проблемы для сообщества. «По-моему, тот факт, что Ubuntu и Linux стали синонимами, в действительности приводит к тому, что я считаю большой проблемой, — сказал он. — Очень уж много развелось дистрибутивов. Разработчики разбрасываются, прилагая массу дублированных усилий ради мелких изменений в своем личном дистрибутиве». Он признает, что его слова нужно принимать с щепоткой соли, так как он и сам работает над Mythbuntu, однако на карту поставлены серьезные вопросы. «Я думаю, что разница здесь в том, что мы посылаем свои труды в архивы Ubuntu, и выигрывает все сообщество Ubuntu. А некоторые другие дистрибутивы, основанные на Ubuntu, вроде Mint, так не поступают. И это, на мой взгляд, вредит всей экосистеме».

Однако можно выдвинуть аргумент, что то же самое говорят об обратной совместимости с Ubuntu разработчики Debian. Клеман Лефевр рассказал нам, что мотивацией Mint было «создать комфортную операционную систему с элегантным рабочим столом, использовав лучшее на рынке и добавив то, чего не хватает». Это совершенно иная стратегия, чем просто улучшение Ubuntu. «Мое личное мнение таково, что Ubuntu был тогда лучшей операционной системой, — сказал он, — и раз уж приходится вставать на плечи гигантов, мы выбрали самого высокого».



Будущее Ubuntu

Прольется ли следующий релиз холодным дождем на парад Ubuntu, или же вознесет нас к облакам?

Процесс разработки дистрибутива Linux прозрачен, в отличие от такового для OS X или Vista. На любой стадии видно, что планируется сделать, и даже допускается скачать последнюю сборку вашей любимой системы. Ubuntu не является исключением, и следующая, октябрьская, версия будет называться Karmic Koala [Кармический Коала]. По названиям релизов Ubuntu можно определить намерения разработчиков, и мы думаем, что Karmic Koala – скорее промежуточный релиз, чем большой скачок. Ну, это если вы не владеете вычислительным центром.

«Хороший коала знает, как разглядеть за деревьями лес, даже если его голова в облаках, – говорил Марк Шаттлворт, представляющий Koala. – Ubuntu собирается держать свободное ПО на переднем

крае облачных вычислений, вобрав API от Amazon EC2 и упростив его настолько, чтобы каждый мог настроить

«Новая функция для облаков — Eucalyptus.»

свое облако, используя полностью свободные инструменты». Так что наш Koala будет витать в облаках. И, что более важно, для настольных пользователей он будет полезен только частично.

Eucalyptus

Если вы до сих пор используете телефонный модем, объясним, что облако – это последний писк среди онлайн-метафор. Например, если вы используете Gmail, то ваша почта находится



➤ Новая версия Ubuntu может стать первым из крупных дистрибутивов, где облачные вычисления сделаны козырем продаж.

«в облаках». Пользователи Google Docs держат там свои документы (на вопрос, где хранятся ваши данные, можете невозмутимо ткнуть пальцем в небо). Шаттлворту видится перемещение данных все дальше от наших домашних папок и USB-брелков – в центральные репозитории, управляемые компаниями типа Google, и он хочет, чтобы Ubuntu был в сердце этой революции. Это не битва за ваш рабочий стол, это битва за вычисления на уровне корпораций.

Важная новая разработка в сфере облачных вычислений называется Eucalyptus. Это вычислительная платформа, позволяющая



Ubuntu Format



➤ Хотя мы радовались каждому новому релизу Ubuntu, наши похвалы редко бывали без ложки дегтя.

Мы следили за Ubuntu с момента выхода Warty Warthog в 2004, и первая наша приглядка к нему всплыла в Сравнении LXF62, где оценивались популярные дистрибутивы по состоянию на конец 2004 года. Наш вердикт гласил: «Это Debian, но без раздражающего меню Debian и с лучше организованными пакетами». Нам не понравилось, что KDE, да и любой другой рабочий стол, нельзя было установить через менеджер пакетов. Это было устранено с появлением Kubuntu. В июле 2005 мы дали детальный двухстраничный обзор для Ubuntu 5.04, с рейтингом 8/10. Наше мнение было «Мощный и исключительно удобный», хотя мы все еще горевали из-за коричневого цвета.

Релиз Breezy Badger доказал, что Ubuntu набрал свой темп. Мы дали ему 10/10 за легкость применения, и назвали «простым в использовании дистрибутивом, который следует попробовать каждому линуксоиду». Мы даже не упомянули, что он все еще коричневый! Версия 6.06, с механизмом установки с Live CD, была первым релизом, урвавшим награду Top Stuff нашего двухстраничного обзора. В ожидании Dapper Drake, мы впервые отме-

тили дополнительные украшения Ubuntu, с надеждой, что в следующую версию будут интегрированы AIGLX и Xgl. «Именно этот дистрибутив мы рекомендуем новым пользователям, тем более сейчас, когда Mandriva немного растерялась».

Но новейшие функции были претворены в жизнь только в следующем релизе, 6.10, который был первой версией после выхода Long Term Support. Противопоставив ему Fedora Core 6, мы были вынуждены сказать: «Обычно мы напропалую хвалим Ubuntu, но на сей раз сделаем это более сдержанно. Он выглядит не так круто, как Fedora, но это потому, что проект Fedora многое наवरстал».

Наши заключения по Ubuntu после версии 7.04 отдавали горечью: порыв первых нескольких релизов выдохся. Например, в обзоре 7.04 мы отметили, что «если у вас нет проблем с проприетарными драйверами, стоит остаться на 6.10» и что мы «ощущаем нотку разочарования». Ситуация выправилась в 2008 году, и мы снова смогли порекомендовать Ubuntu, несмотря на включение бета-версии Firefox 3.0 в редакцию Long Term Support.



➤ Кто знает — быть может, Karmic Koala сбросит старую коричневую шубку, прельстившись зеленым, а то и синим!

динамически масштабировать приложения через сеть «прозрачных» компьютеров, работающих в облаке на вашем собственном оборудовании. Наиболее популярным сервисом облачных вычислений является EC2 от Amazon. Клиенты могут выбрать емкость и характеристики своего виртуального оборудования, работающего на EC2, а затем запускать задачи на супер-растяжимой платформе. Это невероятно масштабируемо и относительно дешево, и добавление Eucalyptus к Ubuntu приведет к встраиванию этой возможности в операционную систему, дав любому вычислительному центру возможность либо создать свое облако, либо упростить использование EC2.

Вы получаете не только преимущества масштабируемости и мощности ваших приложений, но также потенциал для экономии на счетах за электричество: ведь когда в облаке нет активности, энергопотребление фактически равно нулю. Тактика внедрения виртуальных машин в серверные решения применялась и в Novell, и в Red Hat; интересно будет увидеть, как Canonical применит другую, но родственную технологию для своего дистрибутива. И хотя настольным пользователям Ubuntu она мало что даст, однако серьезно подымет профиль серверной версии, а значит, и дистрибутива Ubuntu в целом.

Цвет — тоже функция

Ну, а по части поверхностных сдвигов — была также обещана новая палитра рабочего стола. Насчет полной отмены коричневого Шаттлворт высказался уклончиво, но похоже, что стандартный рабочий стол в Karmic совершенно преобразится. Также вероятно, что обновится стартовая графика и произойдет дальнейшая оптимизация загрузки. Марк как-то упомянул об использовании установки видеорежимов на уровне ядра (kernel mode setting), и мы надеемся, она будет включена по умолчанию. Тогда Ubuntu блеснет полностью графическим запуском, по типу заставки Plymouth, используемой сейчас в Fedora. Ubuntu может применить Plymouth, или Upstart от Canonical, однако конкретный выбор для Koala пока еще на стадии обсуждения. Все эти изменения — шаги мелкие, не революционные; так что, видимо, у нас будет здоровый и стабильный 11-й релиз Ubuntu.

Однако яблоком раздора может стать шестимесячный релиз-цикл. На наш вопрос, какие изменения он бы поприветствовал, Клеман Лефевр, разработчик основанного на Ubuntu дистрибутива Linux Mint, сообщил нам, что желал бы большей консолидации базовой версии, а не акцента на новейшие функции. «Конечно, — сказал он, — это не имеет смысла для Ubuntu, пока

Голос хозяина

Мы могли и сами догадаться о целях Ubuntu, но предпочли пойти к голове лошади — Марку Шаттлворту.



➤ Да, мистер Шаттлворт когда-то был долгогрив и мечтал стать космонавтом.

Еще в 2005 Linux Format проник в бункер Canonical, где Марк Шаттлворт строил планы по завоеванию мира дистрибутивов. Он был точен в своем первом предсказании: «Мне кажется, что маятник индустрии программных продуктов качнулся в сторону свободного ПО». Прав он был и в оценке тогдашней неприязни к Ubuntu. «Я убежден, что мы на пике этой непроизвольной реакции. Вокруг витает FUD (Fear, Uncertainty, and Doubt — страх, неуверенность, сомнение)».

Когда мы спросили Марка, чем он занимается в настоящее время, он описал нам то, что, возможно, впоследствии стало Launchpad'ом — репозиторием проектов Canonical. «Я много работаю над инфраструктурой web-сервисов, — сказал он. — Их концепция спо-

собна реально улучшить сотрудничество людей из сообщества открытого ПО. Нас ждет эра сотрудничества, и грядущие 20 лет определят программы и инструменты для совместной работы». Вспомнивая эти слова, удивляешься, что Canonical не подняла идею об облачных вычислениях раньше.

Затем мы беседовали с Марком в середине 2006, сразу после релиза 6.06, и спросили его, пострадает ли качество 6.10 в результате двухмесячной задержки выхода 6.06. «То есть отведем ли мы на Edge полные шесть месяцев, отстав от Gnome на два месяца, или за месяц-два все наверстаем? Кажется, имеется сильная поддержка второго варианта, и мы быстро сделаем Edgy современным релизом».

Так оно и вышло.

мы не станем восходящим компонентом их дистрибутива. Я реально счастлив от того, что делает Ubuntu, и если бы я мог что-то поменять... это был бы отказ от расписания релизов и возврат к политике «выпуска по готовности», чтобы обеспечить более высокий уровень качества по сравнению с предыдущими версиями».

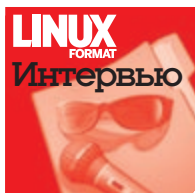
Марио Лимончелло, разработчик Mythbuntu, рассуждает примерно так же. «Я бы предпочел, чтобы цикл релизов был не строго шестимесячным, — сказал он нам. — В нескольких последних выпусках был ряд ошибок, которые оставили «живых», переложив их исправление на Stable Release Update. По-моему, имея исправление, которое точно работает, не следует откладывать его на потом, только чтобы выдержать срок выхода CD. Лучше исправлять ошибки побыстрее и радовать пользователя прямо из коробки». LXF

«Ubuntu блеснет полностью графическим запуском.»



Занимательная МЕХАНИКА

Мы давно не пишем программы прямо в машинных кодах, а значит, кто-то должен писать программы, генерирующие код за нас. Александр Горшенев знает эту работу не понаслышке.



Традиционно, в программировании выделяют три задачи, решение которых остается своего рода уделом избранных: разработка ядра ОС, создание компиляторов, а также взлом и защита приложений. Что касается первого, Линус Торвалдс [Linus Torvalds] еще не засветился на страницах

LXF; правда, мы успели (и не раз) переговорить с Эндрью Мортонем [Andrew Morton]. Хакеры (или как вы предпочитаете называть тех, кто занимается изысканиями в третьей области?) — по натуре народ скрытный и к публичной славе не стремящийся. Что осталось не охваченным? Правильно — компиляторы! Руководствуясь этой идеей, мы перехватили разработчика Sun Studio и сотрудника петербургского офиса Sun Microsystems Александра Горшенева вскоре после его выступления на конференции Sun Tech Days, посвященного высокопроизводительным приложениям.

Linux Format: Александр, в списке докладчиков Sun Tech Days вы представлены как «живой пример того, кто может вырасти из ребенка, у которого любимой игрушкой был программируемый микрокалькулятор»...

Александр Горшенев: Да, это была «Электроника БЗ-21».

для разработки компиляторов Sun Studio. Ну, и набрали — в том числе и меня.

LXF: А о компиляторах с каких языков идет речь?

АГ: Сейчас я работаю в группе кодогенерации для процессоров Intel, AMD, так что это не играет роли. А в 1998 году набрали в группу разработчиков компилятора C. Я трудился в ней пять лет, а потом стал плавно перемещаться вниз по стеку.

LXF: А вам приходилось работать только над компиляторами Sun Studio или еще какими-нибудь другими? Ну, скажем, GCC...

АГ: В моем положении принимать участие в создании сторонних компиляторов не очень хорошо: внесешь правку, а потом кто-нибудь скажет, что-де твоя запятая в Sun Studio проистекает непосредственно из GCC.

LXF: Да, я понимаю.

АГ: Кстати, о запятых. Компилятор C от Sun — наследник PCC (Portable C Compiler) Стивена Джонсона [Stephen C. Johnson], который был написан на смену самому первому C-компилятору Кернигана и Ритчи и достался нам вместе с AT&T Unix. Не так давно ребята из FreeBSD снова вытащили PCC на свет, чтобы использовать в своем проекте (GCC они, как известно, недолюбливают), ну и я пошел поглядеть, что он из себя представляет. Выяснилось, что структура очень похожа на то, что есть у нас. Так что в Sun Studio можно встретить запятые если не от Кернигана и Ритчи, то от других отцов-основателей, таких как Джонсон.

LXF: Часто приходится слышать мнение, что на нынешнем этапе развития технологий писать прикладные программы можно как угодно — компилятор все оптимизирует. Да и ассемблер представлять себе вовсе не обязательно. Что вы думаете по этому поводу?

АГ: Знаете, у нас есть поговорка: «доверяйте компилятору в простых вещах и помогайте ему в сложных». Один плюс два компилятор свернет в константу за вас; цикл, наоборот, развернет и инвариант из него достанет. Но есть вещи, в которых ему надо помогать: например, если сообщить дополнительную информацию вроде «один и тот же указатель никогда не может указывать сначала на int, а потом на float», он сможет применить это знание с пользой, но сам до него не догадается.

Что же касается ассемблера, то его знание, безусловно, помогает. Если речь идет о прикладном программировании на C или C++, то надо понимать, что эти языки ориентированы на конкретную архитектуру машины — я имею в виду не Intel, Sparc и иже с ними, а более общие вещи: что у машины есть регист-



LXF: А что было потом? Вы помните свою первую серьезную программу? Что она делала?

АГ: Нет, первую не помню, но помню седьмую. Это был паровоз на БК0010-Ш. Он, знаете ли, ехал.

LXF: И как же вы перешли от паровозов к компиляторам?

АГ: Это было в самое безденежное время, в лихие девяностые... Точнее сказать, шел 1998 год — дефолт, все было не так чтобы совсем плохо, но еще и не хорошо. В то время на матмехе СПбГУ было отделение МЦСТ, которое курировала Sun Microsystems. Однажды они решили набрать группу

ры, их несколько штук, есть память... Если мы возьмем принципиально другой процессор — скажем, стековый или CUDA от Nvidia, где много-много маленьких ядершек, то там и языки должны быть другими. Половина знания об архитектуре заключена в семантике языка. Ну и, конечно, всегда полезно понимать, что на самом деле происходит внутри вашей программы: передается ли параметр функции так или этак. Это не нужно в «совсем прикладных» программах, но если взять тот же *MPlayer*, то без ассемблерных вставок уже никуда.

LXF: А какой компилятор, на ваш профессиональный взгляд, лучше подходит для каких задач? Периодически приходится слышать фразы вроде: «Вот пересоберем ее XYZ, и все будет летать» — хотелось бы понять, имеют ли они под собой основания.

АГ: Я могу говорить за мир Unix. На компиляторы (C и C++) можно смотреть с разных сторон — например, кто на каких платформах работает. Здесь есть явный лидер — *GCC*. Другая широкая группа — компиляторы Sun, Intel, SGI, изначально разработанные для собственных Unix-платформ, а затем портированные в Linux, где они все и встретились. Это было знаменательное событие: до тех пор никто не пробовал запустить их на одной машине. Такие встречи позволяют помериться силами по-честному.

Или, скажем, можно взять соответствие стандарту. Если у вас есть программа, написанная на GNU C (да, есть на свете такой язык), скажем, ядро Linux, то остальные компиляторы тоже могут ею «позаниматься», но с переменным успехом. В сфере высокопроизводительных вычислений ситуация будет третья. Там, кроме *SunCC* и *ICC*, есть еще «компиляторы на букву P»: *PGI* и *PathScale*.

LXF: Но собрать-то ядро Linux при помощи Sun Studio можно?

АГ: Ну, как вам сказать — это и при помощи *GCC* сделать не всегда получается: возьмите «не ту» версию компилятора, и увидите только кучу сообщений об ошибках. Без исправления таких «подводных камней» ядро Linux компиляторам Sun Studio не собрать; ну, а если исправить, так вы ведь сразу же скажете, мол, «с патчами не считается»? Когда мы только портировали Sun Studio в Linux, вопросы совместимости стояли весьма остро, но с каждым релизом ситуация улучшалась. В Sun Studio 12 Update 1 была проделана большая работа, чтобы достичь совместимости с *GCC* на уровне ассемблерных вставок — мы научились не только понимать их, но и агрессивно оптимизировать. Нашей тестовой площадкой служат *Ffmpeg* и *MPlayer*: в них огромное количество встроенного ассемблера, и задачей наших разработчиков было обеспечить корректную сборку обоих проектов от начала и до конца.

LXF: Вы несколько раз упомянули портирование Sun Studio в Linux — а что вообще скрывается за этими словами? Насколько сложно перенести компилятор из одной Unix-системы в другую?

АГ: Трудности при портировании делятся на две категории: ожидаемые и неожиданные. Последние делают жизнь интересной, но ужасной [смеется]. Начинать надо с какого-то «зернышка» —



«Доверяйте компилятору в простых вещах и помогайте в сложных.»

seed compiler, который генерирует родной код. Единственный такой компилятор для Linux — это *GCC*, то есть начинать процесс нужно со сборки наших исходных текстов в нем. При этом мы (и не только мы, а все, кто занимается портированием) сразу же сталкиваемся с рядом неприятных проблем: стандарт языка C отдает некоторые вещи на откуп реализации. Например,

если знаковая перечисления (*enum*) не указана, разработчик компилятора может выбрать ее по своему усмотрению и задокументировать. Разумеется, каждый делает это по-раз-

ному. В итоге одному из наших товарищей пришлось изготовить специальную версию *GCC*, которая бы вела себя в этом смысле как компиляторы Sun, то есть (в данном примере) поддерживала ключ *-fsigned-enum*. Дальше, у нас есть оптимизирующий кодогенератор и не оптимизирующий. Первым делом был портирован последний — это сравнительно простая программа. В итоге у нас получился компилятор C, который «вытянул» остальной C-мир. Ну а потом мы долго-долго возились с C++. С тех пор я еще больше недолюбиваю C++.

LXF: Вот как? Кстати, а как обстоят дела с совместимостью компиляторов C++ из Sun Studio и GCC?

АГ: Смотря что вы имеете в виду. Подозреваю, что речь идет о двоичных интерфейсах (ABI) — здесь есть недоделки, но опять же Sun Studio 12 Update 1 обещает большие положительные сдвиги в данном направлении.

LXF: Остался последний вопрос, который мы, как сторонники свободного ПО, просто не имеем права не задать: а планирует ли открыть исходные тексты Sun Studio?

АГ: Если вас интересует короткий ответ, то процесс идет. В сущности, все упирается в права на интеллектуальную собственность: части кода Sun Studio были получены от AT&T и других компаний. Мы не можем просто взять и выложить исходные тексты где-нибудь на sun.com — необходимо удостовериться, что при этом не будут ущемлены права третьих лиц. Для Solaris, например, на такую проверку кода ушло несколько лет. Сколько потребует Sun Studio — поживем, увидим. **LXF**

Фото: Татьяна Цыганова

Каскад из CAD'ов

Повертев модель Cassini во *FreeCAD*, Николай Колдыркаев был так впечатлен скоростью обработки 300 000 деталей, что полгода уговаривал российских разработчиков Open CASCADE открыть личико. Следующие полгода он потратит на то, чтобы написать портрет.



**Наш
эксперт**

**Николай
Колдыркаев**
Николай – техни-
ческий писатель
и IT-журналист,
а также автор
книги «Открытые
и бесплатные
программы
для Windows».

Открытое программное обеспечение – это не только свободно распространяемые приложения, но и библиотеки, на основе которых можно и нужно создавать вполне коммерческие решения. Разработчики технологии Open CASCADE (www.opencascade.org) придерживаются открытой модели и концентрируются на коммерческой поддержке пользователей. В результате на основе специализированной свободной библиотеки *Open CASCADE* был разработан целый ряд САПР-систем промышленного уровня, а также специализированный дистрибутив Linux.

В 1999 году французская фирма MATRA Datavision, известная в то время как международный поставщик технологических решений и услуг в области CAD/CAM/CAE/PDM-систем, объявила об открытии исходных текстов своей библиотеки *Open CASCADE*. В результате с 2000 года все, кто так или иначе был заинтересован в разработке собственной системы автоматизированного проектирования (САПР), получили бесплатный доступ к функционалу *Open CASCADE*, на базе которой стали создаваться все новые и новые специализированные программные решения в данной области.

Open CASCADE сегодня – ...

...это целый набор библиотек и средств разработки программного обеспечения для трехмерного поверхностного и твердотельного моделирования, визуализации, обмена данными

с другими программами для проектирования и численного моделирования. Кстати, именно для разработки программ численного моделирования *Open CASCADE* подходит лучше всего. Специалисты по системам автоматизированного проектирования утверждают, что это могут быть не только CAD/CAM/CAE/PDM-системы, но также AEC (Assembly Engineering Manager) или даже геоинформационные (GIS) приложения.

Исходный код библиотеки *Open CASCADE* доступен и распространяется бесплатно по лицензии Open CASCADE Technology Public License, являющейся производной от GNU LGPL. Сторонние разработчики могут использовать код *Open CASCADE* в своих продуктах (в том числе коммерческих), однако обязаны, в соответствии с LGPL, отсылать любые изменения в исходных текстах *Open CASCADE* ее оригинальному разработчику – Open CASCADE S.A.S. Последняя являет собой яркий пример успешного ведения бизнеса, в основе которого лежит открытое ПО. Вот уже 10 лет компания зарабатывает не на продаже программ, а на создании специализированных CAD/CAM/CAE-приложений и их технической поддержке.

Соответственно, и лицензии на это ПО самые разные. Среди программ, разработанных на основе *Open CASCADE*, есть и свободно распространяемые, и коммерческие, и даже специализированные закрытые системы для промышленного применения, например, в авиастроении и ядерной энергетике. Есть и «наши» разработки: например, появившаяся в самом начале века украинская система *Океан-CAD*, известная тем, что о ней мало известно, но, тем не менее, предназначенная (судя по плохо сохранившимся экранним снимкам и другим непроверенным данным) для моделирования авиационных двигателей.

Что умеет Open CASCADE?

С момента публикации исходных текстов, *Open CASCADE* пользуется заслуженной популярностью среди разработчиков во всем мире. Для создания специализированных CAD/CAM/CAE/PDM-систем на основе *Open CASCADE* программистам доступны следующие возможности:

- » алгоритмы поверхностного и твердотельного моделирования (скругления и фаски, поверхности смещения и протягивания, булевы операции);
- » обмен данными (как с нейтральными форматами IGES и STEP, так и со специфичными, такими как ACIS, Parasolid и DXF);
- » исправление ('shape healing' – «лечение» некорректных геометрических данных и восстановление поверхностей);

Расшифруем сокращения

По числу использованных ТБА (они же – трехбуквенные сокращения) эта статья явно метит в лидеры. Если вы далеки от систем автоматизированного проектирования, то вот что может пояснить по их поводу всезнающая Википедия:

» CAD – Computer-Aided Design – система автоматизированного проектирования. Это программный пакет, предназначенный для создания чертежей, конструкторской и/или технологической документации и/или 3D-моделей.

» CAM – Computer-Aided Manufacturing – подготовка технологического процесса

производства изделий, ориентированная на использование ЭВМ.

» CAE – Computer-Aided Engineering – общее название для программ или программных пакетов, предназначенных для инженерных расчетов, анализа и моделирования физических процессов.

» PDM – Product Data Management – система управления данными об изделии. Это организационно-техническая система, обеспечивающая управление всей информацией об автомобиле, корабле, самолете и так далее.

Подробности – на <http://ru.wikipedia.org>.



» **Юрген Ригель — разработчик открытой и бесплатной САПР FreeCAD.**

» визуализация (включая прозрачность и наложение текстур).

Однако Open CASCADE — это не только специализированные библиотеки. Эта технология предоставляет полную среду разработки приложений для 3D-моделирования и визуализации, которая включает

- » средства быстрого создания приложений (Application Framework),
- » систему отладки (Test Harness),
- » широкий набор примеров,
- » исчерпывающую документацию.

Таким образом, разработчики совершенно бесплатно получают необходимый и достаточный инструментарий для того, чтобы начать создание собственных САПР-приложений. В то же время, компания Open CASCADE S.A.S. предлагает своим заказчикам ряд закрытых коммерческих компонентов — например, для обеспечения обмена данными с проприетарными САПР-системами. Используя такие компоненты, сторонние разработчики создают так называемое Value Added Software — программное обеспечение коммерческого или даже промышленного уровня.

Свободу САПР'ам!

Было бы странно, если бы на основе свободно распространяемой библиотеки никто не попытался бы создать свободную же систему автоматизированного проектирования. FreeCAD (free-cad.sf.net) — одна из таких попыток. Разработчик FreeCAD, Юрген Ригель [Jürgen Riegel], работает в корпорации DaimlerChrysler. Он позиционирует свою программу как первый бесплатный инструмент проектирования механики, созданный на основе библиотеки Open CASCADE. Сам автор сравнивает FreeCAD с такими развитыми коммерческими системами, как CATIA версий 4 и 5 и SolidWorks. Но полноценной САПР

FreeCAD пока назвать сложно — в интерактивном режиме представлен лишь базовый инструментарий библиотеки Open CASCADE. В результате в программе FreeCAD не так-то просто что-нибудь спроектировать и тем более смоделировать — большинство операций выполняется из командной строки. Тем не менее, в ней можно работать с файлами, форматы которых поддерживает Open CASCADE.

Другой открытый проект на базе Open CASCADE предоставляет больше возможностей для редактирования и даже позволяет подключать дополнительные модули расширения. Программа HeeksCAD (code.google.com/p/heckscad/) позволяет импортировать твердотельные модели из файлов STEP и IGES. Она также умеет визуализировать модели, выполнять редактирование с помощью смешивания и булевых операций, сохранять наработки в форматах IGES, STEP и STL.

Двоичный дистрибутив HeeksCAD пока доступен только пользователям Windows. Но пользователи Linux также могут принять участие в тестировании этой программы, собрав ее из исходных текстов, доступных через Subversion. Подробно этот процесс описан на странице <http://code.google.com/p/heckscad/wiki/CompilingForUbuntu>.

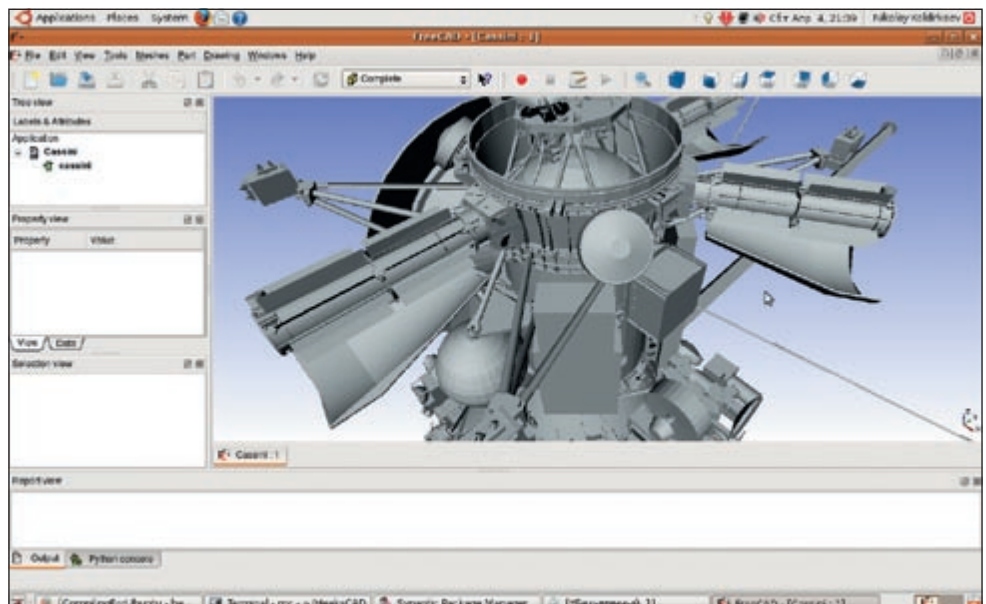
К сожалению, многие создатели САПР на платформе Open CASCADE не стремятся поделиться своими наработками с широкой общественностью, и зачастую это оправдано тем, что в них используются еще и коммерческие компоненты. Сама же фирма Open CASCADE S.A.S., кроме своего основного продукта, продвигает собственную открытую интегрируемую платформу для численного моделирования под названием SALOME.

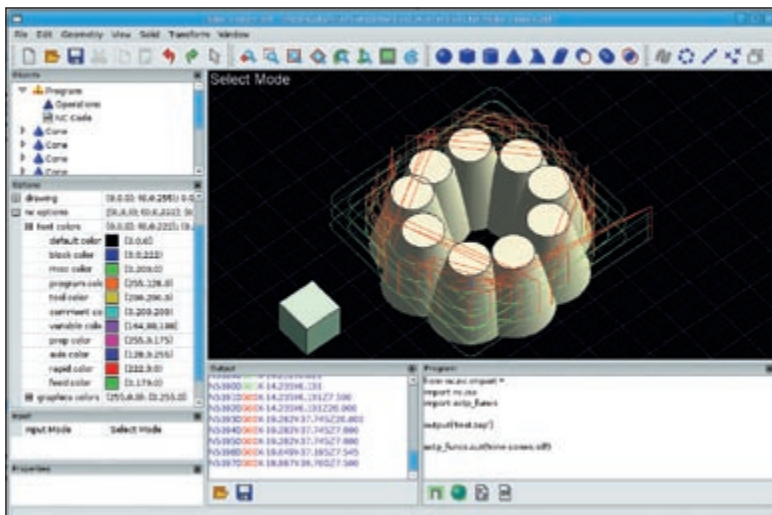
Шалом, SALOME!

Как и Open CASCADE, платформа SALOME распространяется на условиях GNU Lesser General Public License. Первоначально она играла роль связующего звена между CAD- и CAE-системами. В настоящее время SALOME объединяет в себе различные модули, применяемые в приложениях численного моделирования — от создания моделей в САПР до параллельных вычислений. При этом САПР-средства в SALOME как раз и реализуют возможности технологии Open CASCADE.

Проект SALOME на сегодняшний день объединяет 20 участников, включая таких лидеров рынка, как Renault (автомобилестроение), Bureau Veritas (кораблестроение), EADS (авиационное) и другие. При поддержке Правительства Франции, одной из самых серьезных областей применения SALOME стала ядерная энер-

» FreeCAD трудно назвать полноценной САПР, но для демонстрации возможностей возможностей библиотеки Open CASCADE она вполне подходит.





➤ Импорт и экспорт DXF в *HeeksCAD* только разрабатывается, но уже сейчас эта программа предоставляет больше средств для редактирования моделей, чем *FreeCAD*, да и развивается динамичнее.

гетика. Платформа SALOME используется как база для проекта NURESIM (European Platform for Nuclear REactor SIMulations), который предназначен для полномасштабного моделирования ядерных реакторов.

Для использования технологии Open CASCADE группой разработчиков служит специальный инструментарий — WOK (Workshop Organization Kit). Он позволяет большому количеству людей создавать несколько программных продуктов одновременно. При этом разработчики могут генерировать несколько версий программы для разных аппаратно-программных платформ и специальные версии с функциональным наполнением, диверсифицированным под разные рыночные усло-

вия. WOK поощряет использование в разных программах одних и тех же компонентов, для достижения максимальной эффективности, и, в общем, позволяет наладить производство САПР-приложений промышленного уровня.

Более того, в рамках WOK возможно использование специального макроязыка CDL (Component Definition Language) программной платформы Open CASCADE. Поэтому компоненты, созданные при помощи CDL, будут использовать программную архитектуру Open CASCADE.

Все вместе

Технология Open CASCADE и система численного моделирования SALOME стали центральной частью специализированного LiveDVD-дистрибутива для CAD/CAM/CAE. Дистрибутив CAE Linux основан на PCLinuxOS и загружается непосредственно с DVD без необходимости установки, хотя при желании вы можете выполнить и эту операцию. Его создатели проделали большую работу, собрав вместе и обеспечив интеграцию наиболее развитых средств CAD/CAM/CAE-систем для Linux. Например, инструменты численного моделирования системы SALOME в дистрибутиве расширены средствами анализа конечных элементов от Code Aster.

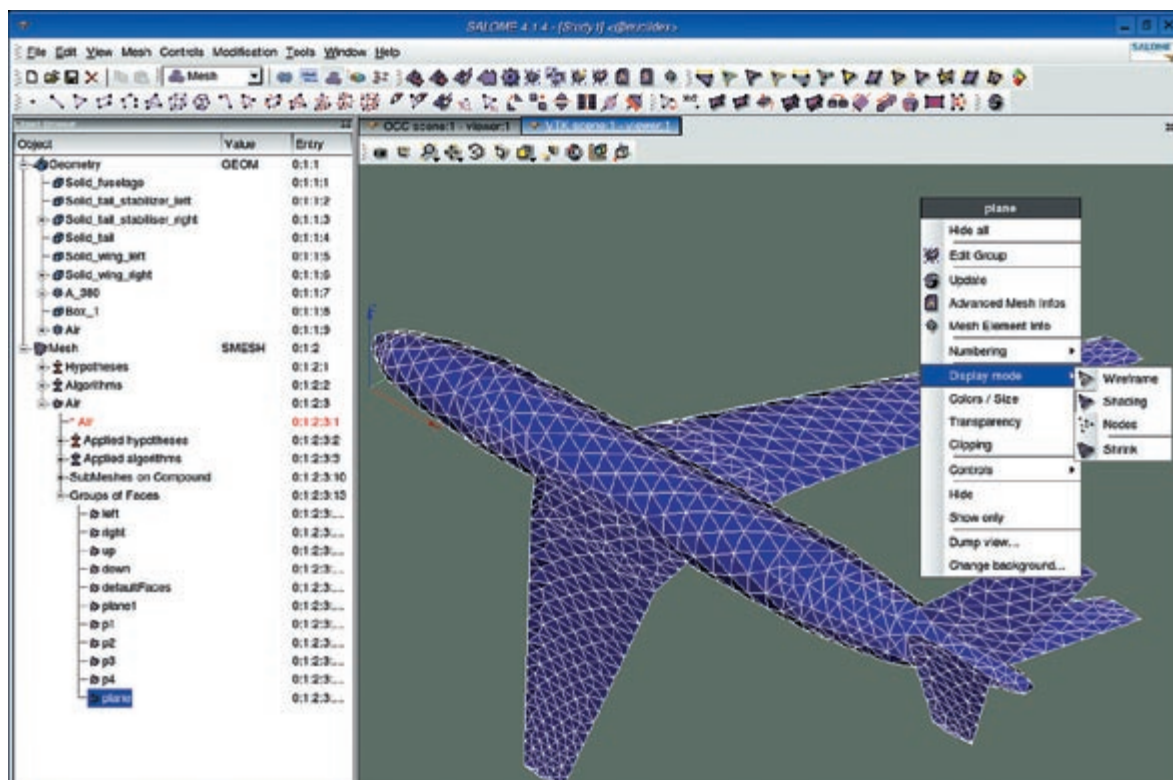
Как можно видеть, спектр применения открытой технологии трехмерного моделирования Open CASCADE настолько широк, что до сих пор не было создано универсальной открытой САПР

на его основе. И это не парадокс. Дело в том, что каждая новая разработка на базе данной технологии обычно нацелена на решение конкретных отраслевых задач. Но если сторонние раз-

работчики объединят усилия для работы над новым открытым проектом, то мы сможем рассчитывать на появление полноценной универсальной САПР для Linux. Лично я считаю, что полная реализация заложенного в Open CASCADE потенциала в открытом ПО для моделирования — это лишь вопрос времени.

«SALOME используют для моделирования ядерных реакторов.»

➤ Собственная платформа для численного моделирования Salome — центральное приложение в специализированном дистрибутиве CAELinux.



Слово — разработчикам

Linux Format не упустил возможности побеседовать с **Сергеем Зерчаниновым**, менеджером по поддержке клиентов (Open CASCADE Customer Support manager) в компании Open CASCADE Russia.

Linux Format (LXF): Расскажите немного о компании Open CASCADE S.A.S. Чем она занимается? Сколько людей в ней работает? И какая роль отводится отечественному филиалу?

Сергей Зерчанинов (СЗ): Open CASCADE S.A.S. — сервисная компания, предлагающая разработку специализированного программного обеспечения для решения инженерных задач. Open CASCADE является дочерним предприятием компании Euriware (Франция), входящей в группу компаний Areva. Опыт работы на рынке — более 25 лет.

Штат сотрудников — около 100 человек, включая квалифицированных программистов, инженеров-экспертов, опытных консультантов и проектных менеджеров, обладающих глубокими знаниями в сферах трехмерного и численного моделирования, обмена данными САПР, визуализации и графического пользовательского интерфейса.

В головном офисе компании Open CASCADE S.A.S., помимо администрации, работает несколько проектных менеджеров, ответственных за исполнение проектов и развитие бизнеса с крупными клиентами, такими как AREVA, CEA, EDF, RINA и так далее.

Основная же техническая работа по реализации проектов и поддержке заказчиков, по выпуску и развитию Open CASCADE Technology и SALOME осуществляется российской командой, которая насчитывает около 70 инженеров. Мы также занимаемся технической поддержкой: отвечаем на вопросы, консультируем, обучаем пользователей.

LXF: Интересно, а каков вклад отечественных разработчиков как в саму технологию Open CASCADE, так и в создание приложений на ее основе?

СЗ: В рамках регулярной деятельности по реализации проектов и поддержке заказчиков компании Open CASCADE S.A.S. происходит и эволюция открытых платформ, которая в первую очередь следует за пожеланиями коммерческих клиентов и призвана улучшить качество исполнения проектов заказчиков, создаваемых на основе Open CASCADE Technology. Именно российская команда осуществляет коррекцию ошибок, реализацию новых или улучшение существующих возможностей с точки зрения надежности и производительности.

С момента публикации в 2000 году версии 3.1 (первой, подготовленной в России) платформа Open CASCADE Technology непрерывно эволюционировала: всего было выпущено порядка 20 версий, последней на сегодняшний день является Open CASCADE Technology ver. 6.3. За это время структура платформы весьма изменилась, став намного более удобной; все модули платформы подверглись существенным изменениям и улучшениям, добавлялись или оптимизировались алгоритмы, добавлялась поддержка новых форматов данных, новых операционных систем.

LXF: А как много программ на основе Open CASCADE распространяется на коммерческой основе? И сколько они стоят?

СЗ: Среди заказчиков компании Open CASCADE S.A.S. большой процент составляют промышленные компании (например, TATA Motors), имеющие необходимость в специализированном про-

граммном обеспечении для внутренних нужд. Естественно, такие программы не распространяются, и более того, информация о них не может быть опубликована по соображениям защиты от конкурентов.

Другую группу составляют компании — производители программных продуктов, такие как Samtech, Midaslt. Как правило, информация об их решениях доступна в Интернете.

И наконец, Open CASCADE S.A.S. производит собственные коммерческие программные продукты, расширяющие возможности открытой платформы в области обмена данными с другими CAD-системами, специальных алгоритмов, работы с дискретными моделями (mesh) и так далее.

LXF: А можно подсчитать, сколько существует открытых проектов, использующих Open CASCADE? Вы ведете такую статистику?

СЗ: Точно оценить количество открытых проектов, основанных на наших платформах, сложно, поскольку наши пользователи не всегда информируют нас о своих разработках. Однако их не слишком много, поскольку создание такого рода программ требует высокой квалификации и значительных ресурсов, которые не всегда могут быть обеспечены даже в крупных коммерческих компаниях.

Тем не менее, на нашем сайте в разделе «Сообщество / Community» представлено около 60 интересных проектов Open Source, созданных с использованием технологии Open CASCADE. Помимо уже упомянутой вами FreeCAD, могу отметить Cadmai Рихарда Бренделя [Richard Brendel], адаптивную систему, интегрируемую в приложения (например, Internet Explorer или Excel) и позволяющую изменять их дизайн или отображать двух- и трехмерные объекты. В настоящее время разрабатывается еще одна открытая и бесплатная CAD-программа HeeksCAD.

LXF: Как вы думаете, почему выгодно использовать именно открытую технологию, свободно распространяемую библиотеку Open CASCADE? Какие это дает преимущества?

СЗ: Я думаю, основных преимуществ два:

» Экономия средств клиента благодаря модели Open Source. Не нужно покупать лицензии и делать лицензионные выплаты — вы инвестируете средства только в решение своих конкретных проблем. Экономия средств особенно высока в долгосрочной перспективе.

» Гарантия надежности решений — клиент имеет полный доступ к исходным текстам программных средств, как самих решений, так и базовых платформ. **LXF**



Ваше мнение

Знаете незаслуженно обойденную вниманием CAD/CAM/CAE-систему на основе Open CASCADE? Собрали собственную, расширенную версию FreeCAD? Знаете еще один САПР-ориентированный дистрибутив Linux?

Мы ждем ваших откликов по адресу: letters@linuxformat.ru

Управление и контроль



Джульетта Кемп оценивает трех главных претендентов на должность менеджера изменений в ваших данных: *Bazaar*, *Subversion* и *Git*.



Наш эксперт

Джульетта Кемп считает контроль версий самой полезной вещью после автоматизации резервного копирования, особенно с тех пор, как случайно нажала `rm` после удаления 2500 слов.

Системы контроля версий (СКВ) необходимы при коллективной работе над проектами, но весьма полезны и для предпочитающих сольную деятельность. Отслеживание сделанных изменений обеспечит вам откат к одной из прежних версий, если вы случайно что-нибудь испортите.

Со времени выхода устаревшей *CVS* появилось множество новых вариантов – какой же из них предпочесть? Распределенную систему или централизованную? Мы рассмотрим самые известные реализации – *Bazaar*, *Subversion* и *Git*, чтобы вы осознанно выбрали наиболее подходящую именно вашим нуждам, будь это крупный проект, любительское программирование, отслеживание изменений в конфигурационных файлах или что-нибудь более экзотичное.

Клиент-сервер против распределенки

Существует два основных типа систем контроля версий: клиент-серверные и распределенные. Бывают и чисто локальные системы, типа *RCS*, которые действуют исключительно в рамках одной машины, но сейчас они применяются редко: иметь дело с современными разновидностями в любом случае и проще, и удобнее.

Клиент-серверные системы работают в централизованном режиме: на центральном сервере существует текущая копия проекта, из которой «выписывает» ('checkout') данные локально работающий пользователь. Внеся желаемые изменения, он (или она) обновляет локальную копию с сервера (чтобы учесть изменения, которые успели сделать за это время другие пользователи), разрешает конфликты, если таковые возникают, а затем фиксирует ('commit') свою версию данных в центральном репозитории. После этого внесенные изменения становятся доступны для выписки другим людям.

Распределенные системы построены по принципу одноранговой сети – репозиторий у каждого свой, а работа синхронизируется за счет обмена пакетами изменений в форме заплат

(«патчей») или слияния ветвей проекта. На практике большинство мало-мальски значительных проектов имеют одну копию, которая считается главной ветвью разработки; но это скорее социальное различие, чем техническое.

Оба варианта имеют и достоинства, и недостатки. Вот некоторые из преимуществ распределенных СКВ:

- » Обеспечивается полное резервное копирование кодовой базы и истории изменений для каждой ветви проекта; возможно существование нескольких ветвей.
- » Упрощается работа без подключения к Интернету, так как изменения можно до поры фиксировать в локальном хранилище.
- » Проще взаимодействовать с коллегами: для этого не нужно обращаться к централизованной системе.
- » Легче создавать и ликвидировать ветви разработки: тем самым упрощается проведение экспериментов в ходе развития проекта.
- » Есть мнение, что такой способ более демократичен, и позволяет вовлекать в проекты большее число участников.
- » Можно организовать несколько «центральных» ветвей с различной специализацией (например, стабильную, текущую и релиз).
- » Фиксация изменений, просмотр истории и другие подобные операции происходят быстрее: для них не нужен доступ к центральному серверу.
- » Упрощается слияние нескольких частей проекта.

Централизованные СКВ тоже обладают рядом достоинств:

- » Любой отдельный пользователь или группа может получить полный доступ к истории и содержанию проекта (правда, иногда это достоинство может рассматриваться как недостаток!).
 - » Главная версия кода содержится централизованно, а не дробится на несколько вариантов.
 - » Проще обеспечить отказоустойчивость одного централизованного сервера (хотя бы за счет квалифицированного обслуживания), чем множества пользовательских персональных машин.
- Короче, внимания заслуживают оба варианта, хотя в наши дни популярность набирают именно распределенные системы.

«Системы контроля версий бывают двух основных типов.»

Bazaar

» Сайт <http://bazaar-vcs.org>
 » Лицензия GPL
 » Применяется в MySQL, Gnash, Squid

Распределенная система не без способностей к централизации.

Bazaar (в командной строке – **bzr**) – распределенная система, называемая авторами «СКВ, задуманной для людей». Она рассчитана на поддержку рабочих процессов разнообразных типов и предоставляет вам значительную свободу в выборе способа работы и управления версиями. Возможно использование Bazaar совместно с другими аналогичными системами (например, CVS или Subversion).

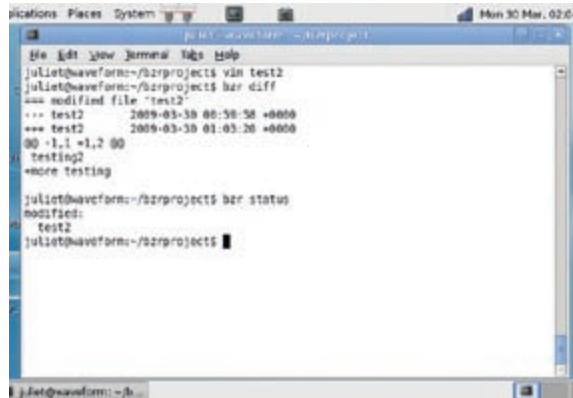
Применяя Bazaar в распределенном сценарии, вы создаете по ветви на каждую новую функцию (их называют 'task branches' – ветви для задачи), а для отправки изменений на сервер используется локальная ветвь-зеркало. В централизованной модели изменения периодически отправляются непосредственно на общий сервер. Девиз команды разработчиков Bazaar – «не стоит прогибаться под причуды ПО, пускай ПО прогнется под вас».

Одна из приятных особенностей Bazaar – работа индивидуально, вы не должны (как в Subversion, например) создавать репозиторий, импортировать файлы, а потом выписывать рабочую копию. Вы работаете себе в каталоге проекта, а Bazaar сам отслеживает сделанные изменения. Одно из очевидных неудобств – усложнение резервного копирования: следует либо держать все проекты внутри главного каталога, либо обеспечить регулярное сохранение абсолютно всех каталогов (кстати, идея сама по себе неплохая). Кроме того, ненароком ошибившись в команде **rm -rf**, легко сгубить репозиторий. Начать проект очень просто: вместо импорта и последующей выписки кода достаточно инициализировать проект внутри его собственного каталога. При желании работать в более централизованном режиме можно создать репозиторий в отдельном каталоге и выписать ветвь из него. Это делается всего лишь одной командой: **init-repo**.

Работа оффлайн

Распределенная природа Bazaar позволяет работать и вносить изменения, не подключаясь к Сети. В централизованных системах это тоже возможно с помощью локального репозитория, но его совмещение с основным сервером чревато сложностями. В Bazaar локальный контроль версий весьма удобен: вы загружаете основной проект командой **bzr branch**, и на вашей машине создается его локальная ветвь. Можно работать внутри нее, а можно создавать подветви и фиксировать изменения в них с той частотой, какая вам заблагорассудится. Команда **bzr merge** сольет вашу ветвь проекта с родительской, а завершив работу над кодом, нетрудно создать заплатку для отправки «наверх» командой **bzr send -o patchname.patch**. Ответственный за родительскую ветвь проекта может внести в нее ваш патч – а может и не внести; для этого используются те же команды, что и при слиянии ветвей. Теоретически Bazaar может обходиться без центрального дерева, но на практике большинство проектов имеют такое и сливают с ним отдельные ветви.

Алгоритм объединения ветвей Bazaar позволяет осуществлять множественное слияние, а также определять последнего общего предка ветвей. Возможно не только объединение, но и «сплетение» различных ветвей, а также разрешение весьма непростых



» Запрос изменений из Bazaar и проверка статуса репозитория.

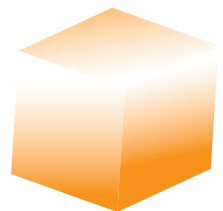
конфликтных ситуаций. Но требование общего предка у ветвей является обязательным (а вот Git умеет совмещать совершенно не родственные ветви).

Bazaar поддерживает выборочное слияние, при котором проводится не полное, а частичное совмещение изменений ветви (например, только до версии 104, или только из версий 105–107). Есть возможность отложить свои изменения до поры, изъяв их из рабочего дерева (например, чтобы упростить слияние с крупным обновлением, произошедшим в родительской ветви), а затем вернуть в проект. Это удобно, если вы работаете над несколькими заплатками, либо хотите воспользоваться правками коллег. Как и в Subversion, можно применять «хуки» ('hooks'; это скрипты, исполняемые перед определенными действиями или после них).

Полезное качество для крупных проектов – интеграция с системами отслеживания ошибок. С помощью ключа **--fixes** можно привязать к изменению номер ошибки в системах разных типов (поддерживаются Bugzilla, Launchpad, Trac, Roundup и пр.). Например, строка:

```
bzr commit --fixes project:23400 -m «Stores user birthdates properly»
```

добавит в журнал ссылку на ошибку № 23400 трекера Bugzilla для проекта project. Для Bugzilla и Trac поддерживаются упрощенные режимы настройки.



»

Прочие претенденты

Perforce Популярная схема с использованием клиент-серверной модели. Активно развивается; плохо одно – закрытая лицензия. Для открытых проектов (или с числом участников не более двух) продукт бесплатен, в остальных случаях – \$900 «с носа».

CVS Выпущена в 1986; на сегодняшний день – одна из старейших СКВ. Она централизованная, имеет ряд хорошо изученных недостатков (в частности, сложности с созданием/слия-

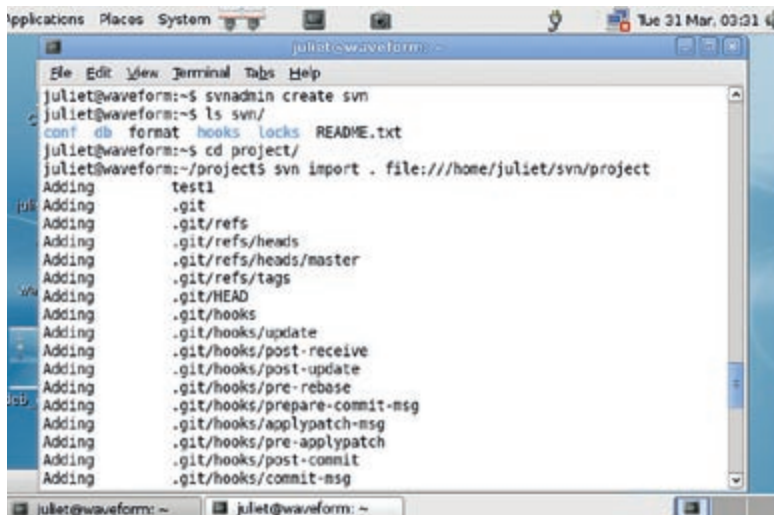
нием ветвей). Встречается до сих пор, даже поддерживается, но новые функции уже никто не добавляет. Поэтому, выбирая систему для нового проекта, присмотрите что-нибудь посвежее.

Mercurial Еще одна активно развивающаяся распределенная система. Наделена удобной системой очереди заплат. Аббревиатура командной строки – **hg**, она освежит ваши школьные знания по химии.

Subversion

» Сайт <http://subversion.tigris.org>
 » Лицензия Apache License
 » Применяется в KDE, Python, Ruby, Mono, Google Code

Централизованная система, призванная устранить часть проблем CVS.



» Организация репозитория и импорт файлов. В данном каталоге уже существует репозиторий *Git*, поэтому заодно импортируется информация из него.

Subversion создавалась на смену популярной CVS, чтобы ликвидировать самые заметные недостатки предшественницы. Как и CVS, система работает в режиме клиент-сервер. Центральный репозиторий может быть локальным (доступ через `file://`) или удаленным (доступ через `http://`, `https://` или собственные протоколы `svn://` и `svn+ssh://`).

В отличие от Bazaar, здесь перед началом работы необходимо создать центральный репозиторий (локальный или удаленный); поэтому процесс передачи файлов под контроль системы чуть более трудоемок. Покончив с ним, вы заново выписываете данные из репозитория, чтобы получить копию для дальнейшей работы.

Что касается цикла «изменение/проверка/фиксация» (внесение правок в файл, проверка на отсутствие конфликтов и отсылка изменений на сервер), команды и основные операции те же, что применяются в большинстве систем управления версиями. В принципе, начав работу с одной из них, вы получаете ключ ко всем остальным: их команды очень схожи между собой.

Разрешение конфликтов

Если в Subversion вы сталкиваетесь с конфликтом, то перед отправкой вашего файла необходимо явно пометить его как разрешенный ('resolved'). Иногда это кажется нудным, зато предотвращает случайное попадание конфликтных данных в репозиторий (впрочем, ничто не мешает попросту снять галочку вместо реального разрешения конфликта!).

Репозитории могут иметь ветви и тэги, и слить ветвь с основным кодом (обычно называемым 'trunk' — «ствол») достаточно легко. Однако многоуровневое ветвление или перекрестные связи между ветвями осуществляются «со скрипом» — с таки-

ми выкрутасами распределенные системы справляются гораздо лучше.

Можно сливать или разделять репозитории целиком — инструмент администратора SVN под названием *svndumpfilter* позволяет обрабатывать проекты с помощью фильтра. И все-таки Subversion недоступен присущий распределенным системам уровень гибкости и эффективности в работе с ветвями. Не существует встроенных команд, позволяющих взять патч и встроить его в дерево проекта; это можно делать с помощью внешней команды *patch*, но возможны проблемы с удаленными или слитыми файлами.

Маркировка изменений

Subversion поддерживает файловые мета-данные. В качестве метки файлу можно присвоить нечто вполне человекочитаемое — это прекрасная возможность добавить сопроводительную информацию.

```
$ svn propset test «test property value» myfile.txt
$ svn proplist myfile.txt
Properties on 'myfile.txt'
test
$ svn propget test myfile.txt
test property value
```

Есть некоторые специальные свойства, названия которых начинаются с `svn:` — этот префикс здесь находится «на особом положении». Например, если установить для какого-либо файла свойство `svn:ignore`, то он будет игнорироваться системой.

Как и в Bazaar или Git, поддерживаются «хуки» (скрипты, исполняемые по поводу определенных событий). Это может пригодиться, например, для тестовой сборки кода перед фиксацией изменений в репозитории; для удаления лишних пробелов; для замены пробелов на знаки табуляции или наоборот; для отправки электронных сообщений коллегам по случаю фиксации новых файлов.

И вообще для выполнения любой задачи, на которую у вас хватит воображения и способностей к скриптописанию!

**«СЛИТЬ СВОЮ ВЕТВЬ
С ОСНОВНЫМ КОДОМ
ДОСТАТОЧНО ЛЕГКО.»**

Ресурсы

» Bazaar

Учебник «Bazaar за 5 минут»
<http://doc.bazaar-vcs.org/bzr.dev/en/mini-tutorial/index.html>
 Руководство по переходу с других аналогичных программ
<http://bazaar-vcs.org/BzrSwitching>

» Git

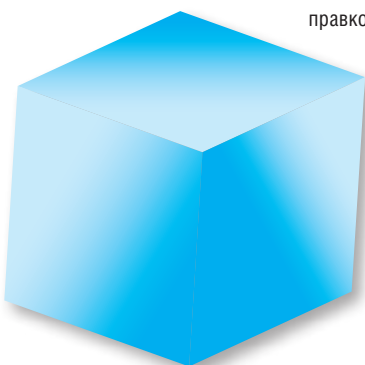
Официальное руководство Git
www.kernel.org/pub/software/scm/git/docs/gittutorial.html
 Краткий курс по переходу SVN—Git
<http://git-scm.com/course/svn.html>

» Subversion

Version Control with Subversion, книга O'Reilly, доступная бесплатно онлайн (есть перевод на русский язык)
<http://svnbook.red-bean.com>

» Общие сведения об СКВ

Статья в Википедии со сравнением разных систем
http://en.wikipedia.org/wiki/Comparison_of_revision_control_software
 Электронное сообщение Линуса Торвальдса по поводу достоинств распределенных систем
<http://lwn.net/Articles/246381>





» Сайт <http://git-scm.com>
 » Лицензия GPL
 » Применяется в ядре Linux, Gnome, Perl, X.org, VLC, Android

Чрезвычайно распределенная, и очень быстрая.

Как и *Bazaar*, *Git* — это вариант распределенной СКВ; изначально она была создана Линусом Торвалдсом [Linus Torvalds] для ядра Linux.

Одна из ключевых особенностей *Git* — поддержка нелинейного процесса разработки: идея состоит в том, что изменения вливаются неоднократно, по мере проверок ответственными лицами (именно так и происходит развитие Linux). На практике это значит очень простое слияние ветвей и деревьев, даже если они никак не связаны между собой и не имеют общих предков. Можно даже встраивать код неопределенной версии в существующее версионное дерево: на такие фокусы не способны ни *Bazaar*, ни *Subversion*. Кроме того, в *Git* значительное внимание уделяется скорости, быстроте работы с крупными проектами.

Распределенная природа позволяет *Git*, как и *Bazaar*, сопровождать каждую рабочую копию отдельным репозиторием (в подкаталоге `.git`), а не полагаться на одно централизованное хранилище, как делает *SVN*. Кроме того, в этой системе несложно создать новый проект — нужно лишь дать команды `git init`; `git add`; `git commit` в его каталоге. Это, правда, немного осложняет резервное копирование. Ну, а при желании всегда можно создать собственный вариант локального централизованного репозитория.

Совместимость

Как и *Bazaar*, *Git* работает с *Subversion*: можно пользоваться репозиторием *Subversion* непосредственно в *Git* с помощью команды `git-svn`. Это очень полезно, если, например, вы хотите просто попробовать систему или работаете с проектом *SVN*, изменять который не следует. При сильном сходстве, команды имеют и отличия от тех, что используются в *Bazaar* и *Subversion*. Есть действительно классные улучшения: например, `git diff` автоматически использует в качестве пейджера команду `less`, и вам незачем помнить о соединении процессов через канал.

Также интересно обеспечение безопасности: история записывается таким образом, что номера версий зависят от ранее сделанных изменений. После публикации версии исправить ее тайком невозможно.

На практике версии распознаются по идентификатору SHA1 ID, 160-битному числу в шестнадцатеричном формате. Естественный недостаток этого способа — длинный и сложный номер. Впрочем, в *Git* есть автодополнение, да и копирование-вставку никто не отменял.

Система тэгов в *Git* довольно мощная. Можно сопровождать любой файл совершенно произвольным описанием; иногда тэг проекта содержит полномерную документацию. Сохраняется имя автора тэга, возможна защита PGP-подписью — то есть обеспечивается подтверждение личности автора, подлинности версии, истории и дерева проекта.

В отличие от манеры *Subversion*, при слиянии сохраняется полная история изменений обеих ветвей; кроме того, ветви можно сливать многократно. Основной приоритет в *Git* отдан гибкости и возможности многократного слияния различных ветвей раз-

```

juliet@waveform:~/project$ git init
Initialized empty Git repository in /home/juliet/project/.git/
juliet@waveform:~/project$ git add .
juliet@waveform:~/project$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   test1
#       new file:   test2
juliet@waveform:~/project$ git commit
Created initial commit 867d9d8: Starting new repository for test project.
2 files changed, 3 insertions(+), 0 deletions(-)
create mode 100644 test1
create mode 100644 test2
juliet@waveform:~/project$
  
```

ного происхождения. Система работает с патчами (пакетами изменений) и обладает развитой поддержкой патчей, доставляемых по электронной почте. Вы можете просто указать программе на почтовый ящик (в формате mailbox), содержащий сообщения с патчами: они будут извлечены и пущены в дело. А для работы с группами патчей еще есть инструмент *StGIT*.

Хук слева, хук справа

Как и *Bazaar* с *Subversion*, *Git* поддерживает хуки: скрипты, исполняемые по случаю некоторых событий. Например, скрипт может проверять наличие лишнего пробела, обнаружив таковые, прервать фиксацию; или разослать электронные сообщения по окончании процесса отправки изменений. Одно плохо — *Git* неэкономно расходует место:

каждый объект в системе хранится в виде отдельного файла. Поэтому файлы, в целях экономии, время от времени «пакуются» друг с другом.

«Git имеет поддержку патчей, доставляемых электронной почтой.»

Вердикт

Все три рассмотренные системы управления версиями — замечательные экземпляры ПО: ваш выбор зависит от ваших же потребностей.

При сверхраспределенной схеме разработки, с большим количеством независимо работающих программистов, *Git* обладает рядом несомненных преимуществ. Для действующих в одиночку использование распределенной системы тоже имеет смысл, ведь в этом случае так просто создать новый репозиторий (даже из существующего каталога). А ведь чем проще пользоваться системой, тем выше вероятность ее выбора. Только не забывайте о регулярном резервном копировании!

Для более централизованных проектов лучше подходит *Subversion* — поддержка для этой системы тоже неплохая. *Bazaar* служит мостиком между централизованными и распределенными системами: несмотря на распределенную природу, в этой системе можно вести и более централизованную работу (если такая вдруг понадобится).

К счастью, затраты на эксперименты со всеми тремя вариантами (особенно распределенными) невысоки: просто возьмите себе экземпляр, да и попробуйте. Не понравится — можно потом перейти на другую систему. **ЛКС**



Intel inside

Новые чипы, виртуализация, инструменты администратора — Intel ломает все преграды, говорит Пол Хадсон, а преимущества первым получает Linux.

Компьютеризация — та же эволюция. Мы берем хорошие идеи и делаем их лучше — каждые 18 месяцев удваиваем число транзисторов, расширяем емкость устройств хранения данных, постоянно отлаживаем алгоритмы и улучшаем интерфейсы, продвигая компьютеризацию вперед.

Иногда, однако, появляется истинно новая технология, и рынок старается отреагировать. ОС с трех-четырехлетними циклами обновления должны бороться, чтобы не отстать, а вот Linux быстро откликается на все новые сдвиги в компьютерных технологиях, благодаря своей открытости.

Так, в Microsoft требуются несколько месяцев на прочтение технических спецификаций и создание кода для Windows. Затем, перед окончательным выпуском в виде части сервисного пакета или даже новой версии операционной системы, код должен пройти тщательное тестирование и процедуру контроля качества. В Linux каждый поставщик инноваций может иметь инженеров, работающих над изменениями в ядре задолго до выхода продукта, а значит, поддержка гарантируется сразу же. А кто напишет код для ядра лучше, чем инженеры той же компании, что изобрела технологию?

Один из самых поразительных подобных примеров — вклад Intel как в ядро, так и в другие свободные продукты, например, X.org. Будучи пятой компанией по величине вклада в ядро,

Intel выдает огромную массу кода: и для драйверов к оборудованию, и по технологиям, связанным с CPU, например, виртуализацией. Она даже пошла дальше и взяла на работу двух высокопрофессиональных программистов ядра и X.org — Алана Кокса [Alan Cox] и Кейта Паккарда [Keith Packard], а также десятки других, чтобы ее новые технологии хорошо работали в Linux.

К сожалению, это сложно разглядеть сквозь фасад корпорации, и в качестве конкретных примеров того, где Intel доказала

свое превосходство, мы расскажем о трех ключевых технологиях, о которых, вы, возможно, даже не слышали: Nehalem, VT-d и vPro. Заодно мы объясним, как другие свободные проекты, вроде GCC, выигрывают

от вовлеченности корпораций и как новые разработки выводят Linux на передовые аппаратные рубежи.

«Intel выдает массу кода для драйверов и виртуализации.»

История Nehalem

В конце прошлого года Intel выпустил новый процессор. Само по себе это не сильно интересно: за последние годы Intel выпускал десятки CPU, и большинство из них было x86-совместимыми. Но если отбросить маркетинговую шелуху и технические мелочи, станет ясно, что за все эти годы было только три крупных изменения.

В 1985 появился 386, и — хотите верить, хотите нет — Intel продолжал его выпуск вплоть до 2007 года, то есть этот процес-

Nehalem в действии

Intel упрямил системную шину (Front Side Bus, FSB), которая тормозила старые Хеон'ы по полосе пропускания и латентности. Также был заменен дискретный контроллер памяти, расположенный в чипе северного моста материнской платы. В новых Хеон'ах 5500 вместо них введены Nehalem QPI (Quick Path Interconnect) и встроенный контроллер памяти.

Первый — это интерфейс от-точки-к-точке, способный развить теоретическую скорость до 6,4 Гб/с на канал или до 32 Гб/с суммарно. Хеон 5500 — это чип с двумя сокетами и двумя QPI-каналами на socket, причем «четверной» Хеон 5500 на базе Nehalem с дополнительными QPI-каналами планируется уже в этом году. Что касается нового контроллера памяти, то за счет расположения внутри CPU он не просто выигрывает по части полосы пропускания и времени ожидания: Intel также захватил новый канал и ввел поддержку сверх-быстрых 1,333-МГц DDR3 DIMM'ов. Результат — резкий рост пропускной способности.

Но это только начало. Nehalem — это масса других функций, направленных на резкое улучшение и общей производительности вычислений, и специфичных серверных нагрузок. Во-первых, Intel расчленила технологию Hyperthreading (HT), позволяющую одному четырехъядерному чипу Хеон 5500 обрабатывать одновременно восемь потоков. Есть еще и новый режим Turbo Mode. По сути, это функция автоматического разгона: на ней лежит тонкое управление частотами, потреблением мощности и температурой. В случае 2,93-ГГц модели Хеон X5570, все четыре ядра могут быть

разогнаны до 3,2 ГГц при соответствующих условиях, однако от одного до трех ядер могут работать на частоте 3,33 ГГц при низкотоковой нагрузке.

Все эти чудеса едва оставляют место на описание мелких выигрывшей в производительности. Проще сказать, Nehalem — это монстр. По сравнению с 2,7-ГГц процессором Opteron с двойным socketом, самым быстрым из распространенных серверных чипов AMD, 2,93-ГГц Хеон X5570 обычно от 50 % до всех 100 % быстрее. Это показали тесты, включающие профессиональную визуализацию, моделирование гидродинамики и кодирование видео.

У нас была возможность побаловаться с новым Boston Quattro 5500 GP — диву даешься, сколько мощи втиснуто в столь малый объем. Когда-то давно нам приходилось рассматривать Boston 2300 GP, с двумя индивидуальными материнскими платами, помещенными в единый модуль 1U. Это нелегко, учитывая, сколько тепла любят выделять серверные CPU. Но в 5500 Boston взяла свою систему с двумя материнскими платами и удвоила ее, применив модуль 2U и, на изумление, 32 ядра. Да, новый Nehalem HT означает, что вы получите 64 виртуальных ядра в одной посуде.

Не так давно блейды считались лучшим способом вместить компьютерную мощь в малый объем, однако Nehalem овесть сервер Boston без ущерба разом и для производительности, и емкости дисков. Это стало возможным с появлением модели Хеон L5520, которая, при тактовой частоте 2,26 ГГц и четырех ядрах, потребляет всего 60 Вт при работе на полной мощности.



Каждая из материнских плат работает независимо от других, на каждой два четырехъядерных Хеон'а, а также свое ОЗУ и свой диск. А значит, остальные платы можно подключать по необходимости: в конце концов, самый энергосберегающий CPU — тот, что выключен. Когда все четыре платы заработают на полную мощность, вентиляторы охладят машину, производя сильный шум, но зато ваш 2U-сервер легко может оказаться самым сильным в мире: предустановленная 64-разрядная CentOS выполнит любую задачу, что ни взбредет вам в голову.

Блейды хоронить пока рано, но Quattro с Nehalem доказывает, что стоечные серверы все еще верховодят в курятнике, обеспечивая грубую силу без взлета счетов за электричество выше крыши.

сор — долгожитель-рекордсмен. Главным его преимуществом была 32-разрядность, то есть адресация до 4 Гб памяти (астрономическая цифра, по тем временам). Однако это также значит, что 386 — общий знаменатель для всех современных CPU от Intel и AMD: например, Debian все еще компилируется для 386, ибо это гарантирует работу на всех более поздних CPU.

Через десять лет Intel выпустил Pentium Pro, впервые шагнул в мир серверов. Несмотря на сохранение старой торговой марки, только с привеском Pro, Pentium Pro кардинально изменил архитектуру Pentium'a, введя то, что сейчас называется i686. Ее используют около 70 % пользователей Linux. Так как i686 был на 100 % обратно совместим с 386, специально для Pentium Pro компилировались только высокопроизводительные серверные приложения. Тогдашние нововведения Pro — исполнение с изменением последовательности, переименование регистров и встроенный

кэш — создали основу для процессоров Pentium II и Pentium III. Те в конце концов вышли на 1,4 ГГц — более чем в десять раз превзойдя исходную частоту. А когда все уже решили, что ядро Pentium Pro забыто, оно возродилось в передовых процессорах Intel Core и Intel Core 2 как Pentium M, доказав мощный потенциал первоначальной архитектуры.

Но сейчас больше всего шума вызывает третья переработка — отчасти потому, что Intel расхвалила его как «самый важный серверный выпуск со времен Pentium Pro», но отчасти так оно и есть: Nehalem, последняя процессорная технология Intel, это громадный скачок вперед в разработке CPU. Она позволяет сисадминам разворачивать больше мощности на меньшем пространстве, причем существенно дешевле, чем всего год назад.

Серверная сторона

Когда Nehalem в прошлом году добрался до потребителей, он стал почти всеобщим любимцем. Мгновенная прибавка в скорости всегда приветствуется, а архитектура Nehalem содержит ряд изменений, обеспечивающих скачки в производительности без ощутимых затрат на энергопотребление. В большинстве случаев Nehalem действительно потребляет меньше мощности, чем его предшественник.

Владельцы серверов восторгались им еще больше, ведь потребление энергии — проблема серьезная. (Вспомним, что обеспечение серверов энергией и их охлаждение — самая большая расходная статья серверной, превышающая затраты на оборудование и персонал.) Новые процессоры Хеон, использующие Nehalem — модель 5500 — экономят столько энергии, что окупают себя менее чем за год. Представьте себе масштаб: взяв ферму из 200 серверов Intel Хеон четырехлетней давности и заменив их тем же количеством Хеон'ов серии 5500, вы получите производительность, большую в 10 раз, а потребление энергии снизится на 20 %. А если вы хотите резко снизить свои затраты, берите 25 Хеон'ов серии 5500, при этом потребление энергии снизится на 90 %, а вычислительная мощность будет та же.



» Когда мы спросили Кейта Паккарда о его работе в Intel, он сказал: «Самая крутая штука сейчас у нас в работе — это синий экран, где выводится информация о сбое».

»

Одна из причин, почему Nehalem настолько эффективен по энергопотреблению — он является вехой, отмечающей возврат Hyperthreading (HT), Intel'овского рыночного наименования для симметричной многопоточности их CPU. HT разрабатывался, чтобы улучшить работу чипов со сложным дизайном, то есть по возможности использовать простаивающие части CPU для других операций. Это не всегда работало — бывало, что процессоры первого поколения HT от Intel в некоторых случаях даже замедлялись. Однако второе поколение HT (начиная с SSE3) было недурным — при неудачном раскладе производительность падала до уровня не-HT чипа, а зачастую CPU превосходили своих лишенных HT соперников. Nehalem представляет третье поколение HT, созданное, чтобы одновременно задействовать еще больше частей CPU и расширить зону оптимальной работы.

Изящество HT состоит в том, что энергии потребуется не больше, чем без нее. Мало того, она может быть даже эффективнее «настоящей» многоядерной технологии, благодаря новой энергосберегающей функции, которая переводит бездействующие ядра в состояние без потребления энергии.

Конечно, ядро CPU, виртуализованное с использованием HT, никогда не обгонит настоящий CPU с той же тактовой частотой; здесь-то и выходит на сцену Linux. Видите ли, наивная реализация многопроцессорности одинаково обращается с каждым CPU, хоть виртуальным, хоть нет; для HT это, очевидно, не годится. Впервые кусок кода, распознающего процессоры Nehalem (его использует сам Линус), появился в ядре 2.6.27, но Intel вложил в написание кода ядер 2.6.28 и 2.6.29, чтобы извлечь из новой архитектуры максимум.

Такой оборотистости может достичь только свободное ПО, причем каждый, кто пользуется современным ядром, получает выгоду от Nehalem немедленно, не дожидаясь выхода новой версии ОС. Более того, некоторые из функций новейших ядер — а именно, мониторинг производительности и экономия энергопотребления для Nehalem — были портированы назад на более ранние ядра; то есть пользователи Red Hat Enterprise Linux 5, по-



» Стикеры vPro находятся на многих корпоративных машинах, однако некоторые сисадмины не знают, что именно скрывается за этим и просто игнорируют их. Увы!

лучив очередное обновление, наслаждаются эффективным энергопотреблением Nehalem.

Виртуализация ввода/вывода

Linux был на передовой в аппаратной виртуализации для x86 с момента ее появления несколько лет назад, за что опять-таки спасибо разработчикам свободного ПО, способным следовать мантре «выпускайте часто, выпускайте рано» и набрасываться на новые технологии, как только они возникают.

Теперь, благодаря технологиям VT-x от Intel и AMD-V от AMD, аппаратная виртуализация означает, что виртуальные процессоры смогут работать очень быстро. Это идеально для операций, привязанных к CPU, так как технология сейчас настолько хорошо отлажена, что CPU, работающий виртуально в Nehalem, не отстает от «го-

лого» Xeon'a прошлого поколения. Однако стоит виртуализации отойти от CPU, как неожиданно возникает проблема: если ваша сетевая или графическая карты были не в курсе виртуализации, они должны управляться программой, которая вызовет большое падение производительности. Если вас раздражало, что *Compiz* не работает на виртуальной машине, вообразите попытку запуска в ней рабочей станции для 3D-визуализации!

Сейчас два решения этой проблемы только начинают входить в мейнстрим, и опять Linux здесь в первых рядах. Первое решение в Intel окрестили Directed I/O или VT-d, а в AMD — просто IOMMU. На самом деле, второе название более точное — хотя обе технологии дают виртуальной машине прямой доступ к таким устройствам ввода/вывода, как видеокарта или сетевой адаптер, как если бы они были для нее родными.

Parallels уже использует VT-d в своем приложении *Parallels Desktop Extreme*, которое пока допускает только виртуализацию видеокарты. Так что если у компьютера внутри четыре GPU, можно назначить по штуке на три разных виртуальных машины, а последний приберечь для операционную систему хоста.

Прежние попытки решить эту проблему обычно применяли отображение вызовов OpenGL гостевой машины на вызовы OpenGL хостовой, что давало существенный выигрыш в скорости — до 40 % реальной производительности видеокарты. Одна-

«В области аппаратной виртуализации Linux всегда был впереди.»

Intel берется за GCC?

Компиляторы — это та область, где Intel конкурирует со свободным ПО: C Compiler от Intel известен своей невероятной производительностью по сравнению со старым добрым GCC, однако он не открыт, так что большая часть дистрибутивов Linux собирает свой код все-таки при помощи GCC.

Проблемой для Intel является то, что AMD немало вложил в GCC и *glibc*, обеспечив их оптимизацию для своих CPU, тогда как наилучшей производительности на CPU от Intel можно достичь только с родным компилятором Intel.

К счастью для нас, кажется, дело движется с мертвой точки. Пару недель назад трое служащих Intel начали процесс присое-

динения к командам разработчиков GCC, *glibc*, *binutils* и *gdb*. Если Intel пожелает поделиться наработками для компиляторов с GCC, то почти любая часть системы Linux — и практически все, что касается мультимедиа — увеличит свое быстродействие.

» GCC: вездесущий, но, увы, пока не самый эффективный компилятор.



ко при прямом доступе виртуальной машины к видеокарте она работает почти на полной скорости: задержка происходит только из-за тонкого слоя, предохраняющего хостовую ОС на случай, если виртуальная вдруг задуриет. *Parallel Desktop Extreme* поддерживает до 16 ядер и 64 ГБ ОЗУ на виртуальной машине, то есть огромное количество данных можно гонять целиком в виртуальном окружении. И, как вы, наверно, уже догадались, Linux был сертифицирован для использования с первого же дня, и как хост, и как гость.

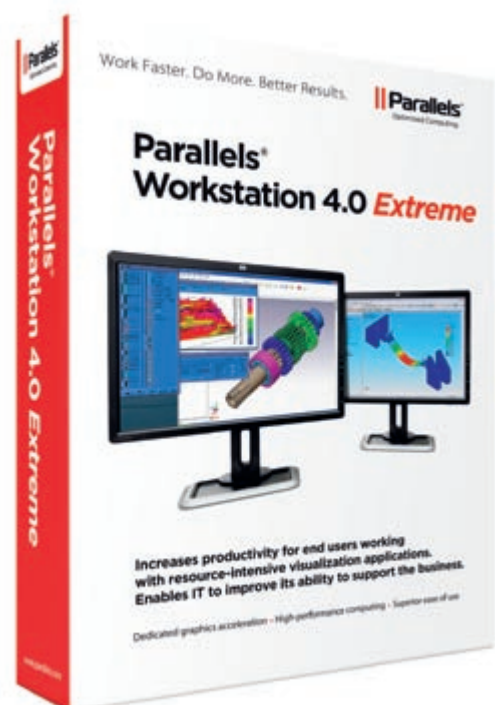
Второе решение проблемы виртуализированного ввода/вывода Intel назвал как VMDq, и у AMD нет соответствующего эквивалента, так как технология реализуется в сетевом оборудовании, а не в CPU. Приведем два примера:

» У вас есть одна ОС, работающая на «голом железе». При входе сетевого пакета он передается прямо в ОС, так как других вариантов нет.

» У вас есть одна ОС, работающая на «голом железе», но с двумя гостевыми виртуальными машинами. При входе сетевого пакета хост определяет, которой из них нужен этот пакет, затем виртуализирует его и передает гостевой ОС.

Второй случай представляет проблему — правда, небольшую: ведь у вас всего две виртуальные машины. А если их 16? Тогда ваша проблема разрастется — виртуализация такого количества ввода/вывода требует немалого объема работы CPU, в частности, из-за того, что придется размышлять над балансом загрузки и убеждаться, что какая-то из виртуальных машин не забирает слишком много сетевого времени.

VMDq делает так, что ваша сетевая карта знает о виртуализации. Таким образом, исключается ситуация, когда одно ядро CPU на хосте обрабатывает весь сетевой трафик. Вместо этого, у каждой виртуальной машины своя очередь на сетевое оборудование — она думает, что все как в жизни, и использует его обычным образом, но за кулисами сетевая карта автоматически тасует все запросы, чтобы гарантировать равноправие доступа к сетевому адаптеру. В результате снижаются требования к CPU и ускоряется работа с сетью — и, опять-таки, поддержка VMDq уже интегрирована в ядро Linux, то есть пользоваться ею можно уже сейчас.



» *Parallels Desktop Extreme* использует *Directed I/O*, чтобы дать виртуальным машинам выделенный доступ к графическому ускорению.



vPro: друг сисадмина

Последняя технология, которую мы рассмотрим, называется vPro, и хотя она не связана непосредственно с Linux (она сидит глубже ОС), но весьма популярна среди системных администраторов Linux, управляющих крупными парками компьютерной техники.

Вы могли видеть указание на vPro в виде наклейки на ноутбуке или других устройствах, но некоторые просто не понимают, что она делает — во многом потому, что она не имеет единой количественной задачи.

В сердце vPro лежит обычный CPU, но он все время активен и потребляет немного энергии, даже когда машина выключена. При подключении к корпоративной сети к этому CPU можно обратиться напрямую, если иметь пароль администратора, и считывать основную информацию о машине, о CPU, ОЗУ и емкости жесткого диска, без необходимости включать компьютер. По сути, это позволяет производить ночную инвентаризацию ресурсов, каждые сутки, без особых затрат энергии. При желании можно даже получить эту информацию по HTTP, но лучше оставить это для аварийных ситуаций.

Настоящее волшебство данной технологии в том, что Intel выпустил открытый код для Linux, который позволяет всем сохранять небольшие объемы данных — около 8 КБ за раз — в постоянном хранилище, которое можно будет читать при выключенной машине. Это значит, что вы можете записывать данные об обновлении системы или последнее обновление антивирусной базы на постоянное хранение и считывать его в процессе инвентаризации.

Но это только основы vPro. Поскольку эта платформа требует только материнскую плату Intel, CPU и сетевую карту, она способна на большее — вы получаете доступ к BIOS до загрузки или можете программировать правила фильтрации, отсеивающие сетевые пакеты перед тем, как они попадут к ОС, а то и аппаратные таймеры, запускающие механизм самозащиты, если определенное действие не происходит. Все эти функции возможны потому, что vPro сидит глубже, чем Linux, и в то же время обеспечивает резервный доступ к ОС; делайте с ним что хотите.

Заветная цель vPro состоит в максимальном уменьшении числа обращений в службу поддержки, так как вы сможете почти все сделать удаленно. Если дела действительно пойдут плохо, вы сможете перенаправить удаленную консоль на ваш локальный компьютер, получив полный контроль над системой, даже если SSH не работает. **LXF**

» vPro встроен в чип сетевой карты, а значит, никакие атаки хакеров не смогут обойти установленные вами правила.

Что за штука... Chromium?

Поклонники Google, Пол Хадсон припас для вас браузер...

» Google я, конечно, уважаю, но и Firefox люблю. Хотите уговорить меня сменить браузер?

Вообще-то я никого ни в чем не собираюсь убеждать. Но рассказать вам о новейшем программном проекте Google — да, собираюсь. А вы уж сами решайте.

» Ну, если так — слушаю вас.
Прекрасно! Итак, начнем: Chromium — проект нового браузера от Google, и...

» Постойте — я вроде что-то об этом слышал. Он не Chrome называется?
Chrome и вправду web-браузер, но так называется законченный, стабильный продукт для Windows. Chromium же — это название открытого проекта, лежащего в основе Chrome.

«Немногим ясно важность JavaScript для развития Интернета.»

» Ух ты, значит, код Chrome открыт?
А как же! Представляя проект, Google сразу объявил, что браузер будет свободным, ведь там очень много открытых компонентов. Ядро браузера основано на KHTML (он же WebKit).

» Ясно. Тогда любой сайт, работающий с KHTML, будет работать и с Chrome?
В общем, да, но есть принципиальное отличие: вместо движка JavaScript, включенного в WebKit (JavaScriptCore), Chrome использует свою собственную реализацию под названием V8.

» И естественно, что вне игры останутся многие сайты, использующие JavaScript?
Вы хотите сказать, Gmail?

» Вот имен... постойте-ка!
То-то и оно: знатокам JavaScript известно, что Google в этом вопросе ли-

дер. Поэтому V8 на 100 % совместим с любым из существующих web-сайтов.

» А зачем он вообще нужен?
Ключ дан в названии: V8 — это скорость. В момент выхода бета-версии Chrome, V8 как движок JavaScript обогнал всех с большим отрывом.

» Вы говорите так, словно с тех пор что-то изменилось...
Так и есть. V8 по-прежнему быстрее всех движков, но отрыв сократился: другие браузеры принялись подтягиваться.

» Значит, и Firefox ускорился благодаря Chromium?
Опосредованно. Программистам Firefox пришлось придать существенное ускорение TraceMonkey (прозвище нового движка JavaScript в Firefox 3.5). Теперь Firefox 3.5 выполняет сценарии JavaScript вдвое быстрее 3.0, а ведь разработка еще не закончена!

» Пожалуй, конкуренция реально приписывает прогресс...
Кто бы сомневался! Фактически ситуация обострилась настолько, что Microsoft, выпустив Internet Explorer 8, получила возможность заявить — причем обоснованно — что ее браузер быстрее существовавшей на тот момент версии Firefox. Apple тоже постаралась, чтобы новый Safari 4 нагнал, а кое-в чем и перегнал Chromium.

» Сколько же труда вбухивают в этот JavaScript...
Это правда, хотя немногим ясно важность JavaScript для развития Интернета. Все больше крупных приложений разрабатывается на JavaScript — возьмите хоть GoogleApps или Google Maps. Все больше сайтов (например, Facebook, Twitter или MySpace) используют JavaScript для управления обновлениями, анимирования и других эффектов, уже вошедших в обиход. И, конечно, большая часть рекламы тоже опирается на JavaScript.

» Не пойму, куда вы клоните.
Давайте посчитаем: если с каждого из сайтов, посещаемых вами ежедневно, загружается 300–500 КБ кода JavaScript, то объем работы, выполняемой браузером, набирается весьма внушительный.

» Хорошо, а почему же Chromium так быстр?
Отвечаю: V8 компилирует JavaScript прямо в машинный код. Этот метод отличается от стандартной строчечной интерпретации JavaScript, позволяя процессору исполнять сценарии почти со скоростью стандартных настольных приложений. Со временем в оптимизацию процесса трансляции JavaScript в родные инструкции процессора вкладывается все больше усилий, а значит, эффективность процедуры в целом постоянно повышается.

» Короче говоря, Chromium — быстрый браузер.
Он не просто быстрый: он очень быстрый, БЫСТРЫЙ, настолько быстрый, что...

» Хватит, хватит! Я понял, что он быстрый за счет JavaScript. Вы меня убедили, спасибо.
Но Chromium быстрый не только поэтому.

» Быстрее быстрого, или просто быстрый?
Быстрее, чем сказать «Вперед!». Благодаря отставке Gecko — неповоротливого HTML-движка Firefox — и переходу на легкий и скорый WebKit, Chromium не только быстро стартует, но и быстро загружает страницы. Ведь в нем используется технология «предварительных DNS-запросов» (DNS pre-fetching), которая сокращает время подключения к web-серверам. Кроме того, новый браузер быстрее в работе: при запуске вы видите миниатюры в духе Opera, с которым можно быстрее переходить на любимые web-сайты.

» У меня 4-ядерный Intel Core i7 и 8 ГБ ОЗУ. Скорости Firefox мне достаточ-



но, а большой разницы с *Chromium* я не вижу...

Что ж, не каждому так повезло с компьютером. Но *Chromium* славен не только скоростью.

» А я было подумал, что вы иссякли. Ладно, удивите меня: на что еще способен *Chromium*?

Как вам понравится такое: каждая вкладка в браузере помещается в отдельный тред.

» Куда, куда?

Ну, в тред — их еще потоками называют!.. Так, буду проще. Тред, или поток — это отдельный процесс на компьютере. Он действует самостоятельно и не влияет на другие процессы, выполняемые на компьютере одновременно с ним.

» И какая мне от этого польза?

Во-первых, если у вашего компьютера четыре ядра, то *Chromium* задействует их все одновременно. Во-вторых, если одна из вкладок даст сбой — например, потребует слишком много вычислительных ресурсов или вообще зависнет — это ничуть не повлияет на выполнение браузером остальных задач.

» А, вот оно что! Ненавижу, когда *Firefox* виснет. Это происходит чуть ли не на каждом *Flash*-ролике...

Точно. В *Chromium* самое серьезное, что может произойти — это отказ одной из вкладок. Все остальное остается в неприкосновенности!

» Ай да производительность — за одно это к *Chromium* стоит присмотреться.

Но и это не все! Браузер умеет работать в режиме инкогнито, скрывая историю и cookies, пока ему не скажут обратное. Инструмент поиска встроен прямо в адресную строку, а значит, сам Google будет подыскивать необходимые вам страницы уже во время ввода адреса. Возможно даже создание графических ссылок на рабочем столе: это упрощенные окна браузера, в которых сайты «живут» так, словно это обычные штатные приложения.

» Радужную картину рисуете. А что если все это лишнее? Не уверен, что мне нужно какое-то там «инкогнито».

Допустим. Но у *Chromium* масса мелочей, облегчающих жизнь. Скажем, улучшено автозавершение при вводе URL: оно теперь «помнит» только адреса, которые вы набирали вручную, исключив ситуации типа получения www.tuxradar.com/content/some-great-article-name-here/comments/slow?hello=world вместо www.tuxradar.com. Кроме того, интерфейс пользователя подчищен, чтобы остава-

лось больше места для сайтов. Например, при работе в полноэкранном режиме *Chromium* помещает корешки вкладок в область, где обычно находится рамка окна; строка состояния в пассивном режиме убирается.

» Вы меня убедили! Давайте адрес!

К несчастью, браузер под названием *Chromium* для Linux пока не вышел из стадии альфа.

» Так что ж вы мне голову морочите! Хотите подразнить технологией, которая мне недоступна?

И в мыслях не было! Учтите, мы здесь толкуем о вещах, про которые другие крутые парни и слыхом не слышали. А когда выйдет официальная версия *Chromium* для Linux, вы с умным видом порассуждаете об этом в пабе и всех удивите. Я прав?

» Ну... да. Но дайте хотя бы URL, где можно прочесть об этом подробнее!

А вот это запросто: домашняя страница *Chromium* — <http://code.google.com/chromium>. Если же вам не терпится опробовать новый браузер, хотя бы и в нынешнем нестабильном состоянии, то берите его из PPA Ubuntu: <https://launchpad.net/~chromium-daily/+archive/ppa>. Удачи! LXF



Д-р Крис Браун

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

По рецептам доктора Брауна

Добротное администрирование систем из причудливых заворотов кишок серверной.

Старый ворчун и его почта

Как вы поступаете с цепочками писем? Вы знаете, о чем я — об этих перекидываниях текста туда-обратно по ходу переписки. Мой личный подход — проявить вежливость к своим читателям, кратко резюмировав предыдущий разговор, прежде чем вставить свои пять копеек. Я не думаю, что в сообщении нужно цитировать все слова до одного, с самой первой подачи голоса. (Или, более тонко, я попросту вырезаю то, что не вписывается в мое сообщение!) Но большинство людей сразу тыкают в кнопку «Переслать» или «Ответить», постепенно накапливая в поле «Тема» сущее безобразие. Вот свежий пример из моей почты:

[Fwd: [Fwd Re: [Fwd: Re: [Fwd Re: Проект сервера — Время пошло]]]]

Самый вопиющий пример такого подхода — это письмо, в котором повторно 200 строк из предыдущей переписки и в конце добавлено что-то типа «Я согласен!». А еще мне на нервы действует то, что каждая почтовая программа считает своим долгом переформатировать сообщение, обычно вставляя совершенно ничем не оправданные пустые строки и символы >, так что простая фраза типа «Трава зеленая, а небо голубое» в конце концов превращается в

>>>> Трава

>>>> зеленая,

>>>> а небо

>>>> голубое

Зачем программы так делают? Может быть, чтобы оправдать свое существование?

Это напомнило мне кое-что. Если у вас есть пара минут, вы хотите улыбнуться и вы солидарны со мной насчет попыток программ оправдать свое существование, взгляните, например, на <http://miskatonic.net/fooo> как на крайний вариант страницы для ошибки 404.



LDAP с GUI

Luma и JeOS Просматривайте каталоги LDAP и создавайте виртуальные модули.

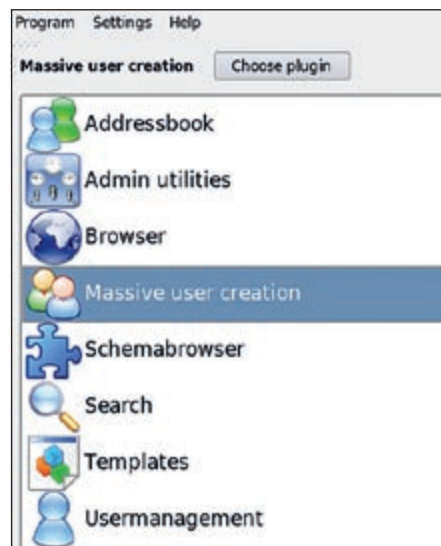
Если вы когда-нибудь пользовались OpenLDAP для организации службы каталогов, то почти наверняка сражались с загадочным синтаксисом его утилит — таких как *ldapadd*, *ldapsearch*, *ldapmodify* и т.д. Я сражался. Раньше я пользовался графическим браузером LDAP *GQ*, но в последних дистрибутивах Linux с ним проблемы.

Поэтому я был рад открыть для себя *Luma*, графический LDAP-браузер, написанный на Python (возьмите его с <http://luma.sourceforge.net> или с нашего DVD). Это расширяемая утилита с модульной архитектурой, в которой есть модули для просмотра схем, поиска, управления учетными записями пользователей или просто обзора сведений о каталогах LDAP. В модуле управления пользователями, например, добавление пользовательской записи в каталог организовано как заполнение формы, и это, конечно, гораздо удобнее обычных скриптов LDAP вроде *ldapuser*, *ldapaddgroup* и *ldappasswd*.

Сочный и сладкий

В этом месяце в поле моего зрения попала и JeOS, что расшифровывается как Just Enough Operation System и часто произносится как «juice» [сок]. Это облегченная версия Ubuntu Server с оптимизированным ядром: в ней есть только базовые компоненты, необходимые для запуска на виртуальном оборудовании, предоставляемом *VMWare ESX*, *VMWare Server* или *KVM*.

Целевой рынок JeOS — создание виртуальных модулей (virtual appliances). Базовая установка JeOS содержит минимальный набор пакетов,



В *Luma* есть различные модули для взаимодействия с каталогами LDAP.

занимает менее 300 МБ на жестком диске и требует всего 128 МБ оперативной памяти. В частности, графическая среда не загружается заранее. Идея состоит в том, что все необходимое вы установите из репозитория. (Конечно, некоторые вспомнят загрузку Unix с диска объемом в 2,5 МБ и 64 КБ оперативной памяти...)

JeOS прилагается к Ubuntu 8.04 в виде отдельного ISO-образа (см. <http://cdimage.ubuntu.com/jeos/releases/8.04>), но в версиях 8.10 и 9.04 она включена как опция на стандартном серверном CD.

Летите, голуби

Munin Отслеживает тенденции в загрузке сервера, предотвращая нехватку ресурсов до того, как она случится.

Недavno мне попалась программа для мониторинга *Munin*. Она наблюдает за сервером или несколькими серверами и строит графики изменений в их поведении в долгосрочной перспективе. В *Munin* используется архитектура «мастер–узел».

На главном узле *Munin* активизируется раз в пять минут (заботой *Cron*) и соединяется с утилитой *munin-node*, работающей на каждом из отслеживаемых узлов. По каждому узлу набор плагинов *Munin* собирает коллекцию характеристик загрузки сервера – среднюю загрузку, использование памяти, сетевой трафик, использование файловой системы, использование файла подкачки, почтовая активность и многое другое. Данные собираются и представляются в графическом виде в формате, подходящем для их доставки web-сервером системному администратору.

На серверной стороне у *Munin* есть довольно простой файл конфигурации **munin.conf**. Вот его минимальная версия:

```
# Здесь munin держит свою базу данных:
dbdir /var/lib/munin

# Сюда munin будет писать свой HTML-вывод:
htmldir /var/www/munin

# Здесь будут находиться файлы журналов munin:
logdir /var/log/munin

# Далее определяются узлы, подлежащие мониторингу:
[server1.example.com]
address 192.168.1.71
[server2.example.com]
address 192.168.1.73
```

Для переименования графиков или объединения графиков с разных клиентов есть и более причудливые конструкции.

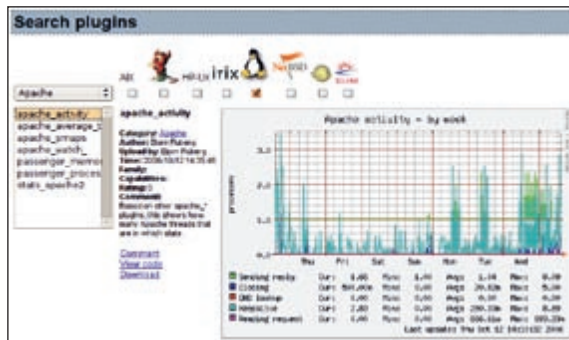
На клиентских компьютерах программа *munin-node* слушает соединения на порту 4949 и собирает данные с помощью набора плагинов – скриптов в каталоге **/usr/share/munin/plugins**. Большинство из них написаны на Perl, есть и сценарии оболочки; вообще они могут быть написаны на любом языке, позволяющем отправлять данные в стандартный вывод. *Munin* поставляется с большим набором модулей (почти 90 в моем Ubuntu, хотя только около двадцати активированы по умолчанию), а на «бирже плагинов» *Munin* по адресу <http://muninexchange.projects.llnpro.no> доступно еще больше.

Розыск в сети

На «бирже» есть поиск по категории, от *Amavis* (почтовый антивирус) до *Yum* (менеджер пакетов). Статистику можно собирать на самых различных данных, включающих (навскидку) свободное место на диске, уровень заряда батареи ноутбука, частоту ошибочных HTTP-ответов, температуру процессора, число активных подключений к серверу LDAP, число запросов к серверу DNS в секунду и многое другое.

На ушко...

Просто если вы подзабыли скандинавскую мифологию, напомним, что Хугин и Мунин – это два ворона: первый – мысль, а второй – память. На рассвете они вылетали, путешествовали по миру, узнавая, что происходит на свете, а вечером возвращались, садились на плечи верховному богу Одину и шептали ему на ушко новости.



► На «бирже плагинов» *Munin* множество написанных пользователями модулей; многие предоставляют предпросмотр, как этот.

Графики строятся за период в день, неделю, месяц, год, и главная сила *Munin* – в долгосрочных графиках, позволяющих легко оценить тенденции. Некоторые модули получают информацию из **/proc**, другие манипулируют данными от утилит мониторинга ресурсов типа *df*, *NetStat* и *vmstat*.

Рассмотрим плагины более подробно. Этот (сценарий оболочки) отслеживает размер очереди *Sendmail*:

```
MSP_QUEUE=/var/spool/mqueue-client
MTA_QUEUE=/var/spool/mqueue
if [ "$MSP_QUEUE" ]; then MSP_QUEUE=$MSP_QUEUE ; fi
if [ "$MTA_QUEUE" ]; then MTA_QUEUE=$MSP_QUEUE ; fi
if [ "$1" = "autoconf" ]; then
if [ -d $MSP_QUEUE -a -d $MTA_QUEUE ]; then
echo yes
exit 0
else
echo no
exit 1
fi
fi
if [ "$1" = "config" ]; then
echo 'graph_title Sendmail queued mails'
echo 'graph_order mails'
echo 'graph_vlabel mails in queue'
echo 'graph_category sendmail'
echo 'mails.label mails'
exit 0
fi
```

```
mspmails=`find $MSP_QUEUE -type f -name '[qQ]*' 2>/dev/null | wc -l`
mtamails=`find $MTA_QUEUE -type f -name '[qQ]*' 2>/dev/null | wc -l`
echo "mails.value `expr $mspmails + $mtamails`"
```

Главное, что нужно отметить – модуль, вызванный с аргументом **config**, выводит текстовое описание графика, с помощью которого внутренняя утилита *rrdtool* построит его. Если он вызывается без аргумента, то выводит строку вида

```
mails.value 47
```

Взглянув на три последние строки скрипта, вы увидите, что он вычисляет результат, просто подсчитывая количество файлов в очередях в каталогах **/var/spool/mqueue-client** и **/var/spool/mqueue**. Руководство по написанию и установке модулей можно найти по адресу <http://muninprojects.llnpro.no/wiki/HowToWritePlugins>.

Открываем DHCP

DHCP В руководстве «назад к основам» этого месяца мы поговорим о DHCP и узнаем, откуда ваш ноутбук, по всей вероятности, получил свой IP-адрес.

DHCP – сокращение от **Dynamic Host Configuration Protocol** (протокол динамической настройки хостов). Эта технология позволяет назначать компьютерам сетевую конфигурацию «на лету» вместо того, чтобы жестко прописывать ее в статических параметрах машины. Эти настройки назначаются в процессе загрузки клиента, а точнее, в момент запуска его сетевых интерфейсов. Наверное, самая очевидная из них – это IP-адрес, но DHCP может предоставить и другие, например, маску подсети и шлюз по умолчанию, и многие настройки уровня приложений, такие как адреса первичного и вторичного DNS-серверов, серверы времени NTP и серверы TFTP для загрузки PXE.

DHCP-сервер не назначает настройки навсегда, а сдает их в аренду на определенный срок. Клиент указывает этот срок в запросе, но максимальное время аренды в настройках сервера может быть меньше запрошенного клиентом. При желании продолжить пользоваться настройками, клиент должен повторно отправить запрос до того, как его время истечет. Если клиент этого не сделал, сервер по истечении срока аренды вправе назначить

его IP-адрес другому клиенту, хотя передача ранее назначенного адреса – вещь не столь простая. Сначала сервер прове-

«В былые времена IP-адреса могли быть дефицитом.»

рит, не используется ли этот адрес кем-то другим. Клиент также может явно отказаться от аренды DHCP и должен это сделать, если его сетевые интерфейсы корректно остановлены.

В былые времена, до появления NAT и использования пространства частных IP-адресов 192.168.0.0/24 во внутренних сетях, IP-адреса могли быть дефицитным ресурсом, и установка короткого срока аренды (например, час или 10 минут) означало, что сеть могла растягивать свои скудные запасы IP-адресов, назначая их только компьютерам, активным в данный момент. Сейчас это меньшая проблема, и вы скорее увидите сроки аренды в день или даже неделю; тем не менее, этот механизм остается удобным для повторного использования адресов выключенных компьютеров. Классический сценарий – приходит человек из вашего офиса

в Бобруйске, на денек подключает ноутбук к вашей сети, а потом едет обратно в Бобруйск.

Тем не менее, DHCP-серверы стараются выдавать тот же самый IP-адрес и те же настройки, которые были назначены в прошлый раз, клиенту, который периодически возникает в сети. Для этого они записывают сроки аренды в файл, где они привязываются к MAC-адресам клиентов.

Все это напоминает мне студенческие годы, когда книг в библиотеке было гораздо меньше, чем студентов. Однако взять всегда было что, при достаточно быстром обороте.

Если взглянуть на это с клиентской стороны, то у развездного сотрудника офиса появляется еще более удобный способ использования DHCP. Он просто включает свой компьютер в ту сеть, рядом с которой оказывается, и получает подходящие настройки для работы в ней. Ему не нужно возиться с диалогами, чтобы ввести подходящие IP-адрес, адрес сервера DNS и так далее.

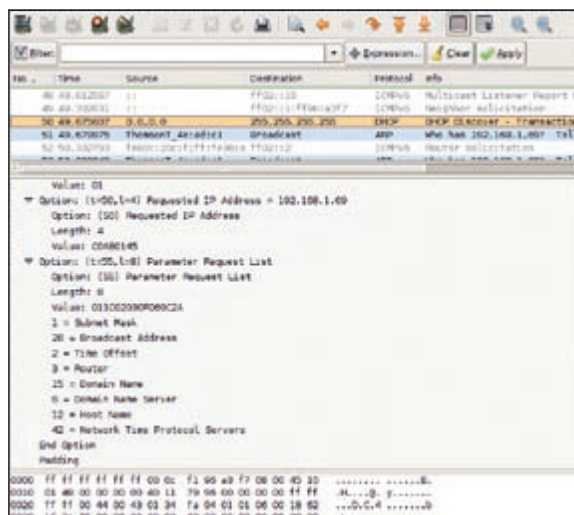
Сперва найдем свой сервер

На рисунке внизу слева показан типичный DHCP-обмен при запуске сетевого интерфейса. Клиент начинает с отправки пакета DHCPDISCOVER. Этот пакет широковещательный, так как клиент не знает адреса DHCP-сервера (он даже собственного адреса не знает!). DHCP-сервер по адресу 192.168.0.1 отвечает, предлагая срок аренды, и клиент отправляет второй пакет, с запросом на аренду. Наконец, сервер отвечает пакетом DHCPACK, предоставляя клиенту адрес 192.168.0.3, который затем связывается с сетевым интерфейсом (в данном случае, eth2):

```
$ sudo ifup eth2
Internet Systems Consortium DHCP Client V3.0.4
Copyright 2004-2006 Internet Systems Consortium.
All rights reserved.
For info, please visit http://www.isc.org/sw/dhcp/
Listening on LPF/eth2/00:10:60:60:3e:8e
Sending on LPF/eth2/00:10:60:60:3e:8e
Sending on Socket/fallback
DHCPDISCOVER on eth2 to 255.255.255.255 port 67
interval 5
DHCPOFFER from 192.168.0.1
DHCPREQUEST on eth2 to 255.255.255.255 port 67
DHCPACK from 192.168.0.1
bound to 192.168.0.3 -- renewal in 97583 seconds.
```

На рисунке сверху справа показан этот обмен, а на экранном снимке *Wireshark* пакет разобран детально. Кто хочет рассмотреть

► Разбор пакета DHCPDISCOVER. Обратите внимание на широковещательный адрес назначения (255.255.255.255) и список параметров запроса.



Провайдер Dynamic DNS

Это немного не в тему, но если ваш провайдер назначает видимые извне IP-адреса по DHCP, то нет гарантии, что адрес каждый раз будет одним и тем же, поэтому вы не можете зарегистрировать в DNS статическую запись. Некоторые провайдеры предоставят вам статический адрес (обычно за дополнительную плату); или можно воспользоваться услугами провайдера Dynamic DNS (например, www.dyndns.com), чтобы зарегистрировать имя компьютера, которое будет постоянным всегда, даже если ваш IP-адрес изменился. Для этого на компьютер потребуется установить клиент DDNS, например, *ddclient*.

Поправка LTSP

В LXFI17 я писал о Linux Terminal Server Project и упоминал Atlanta Public Schools как пользователя этой технологии, назвав цифру в 2200 клиентов. Мы получили письмо от Джеймса Швайцера [James Schweitzer] с новыми данными – теперь у них 12000 тонких клиентов, обслуживаемых 150 серверами в 54 школах. Даже если не брать во внимание сами цифры, число клиентов на сервер меня впечатляет. Спасибо, Джеймс!

DHCP-обмен более подробно, обратите внимание на *dhcpcd*, утилиту пост-обработки вывода *tcpdump* для анализа перехваченных DHCP-пакетов.

Конфигурация сервера

Программа *dhcpc3* (версия 3 реализации DHCP от Internet Software Consortium) – широко используемый сервер DHCP. Рассмотрим самые важные фрагменты его конфигурационного файла (скорее всего, */etc/dhcp3/dhcpd.conf*). В начале мы видим глобальные определения, применяемые ко всем подсетям, которые обслуживает сервер:

```
option domain-name "example.com";
option domain-name-servers 192.168.122.1;
default-lease-time 600;
max-lease-time 7200;
```

Эти строки определяют, какую информацию сервер передаст клиенту в DHCPACK, тут все понятно.

Следующий фрагмент определяет одну подсеть, которую будет обслуживать сервер. Предполагается, что *dhcpc3* слушает порт на интерфейсе, подключенном к сети 192.168.122.0:

```
subnet 192.168.122.0 netmask 255.255.255.0 {
range 192.168.122.20 192.168.122.190;
option broadcast-address 192.168.122.255;
option routers 192.168.122.1;
}
```

Здесь мы видим диапазон IP-адресов, назначаемых сервером, а также настройки подсети для широковещательного адреса и шлюза по умолчанию, которые будут переданы клиентам этой подсети.

Можно также привязать настройки (включая имя хоста и фиксированный IP-адрес) к заданному хосту по его MAC-адресу – фрагмент ниже показывает, как это сделать:

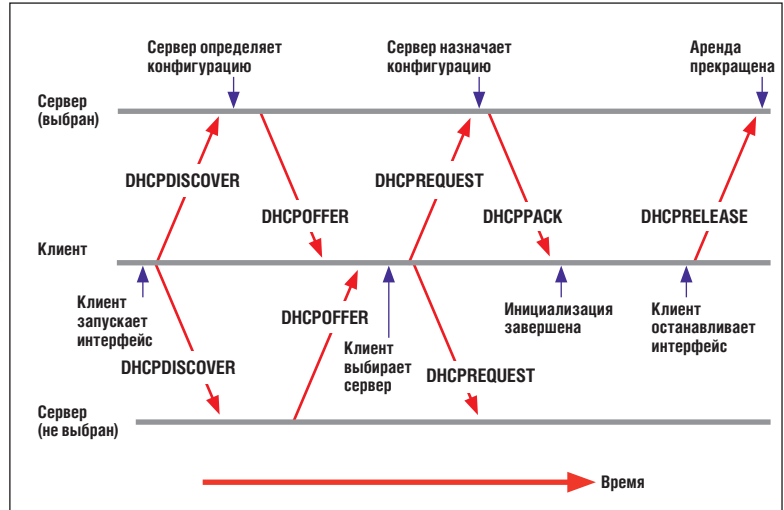
```
host lxfdesktop14 {
hardware ethernet 52:54:00:12:34:56;
option host-name "lxfdesktop14";
fixed-address 192.168.122.181;
next-server 192.168.122.1;
filename "webserver/pxe/linux.0";
}
```

Параметры *next-server* и *filename* нужны для загрузки клиента по сети (PXE). Они задают IP-адрес TFTP-сервера и имя файла, который будет с него запрошен.

Подружим DHCP и DNS

У DNS и DHCP непростые отношения. И проблема в том, что DNS создавался из расчета, что IP-адрес компьютера остается постоянным достаточно долго (как минимум, несколько дней или недель), а DHCP идет вразрез с данным предположением. Чтобы решить эту проблему, динамический DNS позволяет сообщить связанному с ним DNS-серверу назначенный клиенту адрес. Этот протокольный метод обновления DNS описан в RFC 2136. Вот типичный фрагмент конфигурационного файла DHCP, позволяющего реализовать такое обновление:

```
ddns-update-style interim;
```



► Временная диаграмма обмена пакетами DHCP. В этом примере два сервера DHCP предлагают аренду, и клиент выбирает первый отозвавшийся.

```
ddns-domainname "example.com";
ddns-rev-domainname "in-addr.arpa";
update-static-leases on;
zone example.com. {
primary 192.168.122.6; #update the local server
}
zone 122.168.192.in-addr.arpa. {
primary 192.168.122.6; #same for reverse addresses
}
```

Конечно, на стороне DNS-сервера для динамического обновления нужно настроить еще кое-что, но здесь мы не будем в это углубляться.

Так как для поиска DHCP-сервера клиенты отправляют широковещательные запросы, а широковещательные пакеты скорее всего не пройдут через маршрутизаторы вашей сети, в каждой подсети должен быть DHCP-сервер. Альтернатива – запустить в подсети DHCP-ретранслятор (*dhcrelay*). Он слушает DHCP-запросы и перенаправляет их серверам DHCP из заданного списка. Так с помощью одного DHCP-сервера обеспечивается выделение адресов для всей сети.

Наконец, поскольку DHCP – критически важный сервис сети, вы можете задуматься о создании для него резервного. В случае с двумя серверами, адресные пространства которых не перекрываются, это довольно просто, но конфигурация реальной отработки отката с основного сервера на запасной, обрабатывающий тот же самый набор адресов, требует больше усилий и предполагает специальную секцию в файле конфигурации обоих серверов. Поддержка имеется только в третьей версии DHCP. Документация не особенно хороша, но по адресу www.madboa.com/geek/dhcp-failover есть неплохая статья. А если вам нужно официальное описание DHCP, прочтите RFC2131 (<http://left.org/rfc/rfc2131.txt>). **LXF**

Один — компания, два — толпа

Два конфликтующих DHCP-сервера в одной сети – это Плохая Идея: очевидно, что между ними начнутся гонки. С годами я обнаружил, что установка неконтролируемых DHCP-серверов (случайно или намеренно) в чужой сети – отличный способ позлить системного администратора. Чтобы заметить это, нужно время (должны истечь существующие сроки аренды или в сети должен появиться новичок), но в конце концов администратор будет ходить вокруг с обиженным видом, держа в руках свой ноутбук с обличительной трассировкой пакетов, и бормотать: «Кто получил 192.168.122.42?»

softline®

Софт со всего света



Сотрудничайте с нами в 53 городах 16 стран:



Москва
Санкт-Петербург
Архангельск
Барнаул
Владивосток
Волгоград
Воронеж
Екатеринбург
Ижевск
Иркутск
Казань
Калининград
Кемерово
Краснодар
Красноярск
Набережные
Челны

Нижний
Новгород
Новосибирск
Омск
Оренбург
Пермь
Ростов-на-Дону
Самара
Саратов
Сыктывкар
Томск
Тюмень
Ульяновск
Уфа
Хабаровск
Челябинск
Ярославль



Минск
Гомель
Витебск
Киев
Харьков
Алматы
Астана
Актобе
Караганда
Ашгабад
Бишкек
Баку
Душанбе
Ереван



Тбилиси
Ташкент
Каракас
Стамбул
Тегеран
Улан-Батор
Ханой

www.softline.ru

Москва, ул. Губкина, д. 8

E-mail: info@softline.ru

(495) **232-00-23**

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Начинал с Агатов.
Когда-то даже знал,
что такое Робик.

Патенты — зло?

Идея принять закон о том, что одни алгоритмы являются математикой, а другие — нет, является абсурдной.
Дональд Э. Кнут

Патент — охранный документ, удостоверяющий исключительное право, авторство и приоритет изобретения, полезной модели или промышленного образца. Впервые был выдан в Венецианской Республике аж в XV веке, с целью распространения технической информации, так как должен был исчерпывающе описывать изобретение. Чтобы авторы не скрывали от общества свои наработки, в патент было добавлено исключительное право на пользование изобретением в течение десяти лет.

В XV веке это был мощный толчок, или даже пинок, к прогрессу, ибо цеховые традиции диктовали необходимость сокрытия всех новшеств от постороннего взгляда. Внедрение патентов позволило сломать замкнутый круг «владельцев» технологий, предоставив их всем желающим. Но в средние века все развивалось гораздо медленнее, чем сейчас, да и темп жизни был совершенно другой. Казалось бы, для сохранения инварианта полезности следовало значительно уменьшить (не оставить неизменными, не увеличивать, а уменьшить!) сроки действия патентов, но уже почти никто не думает о том, ради чего все это затевалось, и из двигателя прогресса патенты превратились в могучий тормоз.

P.S. Дональд Э. Кнут подал прошение в расширенную Комиссию по Апелляциям Европейской патентной организации против патентов на алгоритмы. Видимо, достали...
E.m.Baldin@inp.nsk.su

В ЭТОМ ВЫПУСКЕ...



Освободи мышление 50
Долой бумажки-наклейки: **Basket** и **Энди Ченнел** предлагают куда как лучший способ содержать мысли в порядке. Плюс: управление структурой документа в *OpenOffice.org*.



Программируем в облаках 58
Вы просили — мы сделали. **Евгений Крестников** научит вас, как управлять документами в Google Docs через Google Data API.



Питонизируем Сеть 66
Свободный как Linux **Ник Вейч** и его ручной змей с легкостью заставят ваш рабочий стол соответствовать погоде за окном...



Баг не пройдет! 72
Отлаживание — это, как известно, процесс избавления программы от... ошибок. **Андрей Боровский** продемонстрирует его на примере *GNU Debugger* и *Cuneiform*.



Интернет-телефония 80
Любите акронимы и не любите платить по счетам за телефон? У **Майкла Брукса** есть как раз то, что вам нужно — PBX VoIP под названием *Asterisk*.



Нарисуй мечту! ... 54
О, кошмар! Взяв новенький планшет Wacom Bambo и *GIMP*, **Майкл Дж. Хэммел** недрогнувшей рукой превращает цветущий город в адские руины.



Ваша почта, сэр! 62
Считаете локальные почтовые клиенты пережитком эпохи Web 1.0? **Нейл Ботвик** сделает их сетевыми и доступными в браузере из любого завалающего интернет-кафе.



Общаемся с VirtualBox 70
...но Python способен и на большее. **Николай Иготти** применяет его для автоматизированного управления виртуальными машинами.



Графический JavaFX 76
Пока что наша технология для создания продвинутых интернет-приложений выглядела не такой уж продвинутой. **Антон Черноусов** спешит исправить этот недостаток.

Совет месяца: Пожиратели трафика



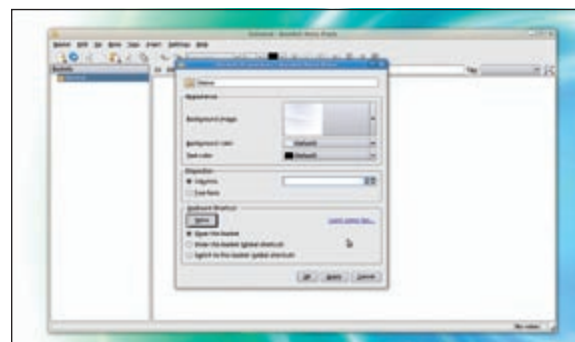
Случалось ли вам замечать, что интернет-соединение стало медленным, хотя вы ничего не скачиваете? Асимметричная природа беспроводных широкополосных каналов такова, что если вы занимаете всю исходящую полосу, загрузить что-либо становится нереальным. Причина кроется в способе управления трафиком в вашем модеме или у интернет-провайдера. Даже самые нетребовательные и малоскоростные операции, вроде удаленной оболочки или DNS-запросов, становятся невообразимо медленными или завершаются по тайм-ауту. Если вы используете что-то типа клиента BitTorrent, можно ограничить в нем скорость выгрузки и устранить проблему. Некоторые другие программы, например, *rsync*, тоже предлагают подобную функциональность, но многие ею не озабочиваются. Одновременный запуск двух таких прило-

жений тоже может стать проблемой, если установить для каждого из них ограничение в 90 % от ширины канала.

Одним из решений является удобный сценарий *Wonder Shaper*, использующий команду *tc* (traffic control) для ограничения общей пропускной способности на уровне чуть ниже максимального. Загрузите его с <http://lartc.org/wondershaper>, поместите скрипт *wshaper* куда-то в пределах PATH — скажем, */usr/local/bin*, и отредактируйте системные параметры в начале. Установите значения DOWNLINK и UPLINK чуть ниже максимума (в кбит/сек) и запустите программу. Вы увидите, что операции вроде выгрузки фотографий во Flickr больше не тормозят систему. Подобрать желаемые настройки, добавьте вызов *wshaper* при загрузке дистрибутива.

Basket: Соберем

Храните свои идеи в виде рожицы из стикеров? Тогда позвольте **Энди Ченнелу** показать, как *Basket Note Pads* преобразит вашу жизнь...

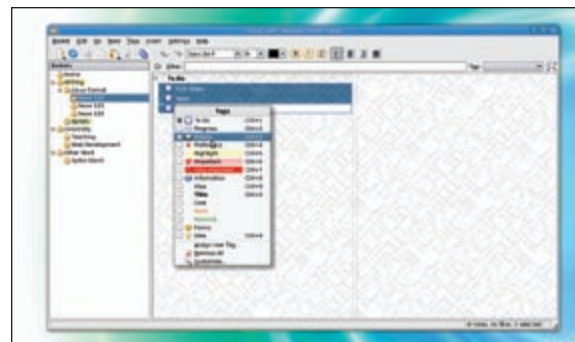


» Индивидуальные стили корзин помогут разграничить сферы вашей жизни.

щелкните на них правой кнопкой мыши и удалите, чтобы начать с чистого листа. В конечном счете должна остаться только корзина General [Общее]. Переименуйте ее – щелкните на ней правой кнопкой мыши, выберите Properties [Свойства] и введите новое имя; мы вписали Home. В этом диалоговом окне можно настроить внешний вид, добавив столько колонок, сколько нужно, или выбрав свободную форму – но учтите, тогда страницы получатся не шибко упорядоченные. Еще можно определить цвет текста и фона корзины, а также назначить комбинацию клавиш для ее быстрого открытия.

Для нашего раздела Home мы выбрали раскладку в две колонки и вернулись в главное окно, где видна граница между двумя этими столбцами. Границу можно взять и перетащить вправо или влево, чтобы изменить ширину колонок.

Теперь добавим вторую корзину, щелкнув в любом месте панели дерева правой кнопкой мыши и выбрав New [Создать]. По умолчанию создается корзина первого уровня (корневая), но вы можете определить ей родительскую через выпадающее меню внизу окна корзины, которое также настраивает их раскладку.



» Тэги — удобные индикаторы статуса задач, и пользы ради можно добавлять более одного.



Наш эксперт

Энди Ченнел

Энди, видимо, будет делать Первые шаги в Linux вечно, а технологиями он заинтересовался, открыв для себя Dragon 32.

Число источников данных в мире все растет, и управление своими собственными мыслями, плюс еще горы информации, становится неподъемным. К счастью, есть целый ряд отличных приложений для заметок, способных хранить ваши соображения, идеи, списки и ссылки.

Мы решили взять программу под названием *Basket Note Pads*. Это приложение KDE, но оно работает и в дистрибутивах на базе Gnome. На данном уроке мы покажем, как организовать корзину [не путайте с той корзиной, куда попадают удаленные файлы! – прим. ред.] и добавить в нее всевозможное содержимое, а завершим кратким обзором надежности *Basket*.

Для начала через менеджер пакетов *Synaptic* загрузим *Basket* версии 1.0.2. После этого она окажется в разделе Дополнительные в меню Приложения. Кроме того, запущенное приложение поместит иконку в системный лоток – щелкайте по ней, когда *Basket* вновь понадобится. Интерфейс программы состоит из трех частей. У верхнего края – меню и панели инструментов; по левую сторону размещается панель-дерево для организации коллекции корзин и подкорзин. И, наконец, главное окно – для управления отдельными корзинами информации.

По корзинам

Начнем с создания небольшого набора корзин для различных сфер вашей жизни, которые, возможно, нуждаются в реорганизации. Несколько элементов в древе корзин уже имеется, поэтому

» **Месяц назад** Мы создали слайд-шоу в *Smile* и поместили кое-что на YouTube.

МЫСЛИ В КУЧКУ

Настало время подумать, как разграничить вашу жизнь, и создать корзину первого уровня для каждой части. Возможно, вы захотите задать форматирование для каждой из частей в отдельности, чтобы выделить их. Если вы собираетесь использовать несколько корзин, может также потребоваться изменить свойства каждой из них (правый щелчок > Properties) и определить фоновые цвета в тон и новые иконки (просто щелкните на пиктограмме корзины рядом с именем).

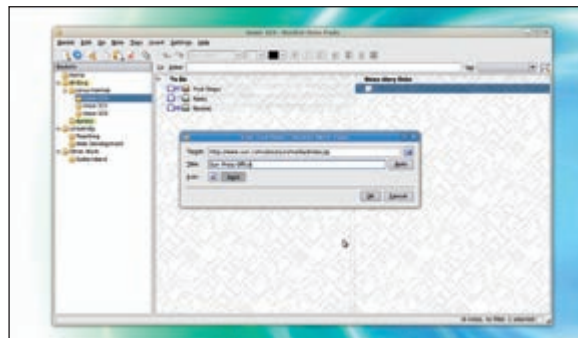
Каждую корзину можно населить подкорзинами (элементами второго уровня), щелкнув на ее имени правой кнопкой мыши и выбрав New > New Sub-Basket [Создать > Новая подкорзина]. Подкорзины наследуют свойства родителей, и при разных оформлениях у предков легко определить, где вы находитесь в данный момент.

Что дальше

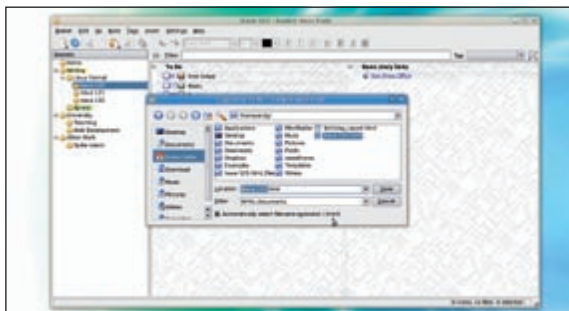
Пора добавлять содержимое. Начнем с создания списка задач. Выберите корзину и щелкните где-нибудь в главной рабочей области – создастся контейнер. Введите заголовок, а затем щелкните вне контейнера и вновь подведите курсор мыши к новому заголовку. Элемент станет выделенным, и слева от вашего текста появится стрелочка. Щелкните на ней и выберите из доступных пунктов Title [Заголовок].

Теперь щелкните под заголовком и начинайте вводить первый элемент списка ваших задач. Вновь щелкните вне нового элемента и наведите на него курсор мыши, затем в выпадающем меню выберите тэг To Do [Выполнить]. Рядом с элементом автоматически появится флажок. Щелчок на флажке отключит пункт и перечеркнет его. Щелкните еще раз и добавьте новые элементы списка задач под уже созданным. Заметьте, что приложение чутко реагирует на расположение мыши: например, при щелчке чуть ниже элемента списка задач оно сочтет, что вы желаете добавить пункт в список, и отформатирует его соответствующим образом без всяких усилий с вашей стороны. При щелчке подальше *Basket* решит, что вы перешли к другому действию, и создаст стандартную область.

После создания списка щелкните на элементе-заголовке, полученном ранее, а затем выполните Shift+щелчок на последнем элементе списка. Щелкните правой кнопкой и выберите Group [Группировать]. Это позволит перемещать весь список как единое целое и, задержав мышью над заголовком, сворачивать его в одну строку.



➤ Создавайте ссылки на web-сайты, локальные файлы или даже приложения.



➤ Экспорт в HTML – прекрасный способ создания универсальной копии ваших корзин.

Кроме простейших заголовков, приличные списки задач предоставляют пользователям сведения о ходе выполнения отдельных пунктов. Для присвоения индикаторов элементам списка, щелкните на первом из них, затем выполните Shift+щелчок на последнем элементе и выберите в выпадающем списке пиктограмму Priority [Приоритет]. Вам будет предоставлено три варианта на выбор – Low [Низкий], Medium [Средний] и High [Высокий]; их можно установить правым щелчком на иконке рядом с именем каждого элемента. Таким же образом добавляется индикатор прогресса [Progress] (щелчок на нем увеличит его состояние), он позволяет визуально оценить процент готовности пункта.

Единственное важное упущение в этом разделе – поля даты или сроков; их придется добавить в заголовок.

Упорядочиваем все

Хотя списки задач – это здорово, вы скоро осознаете, что с *Basket Note Pads* можно делать намного больше. Например, если в связи с неким проектом вы регулярно посещаете несколько web-сайтов, их можно собрать отдельно в корзине проекта. Просто создайте заголовок (как указано выше) и в панели меню выберите Insert > Link [Вставка > Ссылка]. Теперь добавьте web-адрес и заголовок для каждого сайта, который необходимо вставить. Сгруппируйте их вместе для упрощения организации, и у вас получится настоящая система закладок, специфичная для задачи. Аналогично можно добавлять изображения, длинные заметки и ссылки на приложения. А те, кто занимается разработкой, могут добавлять к заметкам куски кода или создавать элементы, основанные на шестнадцатеричном цветокоде. Например, можно создать корзину для web-проекта и определить палитру сайта внутри нее. Экспериментируйте с базовыми темами, которые мы описали, и вы, несомненно, скоро поймете, что *Basket* – ценнейший инструмент.

Скорая помощь
Используйте Ctrl+Y для быстрой вставки ссылки в корзину.

Резервируйте корзины

Чем дольше вы пользуетесь *Basket*, тем сильнее желание обеспечить сохранность предыдущих мыслей и идей, благодаря резервным копиям. Для этого выберите в меню Basket > Backup & Restore [Корзина > Сохранить или восстановить]. Выполнится именно то, чего вы и ждали: создание резервной копии или восстановление из нее. Автоматического резервирования нет, все это необходимо регулярно проделывать вручную.

Если вы решили убрать старую корзину из дерева – например, завершив проект – можно создать ее архив: Basket > Export > Basket Archive [Корзина > Экспорт > Архив корзины]. Альтернатива – экспорт корзины в виде HTML-страницы через Basket > Export > HTML Web Page. При выборе последней опции создастся снимок корзины, с оформлением и ссылками, доступными для web-браузера. Превосходное решение для архивирования!

OpenOffice.org:

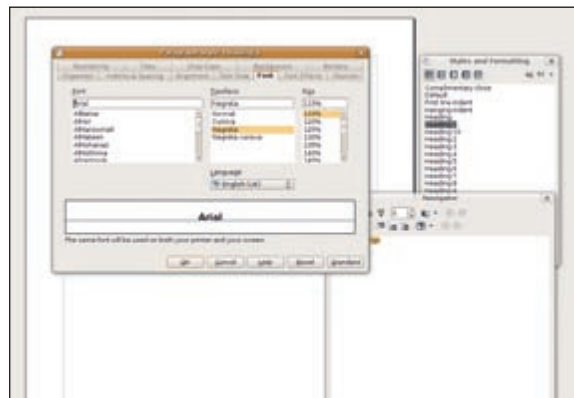
Не ограничивайтесь жесткой структурой – пусть *Writer* придаст вам свободы.

Планировщики обычно используются для создания структуры документа – будь это роман, отчет или письмо – еще до его написания. Концентрируясь на структуре, обдумывая процесс, писатель получает лучшее представление о развитии проекта, да и вообще это один из способов борьбы с «синдромом белого листа», одолевающим многих авторов. В нашем уроке мы создадим структуру в текстовом процессоре *OpenOffice.org Writer*, затем применим приложение для переупорядочивания элементов текста на различных уровнях, и, наконец, экспортируем ее для использования в качестве основы презентации *Impress*.

OpenOffice.org установлен по умолчанию во многих настольных дистрибутивах. Мы используем последнюю версию 3.x, но инструмент создания структур в 2.x практически такой же. То есть старая версия, вероятнее всего, позволит вам повторить все, что мы здесь проделаем.

Инструмент Стили использует стили абзацев, уже имеющиеся в *Writer* для определения иерархии документа. Определить свой стиль и взять его за основу структуры весьма просто, но будем уж держаться того, что есть. Для начала заглянем в Сервис > Структура нумерации [Tools > Outline Numbering]. В левой части окна вы видите различные уровни. Выбор одного из этих чисел показывает стиль абзаца, связанного с ним. Например, Заголовок 1 – элемент высшего уровня, и все, что отформатировано в этом стиле, также становится заголовком высшего уровня.

При написании романа или эссе можно использовать этот стиль абзаца в названиях глав. Для этого надо ввести заголовки



► Планировщик использует Стили и Форматирование *OOo*, и можно использовать ту же структуру для создания оглавления.

глав, выделить их все и выбрать Заголовок 1 в выпадающем списке Стил – он находится в верхней части окна слева (или в окне Стили, вызываемом выбором Формат > Стили).

Применив тот же метод, но выбрав в качестве формата Заголовок 2, можно измельчить структуру каждой главы, вынося наши мысли или идеи в отдельные строки. Под этими разделами можно вводить заметки, диалоги или прозу, или добавлять цитаты и ссылки на источники в эссе.

Пишем, чтобы править

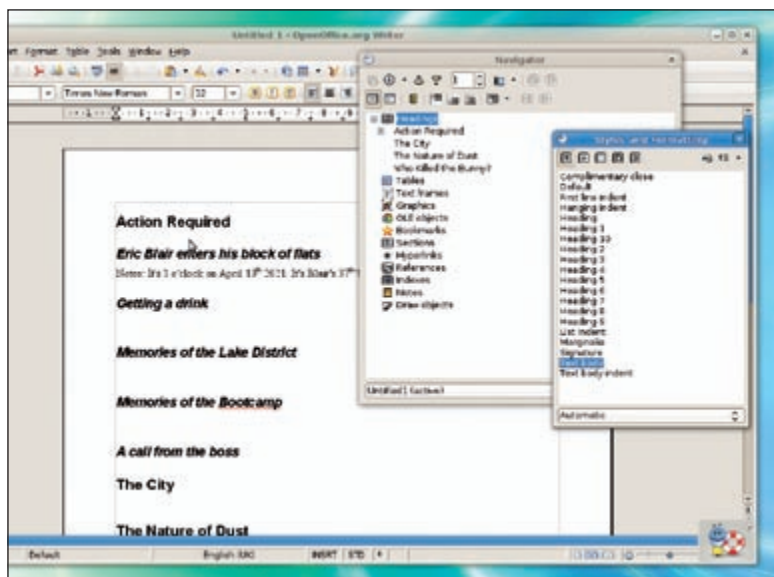
Нужно принять решение о глубине структуры, поскольку каждый элемент, который вы захотите переместить позднее (мы здесь не говорим об операциях «вырезать» и «вставить»), должен быть отформатирован при помощи одного из ее стилей. Пусть, например, отдельные сцены рассказа написаны и расположены под заголовками с определенными стилями. Тогда потом вы сможете изменить порядок сюжета, просто поменяв положение заголовков сцен, поскольку при перемещении элемента структуры его дочерние элементы также переносятся.

Вдобавок к возможности подправить структуру проекта до отправки на печать, использование подобной иерархии разбивает задачу написания большого труда на более короткие куски, повышая управляемость.

Упрощая написание, созданная в начале работы структура облегчит также и улучшение внешнего вида документов, особенно благодаря тому, что стили заголовков и подзаголовков можно редактировать в окне Стили (для просмотра доступных опций выполните правый щелчок на имени стиля и выберите Изменить). То есть важные стилистические изменения выполняются с минимумом усилий. Другое достоинство хорошей структуры – простое создание оглавления готового документа: благодаря стилям заголовки и подзаголовки легко распознать.

Скорая помощь

При использовании Планировщика остерегайтесь ошибок в преемственности элементов. Очень легко сбиться с пути по сюжету, когда все переплелось.



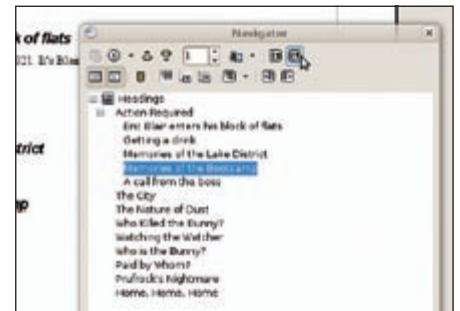
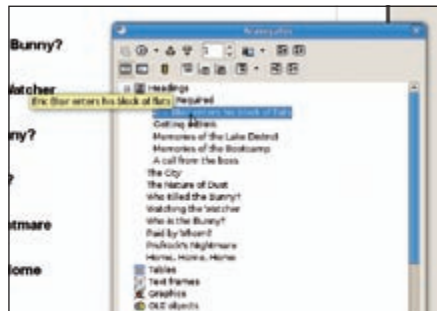
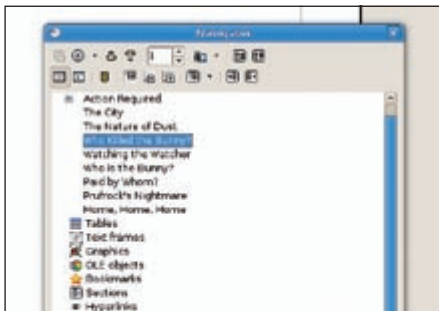
► Успешное создание структур в *OOo* зависит от стилей абзацев и Навигатора.

» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe/!

Применим схемы



Шаг за шагом: Использование Навигатора



1 Просмотр заголовков

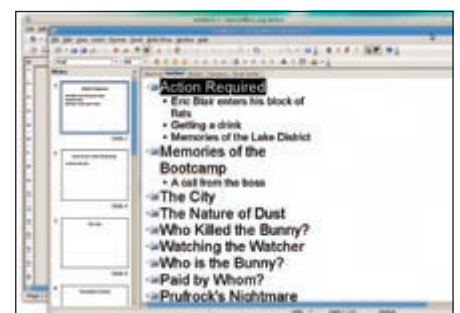
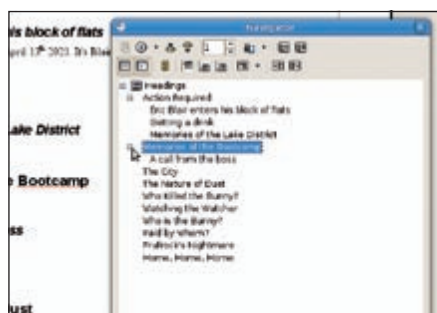
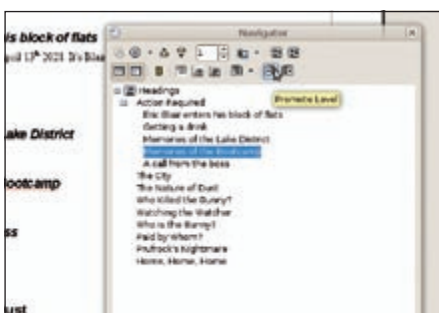
Проще всего переупорядочить части структуры документа с помощью Навигатора. Вызывается он выбором Правка > Навигатор. Пункт в самом верху называется Заголовки, и если вы добавляли их в документ, то увидите небольшой знак + рядом с ним. Щелкните на нем, чтобы увидеть список заголовков.

2 Уложим содержимое

Любой элемент, имеющий подчиненные (дочерние), также имеет рядом с именем иконку +. Щелкните на ней, чтобы увидеть все подпункты родительской записи. Второй щелчок закроет стопку. Даже если вы закроете родительский элемент, имеющий внуков (дочерние элементы дочерних), *Writer* будет помнить статус этих элементов.

3 Перетряхнем заголовки

В верхней строке панели инструментов Навигатора расположены две кнопки для перемещения глав вниз и вверх. Они передвигают выбранные элементы вверх или вниз по структуре. Учтите, если вы переместите главу ниже последнего дочернего элемента родителя, она перейдет в нижележащий родительский элемент, и наоборот.



4 Упорядочиваем

Можно также повышать элементы – скажем, с уровня 3 до уровня 2 – или понижать их. Выберите элемент, а затем используйте кнопки Повысить уровень или Понизить уровень у правого края второй строки панели инструментов. Отметим, что понижение заголовка уровня 1 делает его дочерним элементом заголовка первого уровня, расположенного выше.

5 Повышаем заголовки

Если вы повысили уровень заголовка, то все его бывшие соседи станут его дочерними элементами. Это логично, но может привести к казусам. Если вы планируете повысить заголовок, но хотите, чтобы все заголовки, расположенные ниже, остались при своих родителях, просто переместите его перед повышением к нижней границе списка.

6 Экспорт в Impress

Если вы используете *Writer* для создания начальной структуры презентации, выберите Файл > Отправить > Структуру в презентацию, чтобы открыть структуру в *Impress*. Заголовки уровня 1 станут заголовками слайдов, так что каждый заголовок получит свой слайд, а дочерние элементы станут маркированными элементами списка каждого слайда. **LXF**

» **Через месяц** Изучаем *Basket* дальше и строим личный облачный сервер.

GIMP: Упадок

Даже экспертам всегда есть чему поучиться. Майкл Дж. Хэммел делится впечатлениями от работы в *Gimp Paint Shop* с планшетом Wacom.



Наш эксперт

Майкл Дж. Хэммел
Участник разработки *GIMP* и автор трех книг по данной тематике, включая самую новую — «Эффекты в *GIMP*: руководство для художников».

На прошлом уроке мы изучали не слишком известную, но мощную функцию *GIMP* под названием Tool Presets: она дает пользователю возможность настроить *GIMP* на выполнение конкретных задач, например, на имитацию рисования пером или письма масляными красками. Применение *Gimp Paint Shop*, набора предустановленных настроек инструментов, открывает эту классную возможность для более широкого круга пользователей.

Начинающие обычно рассматривают *Gimp Paint Shop* как инструмент для создания графических рисунков с нуля. Продвинутые же пользователи знают, что это — только верхушка айсберга. Предварительные настройки дают возможность делать в цифровом виде то, чем декораторы голливудских фильмов занимаются уже много лет: превращать обыденность в сказку. При моей пытливости, я возжелал узнать, что еще можно делать с *Gimp Paint Shop*: работа мышью со спецнастройками показалась мне несколько пошлой, так что я прервал многолетнее воздержание от покупки планшета и раскошелился на Bamboo от Wacom (www.wacom.com/BambooTablet/bamboo.php). Это бюджетная модель, и многого я от нее не ждал, но в итоге своих экспериментов был очень и очень приятно удивлен.

Городская антиутопия

Мой эксперимент ставил две цели:

- 1 Узнать, что можно сделать с помощью планшета Bamboo.
- 2 Попробовать применить к нему руководство, написанное для планшета подороже.

Здесь я обратился к статье Making of Urban, автор которой — француз Пьер Фабр [Pierre Fabre] (www.cgarena.com/freestuff/tutorials/photoshop/urban/index.html). Смысл ее в том, чтобы взять урбанистический пейзаж и довести его до экстремально-упаднического вида, добавив стихии воды, огня и земли.

На уроке этого месяца я расскажу, как я научился пользоваться планшетом Wacom Bamboo в *Gimp Paint Shop* и получил собственную *GIMP*-версию дизайна Making of Urban, которую я назвал Десау («Распад»).

Для этой работы понадобилось пять стоковых изображений: городской пейзаж, две фотографии воды, одна фотография женщины и одна фотография вьющихся растений. Окончательная версия состоит из 73 слоев (51 из них — с масками слоев) и трех контуров. На всю работу ушло чуть больше 12 часов.

Основные усилия потребовались для того, чтобы вручную затемнить окна в зданиях. Тут я и понял, что предустановленная настройка Detail Brush для кисти и Digital Airbrush для аэрографа — это две полезнейшие настройки *Gimp Paint Shop* в проектах такого типа.

Wacom Bamboo, Linux и GIMP

Wacom Bamboo — маленький черный планшет с USB-разъемом, пером и подставкой. У него есть четыре клавиши в верхней части и сенсорное кольцо между ними. Ниже — рабочая область 5,8 × 3,7 дюйма. Хотя с виду планшет и мал, его размер оказался вполне достаточным для моих целей.



» Исходный городской пейзаж отыскан среди стоковых фото на iStockPhoto.com.

» **Месяц назад** Мы открыли миру тайные сокровища *Gimp Paint Shop*.

В городе



Wacom определяется из коробки сразу же после подключения в большинстве современных дистрибутивов; правда, не с полной функциональностью. Работают только рисовальная сторона пера и его клавиши. Чтобы заработала другая сторона пера (ластик), а также другие клавиши и сенсорное кольцо, нужно отредактировать файл **xorg.conf** (подробности этих настроек можно найти на **LXF DVD**). После редактирования файла нужно будет перезапустить X-сервер. Если вы не умеете делать это вручную, просто перезагрузите компьютер.

Настройка **xorg.conf** активирует функции Bamboo (или другого планшета от Wacom), но для их использования в **GIMP** их следует связать с нажатиями клавиш при помощи утилиты **wacomcpl**. В некоторых дистрибутивах она включена в пакет **linuxwacom** (его также можно найти на <http://linuxwacom.sourceforge.net>), хотя в Fedora 10 такого пакета нет. В моем случае, поскольку я являюсь пользователем Fedora, пакет **linuxwacom** пришлось скачать и собрать из исходных текстов.



» Он невелик, но оказалось, что больший размер и не нужен, даже для очень тонкой работы над деталями.



Шаг за шагом: Сотворим ад на земле!

1 Расчистим горизонт

Пора приступить к делу. Первая проблема, присутствующая на исходном фото — это нехватка фонового неба: здания чересчур высоки, а посреди изображения слишком много переходов и частично видимых построек.

Высота зданий легко настраивается с помощью выделения их верхних частей, копирования и вставки в новый слой. Затем этот новый слой перетаскивается вниз, чтобы урезать воспринимаемую высоту зданий. С данным фото процесс был прост, поскольку нижняя часть выбранных зданий почти совпадает с соответствующими верхними частями, и шов не бросается в глаза. Фрагменты неба были скопированы и помещены над зданиями и переходами в центре изображения. Благодаря этому в центральной части, освобожденной от загромождающих вид

зданий, засияло небо. Весь процесс укорачивания зданий и расширения неба потребовал пяти слоев. Также использование исходных кусочков неба для заплаток далее облегчит выделение неба для маски.

Полученный набор слоев далее был объединен, и получился новый городской пейзаж, без слоев. Отмечу, что позже я добавил еще несколько маленьких заплаток, чтобы убрать плоские белые стены с правой стороны изображения.

Нижняя часть выбранных зданий прекрасно уместилась на новом месте. Для получения окончательной версии с заплатками (см. справа внизу) понадобился еще один набор заплаток, включая самые высокие здания слева.



2 Цветокоррекция зданий

Следующий шаг — цветовая коррекция изображений. В целом исходное изображение было слишком ярким, с непомерной контрастностью. Для его коррекции я скопировал слой в новый пейзаж и уменьшил его насыщенность.

Затем я прибегнул к помощи диалогов Уровни, Кривые и Тон-Насыщенность для регулировки этого слоя. Далее слой был передвинут под исходный пейзаж, а его режим

выставлен на значение «цвет». Результаты показаны здесь. На печати они могут быть не столь явными, как на экране, но уменьшение контрастности выявило многие детали в изображении.

Обратите внимание на беспорядочные и неправильно расположенные лучи фонарей справа. Это упущенное мною последствие укорачивания зданий, которое я исправлю гораздо позже.

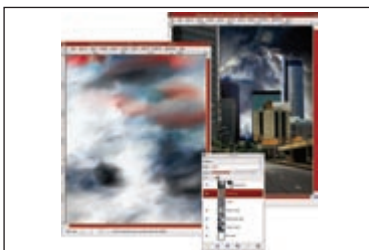


3 Зловещее небо

Почти однотонное небо было легко выделить с помощью инструмента Выделение связанной области. Я добавил белую маску слоя к слою городского пейзажа и залил выделение в маске черным цветом. Затем добавил несколько слоев стоковых изображений облаков, расположив их так, чтобы они находились под слоем с пейзажем. Сперва я пробовал окрасить фон неба при помощи предустановленных настроек, идущих с **Gimp Paint Shop**,

но результат вышел слишком унылым. Тогда я взял другие стоковые фото облаков, зеркально перевернул их (инструмент Зеркало, Flip Tool) и искажил (фильтр Whirl and Pinch). Цвет я добавил в самом конце работы в качестве финального штриха.

Первый вариант окрашенного неба (слева) делать было интересно, и это помогло мне прочувствовать планшет, но то небо выглядело недостаточно угрожающе для моей идеи.



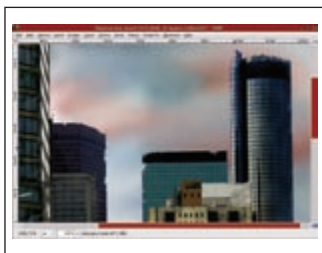
4 Обветшание зданий

Дальше пошла самая затратная по времени часть работы: добавление растительности стенам и разрушение окон и стен. Я начал с добавления к маске неба слоя с городским пейзажем, имитируя повреждение верхних частей отдаленных зданий. Для этого я применил инструмент Аэрограф с предустановками Digital Airbrush из *Gimp Paint Shop* и варьировал давление на перо планшета.

Повреждения верхушек строений на этом этапе были только доказательством правильности концепции. Фактически повреждения добавились позже, однако имеет смысл рассказать о них сейчас. Я грубо выделял каждое строение, копировал его и вставлял выделение как новый слой, затем удалял содержимое слоя. Так я получал слой с очерченными границами зданий, для которых я применял Кисть (предустановленная настройка Detail Brush) и Аэрограф (предустановка Digital Airbrush) для отрисовки затемнения

различных окон. Напомню, что с моей стороны это было чистым экспериментом, поэтому я начал с наиболее удаленных строений, делая мелкие затемнения, и все сработало достаточно хорошо для того, чтобы применить это и на более близких зданиях. В некоторых частях, как показано здесь, я полностью затемнил блок окон и затем нарисовал висящие провода, металлические фрагменты и сломанные рамы с помощью предустановки Detail Brush и белого цвета переднего плана. Затем я еще раз прошелся по этим линиям этой же кистью, но другими цветами. Режим Кисти был установлен на Overlay.

Эти грубые затемнения смотрятся лучше за плющом, который я добавлю в следующем разделе. Позже я вернулся к ним, вооруженный предустановленной настройкой Detail Brush для отрисовки тонких линий, изображающих трещины, сочетающиеся с полностью выбитыми окнами.



6 Починка фонарных столбов

Следующим шагом было залатать фонарные столбы. Чтобы все получилось как надо, я сделал подробное выделение изображения-оригинала с помощью Быстрой Маски. Это заняло очень много времени, и мне пришлось увеличивать масштаб изображения, чтобы обеспечить точность выделения. Нужно было поправить четыре фонарных столба и удалить остатки еще одного, оставшегося после укорачивания строений. Удаление было сделано путем выделения соответствующей области, небольшого перемещения выделения вверх на изображении-оригинале, и затем копированием и вставкой его как слоя-заплатки. Слой-заплатка затем перемещался вниз, перекрывая нехороший фонарный столб. Заплатка сохранила окраску оригинального изображения,

в то время как область, к которой она прилагалась, поменяла окраску после цветокоррекции зданий. Это было поправлено позже, когда я покрыл эту область плющом и чуток подраструшил строения.

После размещения этой заплатки была вставлена копия фонарного столба с исходного изображения и перемещена в одну линию с первым столбом справа. Маска слоя помогла соединить заплатку с изображением, хотя цвета все еще выбиваются из общей гаммы.

Слой заплатки фонарного столба был скопирован и изменен в размерах еще три раза, по одному для каждого оставшегося фонарного столба, который нуждался в заплатке. Каждый новый слой был перемещен в нужное положение и смешан с маской слоя.



5 Земные элементы

Затем начались эксперименты с растительностью, оплетающей здания. Я сделал выделение в стоковом изображении плюща и вставил его в изображение как новый слой, затем добавил маску слоя и использовал Аэрограф для маскировки случайных фрагментов. Этот слой я продублировал множество раз, изменяя маску слоя и помещая каждый слой все выше по высоте здания, тем самым получая все более обширную замаскированную область, уменьшающую видимое разрастание плюща.

Другой фрагмент изображения с выходящими растениями (вверху

посредине) я использовал для слоя бетона на тротуаре и для некоторых других зданий. Вставленное выделение я также уменьшил в размере, чтобы на удаленных строениях растения смотрелись как бы издалека. В некоторых случаях к вставленным выделениям применялась Нерезкая маска.

Я расположил растительность только на трех зданиях, а также на цементном покрытии тротуара по обеим сторонам изображения. Можно было добавить больше, но я решил, что на некоторых зданиях плющу расти неудобно.



7 Водяной поток

Эффект воды был создан в несколько приемов. Сначала я сделал выделение из стокового изображения с водой, вставил его копию в рабочее окно и расположил его как раз над слоем с городским пейзажем. Затем я размазал его по вертикали (фильтр motion-blur), что в основном убрало пену, не тронув цвет струй. Этот слой был очень большим, что позволило мне точно подогнать размазанную струю к размерам улицы. Маска слоя подчистила слой, и зрительно он плотно совпал со сторонами улицы. В заключение фильтр Ripple добавил воде некоторое искажение.

Вода выглядела стеклитко, как замерзшая. Имелись хорошие отражения (счастливый результат размазывания, не имеющий отношения к городскому пейзажу), но дорога под водой не была искажена.

Тот же процесс был повторен несколько раз для того, чтобы затем-

нить воду и добавить побольше волн и ряби, и тогда я понял, что скопировав саму дорогу и применив к ней фильтр Ripple, я частично избавлюсь от стекливости воды.

Чтобы изобразить водяной поток, нужно было добавить отблески на воде по сторонам дороги. Я сделал дополнительные копирования из стокового изображения воды, и затем вставил, изменил размер и расположил их в рабочем изображении. Это изменило вид воды, однако она выглядела еще недостаточно правильно.

Поэтому я добавил дополнительные штрихи белых брызг в разных местах на улице, которые я получил из дополнительно скопированных выделений из стокового фото. Маски помогли смешать их с водой, что полностью убрало ее стекливый вид. Несколько брызг по сторонам улицы и несколько белых полос в середине превратили замерзший пруд в бурлящую реку.



8 Настроение

До этого момента изображение было немного темноватым, но в общем не передавало никакого настроения – вспомните зеленые оттенки в «Матрице» или голубые в «Людах Икс». Сначала я осветлил небо и воду, не трогая зданий. Для этого я использовал радиальный градиент от белого к прозрачному, а также замаскировал здания. Поверх я добавил прозрачный слой и нарисовал белые блики на частях зданий. Также, еще один

радиальный градиент добавил желтое зарево, пронизывающее все изображение. Я очень люблю цвета осени. Вы найдете их во многих моих работах.

Блики на воде (слева сверху) стали более явными после добавления двух слоев с белыми бликами (посередине внизу). Но общее настроение задает желтое свечение (справа сверху), создающее ощущение, что солнце вот-вот сядет.



9 Население

В проекте Making of Urban присутствовала фигура мужчины, прогуливающегося с собакой. Но в стоковых архивах, которые я просмотрел, не нашлось подходящей картинке человека, идущего вдаль от камеры.

Я нашел изображение девушки, но фото было слишком ярким и не очень-то годилось. Я сделал аккуратное выделение, снова прибегнул к Быстрой Маске, скопировал, изменил масштаб и расположил девушку посередине улицы. Мне пришлось поэкспериментировать с диалогами Тон-Насыщенность, Яркость-Контраст, Уровни и Кривые и затемнить фигуру девушки так, чтобы не полностью

исчезли подробности, например, цвет ног.

Я скопировал этот слой и размыл его, поместив копию над исходным слоем. Снова скопировал слой, но на этот раз удалил содержимое слоя, заменив его нарисованными от руки бликами, и расположил их в основном вдоль левого бока девушки.

Последний кусочек человеческой мозаики мы возьмем из двух слоев с водой. Первый – немного окрашенной воды, которая придает девушке очень мягкую тень. Этот слой я поместил под слоем с девушкой. Над слоем с тенью я поместил копию всплеска от камня, брошенного в водоем.



Заключение

По окончании эксперимента я уже очень хорошо понимал, как нужно использовать мой новый планшет Wacom Bamboo (а также истинно полюбил ощущения от его использования), плюс я гораздо лучше разобрался в применении многочисленных готовых настроек, доступных в *Gimp Paint Shop*. Хотя я задействовал только часть из них, я гораздо лучше оценил их общий потенциал.

Окончательное изображение было хорошо принято теми, кому я его показал, но я все еще вижу простор для улучшений. Например, я теперь понимаю, что заплатки надо было вырезать из самого лучшего источника. В основном я вырезал их из исходного изображения, хотя вместо этого надо было брать их из копии зданий

со скорректированным цветом. К счастью, эти проблемы с цветом укрылись под выющимися растениями и общим цветовым настроением изображения.

Другие проблемы касаются только восприятия. Композиция изображения и рядом не стоит с композицией моего источника вдохновения. Картинке не хватает глубины, которую в *Making of Urban* дает дорога, поворачивающая слева направо. Ну и, конечно, выбитые взрывом окна не столь реалистичны.

Тем не менее я вполне доволен результатом. Моя жена считает, что мне надо торговать печатными изображениями вроде этого, а я думаю – что только ни даст мне шанс поэкспериментировать с *Bamboo* и *GIMP*, будет очень кстати. **LGX**

» **Через месяц** Пост-апокалиптическое возрождение, наподобие феникса из пепла!

GData: Курс

Давно перестали удивляться, увидев Карты Google на сайтах, не имеющих отношения к Google? И правильно: никаких секретов здесь нет. Евгений Крестников научит вас легко проделывать такие трюки.



пример, в нашем поиске или Картах), мы предлагаем разработчикам использовать API AJAX».

На следующих четырех страницах мы дадим введение в API данных Google, которое поможет вам понять основные принципы работы с этим замечательным инструментарием. Мы не будем подробно описывать протоколы синдикации или способы установки клиентских библиотек, предполагая, что вы – достаточно опытный web-разработчик, знакомый с Linux и желающий освоить новую технологию. А разобраться в деталях вам поможет документация, доступная по адресу <http://code.google.com/intl/ru/apis/gdata/>.

С чего начать?

Создатели Google Data API рекомендуют начинать его освоение с основных принципов и понятий. Затем вам предлагается остановиться на одной из доступных на сайте клиентских библиотек (ее выбор зависит от используемого вами языка программирования) или работать непосредственно с XML/HTTP. Последним этапом станет выбор службы Google, с которой вы собираетесь взаимодействовать, и чтение соответствующего раздела руководства. Это выглядит вполне разумным, так что на данном уроке мы будем следовать методике, рекомендованной компанией.

Согласно определению Википедии, web-синдикация – временное распространение информации на различные страницы или web-сайты. Основной принцип синдикации заключается в передаче заголовков материалов и ссылок на них (можно, например, получить информацию о новых сообщениях на форуме). Первоначально технология использовалась на новостных ресурсах и в блогах, но постепенно сфера ее применения расширилась. Сегодня существуют два формата оповещения об изменении существующих данных, появлении новых и ретрансляции информации в Сети: созданный компанией Netscape RSS и конкурирующий стандарт Atom. Оба они основаны на XML. Google Data API позволяет разработчику использовать RSS 2.0, Atom 1.0 и Atom Publishing Protocol (AtomPub или APP). Формат Atom

имеет ряд преимуществ перед RSS – он легко расширяется и обладает большими возможностями. С помощью APP можно отправить HTTP-запрос GET к конкретному ресурсу (например, каналу данных

или отдельному сообщению). Ответ от сервера придет в формате синдикации Atom. Кроме того, APP позволяет создавать, редактировать и удалять ресурсы, используя запросы типа POST, PUT или DELETE. К сожалению, ни в одном из стандартов нет возможности отправить серверу запрос на выборку данных по определенному критерию и получить обратно ответ со списком соответствующих результатов. Однако у Atom есть стандартный механизм расширений, при помощи которого GData легко справляется с подобными задачами.



Наш эксперт

Евгений Крестников
Ежемесячно перерывает пол-Ин-тернета в поисках информации для Новостей LXF. API данных Google делает его жизнь не легче, но интереснее.

В LXF117 мы уже писали про онлайн-сервисы Google. Их возможности сегодня не ограничиваются поиском информации в Сети и доступом к электронной почте: пользователь может хранить на серверах Google свои документы и фотоальбомы, работать с картами или вести свой блог. Популярный видеосервис YouTube – это тоже Google. Но нас сегодня будут интересовать не сами сервисы, а возможность вызывать и использовать их в своих собственных приложениях. Это реализуется посредством Google Data API (API данных Google, сокращенно – GData). Прочитав Майкла «Chewy» Тревеллу [Michael Trehwella], консультанта Google по вопросам разработок (LXF117): «Google Data API представляет собой набор интерфейсов чтения и записи данных для широкого спектра онлайн-сервисов Google, использующих единый протокол. С его помощью Google позволяет

сторонним разработчикам расширять функциональные возможности своих продуктов. Вместо того, чтобы создать API, построенный на разрозненном наборе протоколов, мы решили использовать стандарт AtomPub, основанный на технологиях Atom/XML и HTTP. Это значительно упрощает процесс обучения персонала для сторонних разработчиков. С тех пор, как спецификации AtomPub были утверждены в RFC 5023, многие компании стали создавать API, основанные на этом стандарте. В тех случаях, когда использование подобных интерфейсов не имеет смысла (на-

«В Google решили использовать стандарт AtomPub.»

МОЛОДОГО БОЙЦА

Каналы GData позволяют работать с блогами, новостными лентами, электронной почтой и другими сервисами Google. Отправлять запросы и получать их результаты можно через собственные приложения, с помощью стандартных программ для чтения RSS или расширений браузера. Протокол GData нейтрален к языкам программирования – вы можете написать клиент на любом языке, который позволяет отправлять HTTP-запросы и принимать ответы на основе XML. Чтобы получить данные от службы, поддерживающей GData, вам нужно отправить запрос HTTP GET. При этом служба выдаст результат в виде канала Atom или RSS. Она также может поддерживать обновления данных при помощи запроса HTTP PUT. Разумеется, сервис может предоставлять каналы только для чтения (например, канал результатов поиска) или для чтения и записи (скажем, Google Calendar). Есть только одно ограничение: GData не позволяет создавать и удалять каналы – это может делать только служба.

Библиотеки и программирование

Давайте сразу же договоримся, что мы не будем рассматривать вариант работы с XML/HTTP напрямую: он обладает наибольшей гибкостью (и трудоемкостью), но на Google Code есть ссылки на специализированные библиотеки практически для всех языков программирования: Java, .NET, PHP, Python, Objective-C, JavaScript; наиболее заметным исключением из этого списка является Perl. Они значительно упрощают процесс разработки, предоставляя высокоуровневый интерфейс, позволяющий абстрагироваться от деталей. Подробное описание всех форматов и протоколов на английском языке доступно на сайте <http://code.google.com/intl/ru/apis/gdata/docs/2.0/reference.html>.

Помимо официально представленных на сайте Google, в Сети можно найти другие клиентские библиотеки GData. Разумеется, описать все средства разработчика мы не в состоянии (для этого пришлось бы посвятить им весь номер), поэтому сосредоточимся на клиентской библиотеке PHP, поскольку многие динамические сайты в Сети разработаны с использованием именно этого языка программирования.

Клиентская библиотека для языка PHP предоставляется компанией Zend в рамках проекта Zend Framework, однако она может быть установлена отдельно. Рассмотрим последний вариант. Нам потребуется компьютер с Linux (другие операционные системы поддерживаются тоже, но нас они интересовать не будут) и PHP5 версии не ниже чем 5.2.4. Приводить подробную инструкцию по установке PHP также не станем: об этом написано в соответствующем разделе документации вашего дистрибутива (например, в Ubuntu достаточно набрать команду `sudo apt-get install php5`). Вместо этого сразу перейдем к процессу установки библиотеки.

Для начала необходимо скачать ее последнюю версию со страницы <http://framework.zend.com/> (на момент написания статьи была доступна Zend Gdata 1.8.3, но к моменту, когда вы будете читать эту статью, ситуация, естественно, может измениться) и распаковать архивный файл (в нашем случае – **ZendGdata-1.8.3.tar**

gz) в один из каталогов жесткого диска. После этого вы увидите несколько подкаталогов:

- » **demos** – примеры приложений;
- » **documentation** – документация по библиотеке;
- » **library** – исходные тексты;
- » **tests** – файлы для автоматического тестирования.

Кроме того, в корне архива есть несколько текстовых файлов – в частности, инструкция по установке. Установка библиотеки не должна вызывать затруднений у опытных пользователей Linux (вам может потребоваться активировать некоторые расширения PHP, в первую очередь, *openssl* и *iconv*, если вы этого еще не сделали). Для начала работы достаточно добавить директорию библиотеки (подкаталог **library** из архива) в переменную PHP `include_path`. Это можно сделать через файл **php.ini**, конфигурационный файл Apache **.htaccess** или динамически, при помощи PHP-функции `set_include_path()` или ключа **-d** в командной строке.

Теперь нам необходимо проверить работоспособность установленного ПО. Для этого можно воспользоваться скриптом **demos/Zend/Gdata/InstallationChecker.php**. Как и прочие примеры из каталога **demos/Zend/Gdata**, он работает и как CGI-сценарий, и из командной строки. Если установка прошла успешно, вы увидите на экране результат, представленный на рисунке 1 (или аналогичный, но менее красочный вывод в консоли).

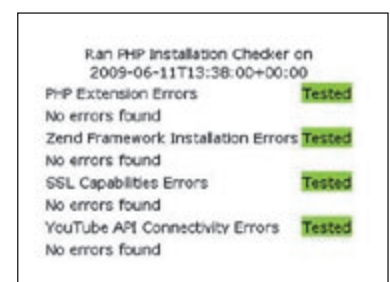
Выбираем сервис

Мы уже достаточно поговорили о доступных стороннему разработчику API данных Google в целом – теперь остановимся на них подробнее. API данных служб Google может быть использован для управления доменами, а также для интеграции решений Google и существующей ИТ-инфраструктуры. Google Base – это масштабируемая база данных, в которой пользователь может хранить общедоступную структурированную информацию. В сущности, сервис можно рассматривать как еще один способ закладывать данные в базы данных Google. Здесь можно разместить весь контент, который Google будет хранить и индексировать для специализированного интернет-поиска. В зависимости от релевантности, ваш контент может также быть включен и в главный поисковый индекс, и в другие продукты Google. API данных Google Base позволяет разработчикам запрашивать информацию, а также вводить данные и управлять ими из своих программ.

Blogger – сервис Google для создания блогов. С помощью API данных Blogger клиентские приложения могут добавлять, удалять или изменять записи, а также производить поиск по определенным критериям. Данные API позволяют добавить на сайт динамический список записей и комментариев, создать модуль расширения для публикации записей или агрегатор блога.

API данных Календаря Google позволяет создавать клиентские приложения для просмотра и обновления событий кален-

» Установка Zend Framework прошла успешно: ошибок не найдено.



» **Пропустили номер?** Узнайте на с. 103, как получить его прямо сейчас.

даря в форме каналов API данных Google. С их помощью можно, например, разработать web-интерфейс для просмотра информации Календаря Google через внешний сайт, включить напоминание о предстоящих событиях, настроить синхронизацию Календаря Google с мобильным устройством и так далее. Логически прилегающие к нему API данных Контактов Google дают клиентским приложениям возможность просматривать и обновлять контактные данные. Если вы хотите, скажем, настроить синхронизацию списка контактов мобильного устройства с Kontakтами Google, можете воспользоваться им.

API данных Блокнота Google, как нетрудно догадаться по его названию, обеспечивает приложения возможность доступа к содержимому Блокнота Google. С его помощью программа-клиент может запросить список общедоступных блокнотов пользователя или содержание существующего блокнота.

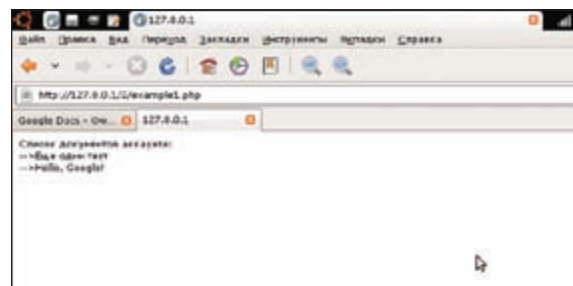
API данных Таблиц Google предназначены для просмотра и обновления содержания электронных таблиц в виде каналов GData. Клиентская программа может получить список таблиц пользователя, а также представлять, редактировать или удалять содержание книги в существующей таблице. API данных Документов Google позволяют загружать документы и сохранять их как каналы API данных Google. Кроме того, можно запросить список документов пользователя, а также изучать содержание существующего документа.

API данных web-альбомов Picasa дает пользователю возможность работать с альбомами Picasa из клиентской программы: получать список альбомов, комментариев и тегов пользователя, а также просматривать, редактировать и удалять содержимое существующей записи. С их помощью легко, например, добавить свои общедоступные фотографии на собственный сайт и дать посетителям возможность комментировать их, сохраняя комментарии в альбомах Picasa.

API данных YouTube предоставляет доступ практически ко всему функционалу сервиса. С его помощью вы можете добавлять видео, изменять избранное, метаданные комментарии и настройки, а также осуществлять поиск роликов по заданным критериям и многое другое.

Google Docs из PHP5

Работу с онлайн-сервисами Google при помощи Zend Gdata 1.8.3 мы рассмотрим на примере Google Docs. Искушенный web-разработчик, пожалуй, назовет наш пример чересчур простым – и, возможно, он будет прав. Однако излишняя функциональность (а также проверка возникающих ошибок, совершенно необходимая при разработке реального приложения) будет только ме-



► GData API позволяет легко прочесть названия имеющихся у нас документов.

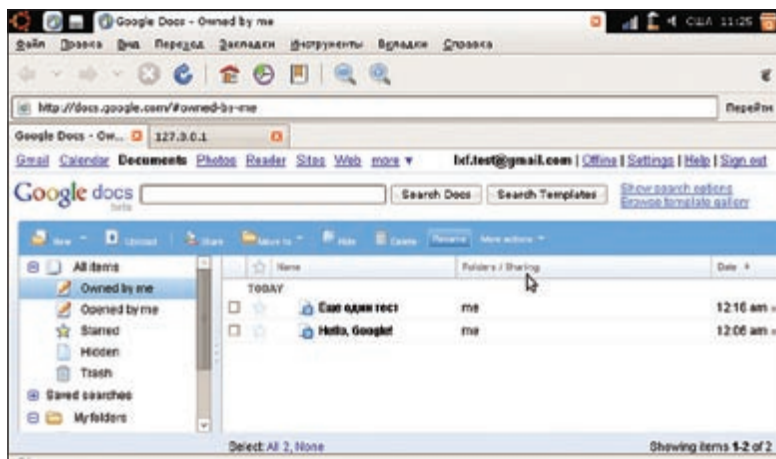
шать основной цели – получению представления о Google Data API, а наш сценарий, несмотря на кажущуюся простоту, включает все ключевые элементы «настоящей» программы: открытие сессии, чтение данных, предоставляемых службой Google, и их изменение. Говоря более конкретно, наша программа будет получать список документов, имеющихся у некоторого пользователя в службе Google Docs, печатать их имена в стандартный вывод, а затем выгружать на сервер новый файл.

Большинство служб Google (и Документы – не исключение) требуют авторизации. Для этих целей в GData предусмотрено три метода: ClientLogin, OAuth и AuthSub. Первый из них наиболее прост в реализации и может быть рекомендован в тех случаях, когда ваша программа может напрямую запросить имя учетной записи и пароль у пользователя. Если это по каким-либо причинам нежелательно (что часто имеет место в случае web-приложений), можно прибегнуть к услугам посредника, который выполнит за вас идентификацию посетителя и сообщит, тот ли он, за кого себя выдает.

```
01: <?php
02: require_once 'Zend/Loader.php';
03: Zend_Loader::loadClass('Zend_Gdata_Docs');
04: Zend_Loader::loadClass('Zend_Gdata_Docs_Query');
05: Zend_Loader::loadClass('Zend_Gdata_ClientLogin');
06: $login = 'user.name@gmail.com';
07: $password = 'password';
08: $srv = Zend_Gdata_Docs::AUTH_SERVICE_NAME;
09: try {
10: $client = Zend_Gdata_ClientLogin::getHttpClient($login,
    $password, $srv);
11: }
12: catch (Zend_Gdata_App_Exception $e) {
13: echo "Ошибка: $e->getMessage()\n";
14: exit(1);
15: }
```

В строках 2–5 мы подключаем библиотеку и загружаем из нее необходимые нам классы: **Zend_Gdata_Docs** (работа с документами Google), **Zend_Gdata_Docs_Query** (поиск по запросу) и **Zend_Gdata_ClientLogin** (авторизация пользователя по первому методу). Реквизиты задаются в строках 6 и 7; переменная **\$login** должна содержать полный e-mail адрес пользователя. Переменная **\$srv**, определяемая в строке 8, задает имя службы Google, в которой вы хотите авторизоваться. Здесь оно представлено символьной константой **Zend_Gdata_Docs::AUTH_SERVICE_NAME**, которая (на момент написания статьи) просто равна строке 'writely' (если вы забыли, откуда ведет свою родословную Google Docs, спросите у Википедии: ru.wikipedia.org/wiki/Google_Docs, и этот набор букв обретет смысл). Статический метод **getHttpClient()**, вызываемый в строке 10, возвращает объект **Zend_Gdata_HttpClient**, который используется для доступа к данным Google Docs. Он может выбрасывать ряд исключений, в том числе **Zend_**

► Наш тестовый пользователь хранит в Google Docs два файла.



» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe/!

Gdata_App_AuthException (ошибка авторизации) или **Zend_Gdata_App_CaptchaRequiredException** (для завершения процесса авторизации пользователь должен ввести текст, указанный на картинке, дабы подтвердить свою человеческую природу). Здесь мы не делаем разницы между этими (и другими) исключительными ситуациями, а просто пишем не слишком информативное сообщение об ошибке и выходим из программы.

Далее, давайте попробуем получить список документов учетной записи. Для этого нам потребуется создать главный объект программы: **Zend_Gdata_Docs** (обратите внимание, что его конструктор принимает в качестве параметра ссылку на объект HTTP-клиента, полученную ранее). Для тестирования программы мы создали специального пользователя – список его документов и вывод скрипта можно видеть на рисунках 2 и 3:

```
$mydocuments = new Zend_Gdata_Docs($client);
$myfeed = $mydocuments->getDocumentListFeed();
echo "Список документов аккаунта:\n";
foreach ($myfeed->entries as $entry)
{ echo "--> $entry->title\n";
}
```

Здесь все просто: объект со списком документов возвращается методом **getDocumentListFeed()**. Его элементы (то есть отдельные документы) доступны как **\$myfeed->entries**; **\$entry->title** возвращает название документа.

Последнее, что мы сделаем – загрузим новый документ на сервер; результат выполнения этой операции можно видеть на рисунках 4 и 5.

```
$localname="/tmp/test.txt";
$googledoc="test.txt";
$newdoc = $mydocuments->uploadFile($localname, $googledoc,
null, Zend_Gdata_Docs::DOCUMENTS_LIST_FEED_URI);
echo "Загружен документ: $newdoc->title\n";
?>
```

Мы берем локальный файл **/tmp/test.txt** (предполагается, что он существует) и выгружаем его на сервер под именем **test.txt**. Два оставшихся аргумента метода **uploadFile()** задают mime-тип выгружаемого файла и URI службы, которая принимает его (константа **Zend_Gdata_Docs::DOCUMENTS_LIST_FEED_URI** или, что то же самое, <http://docs.google.com/feeds/documents/private/full>).

В такой форме наша программа имеет одну неприятную особенность: после выполнения ее N раз подряд вы обнаружите в своих Документах Google N штук файлов **test.txt**. Если это не то, чего вы желаете, перед выгрузкой документа на сервер следует проверить наличие старой версии и, при необходимости, удалить ее. Это делает следующий код (его нужно разместить сразу после объявления переменных **\$localname** и **\$googledoc**):

```
$query = new Zend_Gdata_Docs_Query();
$query->setTitle($googledoc);
$query->setTitleExact("true");
try {
    $doc = $mydocuments->getDocumentListEntry($query);
    $docExists = true;
} catch (Exception $e) {
    $docExists = false;
}
if ($docExists) {
    echo "Удаляю предыдущую версию документа $doc->title\n";
    $doc->delete();
}
```

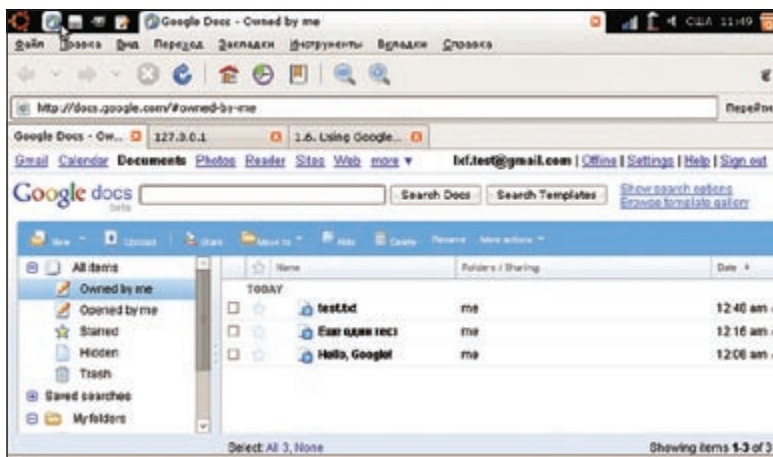
Объект **Zend_Gdata_Docs_Query** представляет собой запрос к службе Google Docs. Запросы – это просто специально сформированные URL (наш, например, будет иметь вид <http://docs.google.com/feeds/documents/private/full?title=test.txt&title-exact=true>), но для удобства работы с ними мы вновь применяем высокоуровневый API. Метод **setTitle()** задает название документа, который мы ищем, а **setTitleExact()** указывает, что нас интересует только точное соответствие, а не совпадение с подстрокой.

Удивительно, но в случае отсутствия требуемого документа метод **getDocumentListEntry()**, выполняющий поиск, не возвращает **null** или пустой объект **Zend_Gdata_Docs_DocumentListEntry**, а выбрасывает исключение. Это не очень удобно, поэтому мы используем специальную логическую переменную **\$docExists**, чтобы удалить старую версию файла с сервера, если она уже существует.

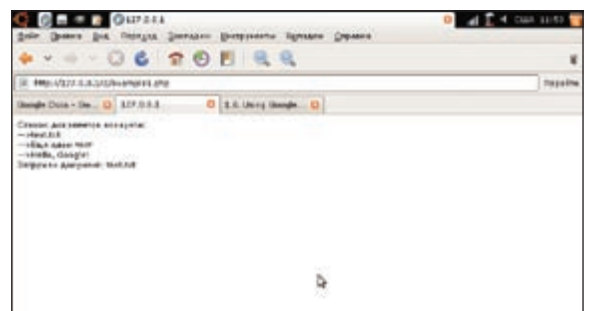
Что дальше?

Вот так, вкратце, выглядят Google Data API. Если вас заинтересовали технологии Google, советуем вам изучить сайт <http://code.google.com/> – там находится, пожалуй, самое полное руководство по GData. Его посещение будет полезно не только начинающим «гуглокодерам», но и знатокам, поскольку продукты и сервисы поискового гиганта развиваются очень быстрыми темпами и, соответственно, регулярно обновляется и API. Так, пока мы писали эту статью, компания анонсировала сразу два интересных проекта: **Webelements** и **Wave**. Первый имеет непосредственное отношение к GData – пройдя по ссылке <http://www.google.com/webelements/>, пользователь может выбрать интересующий его сервис (поддерживаются: Календарь, Карты, Google Talk, поиск в Web, лента новостей, новости YouTube, а также презентации и таблицы Google Docs) и получить готовый код для своего сайта, позволяющий встроить сервис Google в виде плавающего фрейма. В рамках второго планируется разработать альтернативный транспорт для обмена сообщениями, который (по замыслу компании) заменит электронную почту и сервисы IM (ICQ и иже с ними).

Так что держите руку на пульсе, а мы постараемся в этом вам помочь, регулярно публикуя обзоры новых возможностей открытого ПО. **LXF**



➤ Результат выгрузки файлов с точки зрения Google Docs...



➤ ...и нашей программы.

Web-приложения:

Часть 7: Нейл Ботвик узнает, что с настроенным и запущенным web-сервером *Apache* можно добраться до приложений из любой точки земного шара.



» Установка *Gallery* на первый взгляд довольно долгая, но выполнить ее несложно, потому что она состоит из нескольких небольших шагов.

Кроме PHP, имеется еще один компонент, используемый многими web-приложениями – сервер баз данных *MySQL*. Вместе эти элементы образуют классическую платформу LAMP (Linux, *Apache*, *MySQL*, PHP), ставшую стандартом для предоставления динамического содержимого. Вероятно, пакетный менеджер вашего дистрибутива добавил PHP заодно с *Apache*, но на всякий случай убедитесь в этом, прежде чем двинуться дальше. Ну, а *MySQL* – это пакет типа «поставил и забыл»: данные в нем хранят различные приложения, но базовая настройка выполняется при инсталляции, а затем каждый отдельный пакет выполняет собственные настройки с помощью установочных скриптов.

Подготовив платформу LAMP, установим два приложения. Первое – *SquirrelMail*, популярный почтовый web-клиент, а второе – *Gallery*, онлайн-фотогалерея; возможностей у нее столько, что и личной жизни не хватит все их перепробовать. Это типичные web-приложения, поэтому методы, используемые при работе с ними, справедливы и для многих других web-приложений.

Онлайн-фотогалерея

Фотогалерей в Интернете множество, ну и зачем создавать свою?

- » Чтобы хранить все в одном домене.
- » Чтобы хранить большие изображения, без ограничений на размер файлов.
- » Чтобы показывать фотографии без навязанной рекламы – или уж с рекламой, за которую платят вам.
- » А потому, что вам не слабо это сделать.



Наш эксперт

Нейл Ботвик

У Нейла Ботвика по компьютеру в каждой комнате. Но по соображениям безопасности он никогда не скажет вам, где центральный сервер.

Мы начали эту серию уроков (давным-давно, в LXFI13/114, если вы его пропустили) с установки и настройки мощного web-сервера *Apache*. Но в текущем состоянии эта служба может только раздавать статические web-страницы или, максимум, выполнять какие-нибудь доморощенные скрипты. Однако возможности *Apache*, особенно вместе с PHP, гораздо шире.

Видите ли, мы еще не приспособили *Apache* для запуска скриптов, встраиваемых в web-страницы. Скрипты можно писать на разных языках, но PHP – один из самых популярных. Это значит, что найдутся готовые сценарии для чего угодно, от простеньких встроенных функций до полноценных web-приложений. Если вы не знакомы с данной концепцией, то по своим возможностям они похожи на настольные приложения, только доступ к ним осуществляется через браузер. А зачем нам браузер вместо обычной программы?

- » Доступ к данным можно получить из различных мест.
- » С одной программой могут одновременно работать несколько пользователей.
- » Программы выполняются в браузере и не зависят от операционной системы клиента.
- » Для их работы не нужно устанавливать на компьютер никакое ПО, поэтому их можно запускать на общедоступной машине.

» **Месяц назад** Мы влезали в свой компьютер отовсюду благодаря SSH и VNC.

Рулим контентом

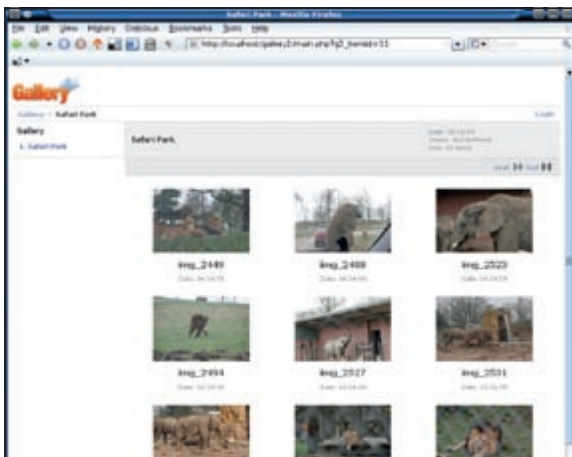
Как мы уже бегло упомянули, *Gallery* – это онлайн-фото-галерея с открытым исходным кодом, имеющая модульную архитектуру, что упрощает добавление нового функционала. *Gallery* написана на PHP, и можно установить ее прямо на web-сервер; а если сделать это с помощью менеджера пакетов, вы получите дополнительное преимущество в виде автоматических обновлений. Если так поступить нельзя (например, в качестве сервера используется компьютер в сети, а не ваш собственный, или нужна более свежая версия, чем та, что в дистрибутиве), то возьмите последний релиз с сайта <http://gallery.menalto.com>. На момент написания статьи это 2.3, хотя для любителей приключений есть и бета-релиз версии 3.0. Состав доступных пакетов различается. Как правило, достаточно версии Typical (обычная) – пакет содержит ядро программы и наиболее популярные темы и модули. Full (полная версия) включает все темы и модули, и это, наверное, перебор для новичков, если только вы не маньяк-настройщик. В версии Minimal есть только базовые компоненты, и она предназначена для тех, кто хочет установить остальное выборочно. Впрочем, с этим лучше не связываться, пока вы не наберетесь опыта и не будете точно знать, чего хотите. То же касается и пакета Developer.

Ваш выбор гвоздями не прибит, и позже можно добавить и тем, и модулей. Также можно выбрать между сугубо английской и многоязычной версией. Английским, слава богу, у нас не мудрено блеснуть, но мы все-таки российский LXF – так что возьмем многоязычный пакет версии Typical: **gallery-2.3-typical.tar.gz**.

Настройка MySQL

Информацию о картинках *Gallery* хранит в базе данных, и *MySQL* – лучший выбор. Если вы устанавливали *Gallery* через менеджер пакетов, он должен был уже настроить базу данных. Если нет, выполните команды

```
mysqladmin -uroot -p create gallery2
```



➤ Установив *Gallery*, вы с необыкновенной легкостью поделитесь воспоминаниями с друзьями.

PHP

Большинство web-приложений написано на PHP, и *Apache* обычно справляется с PHP-страницами и без дополнительных настроек. Однако в некоторых дистрибутивах PHP разбит на отдельные пакеты – например, в Debian нужно установить **libapache-mod-php5**.

Для определения подходящих пакетов и их статуса наберите «apache php» в строке поиска менеджера пакетов. Вы должны понимать, что если PHP в *Apache* сконфигурирован неправильно, то могут возникнуть проблемы, хотя установка из пакетного менеджера должна позаботиться о соответствующих настройках. Если при попытке открыть PHP-страницу *Apache* показывает ее код, измените параметры. Один из файлов

в каталоге **/etc/apache** должен быть похож на этот:

```
<IfModule mod_php5.c>
AddType application/x-httpd-php .php
.phphtml .php3
AddType
application/x-httpd-php-source .phps
</IfModule>
```

Может также потребоваться изменить директиву **DirectoryIndex** в настройках *Apache*. Она перечисляет файлы, которые попытается отобразить сервер, получив запрос на обработку каталога. Каждый из них пробуется по порядку, пока не будет найден подходящий – обычно это нечто вроде

```
DirectoryIndex index.php index.html
```

возможно, и с другими вариантами.

```
mysql gallery2 -uroot -p -e "GRANT ALL ON gallery2.*
To username@localhost IDENTIFIED BY 'password'"
```

Обе команды запросят у вас пароль, который был задан для суперпользователя root при установке *MySQL*; он не обязан совпадать с паролем root для входа в систему. Если база данных была установлена без пароля (не очень хорошая идея), уберите ключ **-p**. Альтернатива – задать пароль root командой

```
mysql -u root -e "set password = PASSWORD('mynew password')"
```

Во второй команде из нашего исходного набора создаваемому пользователю *MySQL*, который будет применяться *Gallery* для соединения с базой данных, назначаются имя и пароль. Пользователи *MySQL* не имеют ничего общего с пользователями на вашем компьютере. Если у вас удаленный сервер, то имя пользователя и пароль вам сообщат. Либо вы можете создать базу данных сами и установить права доступа с помощью *phpMyAdmin*, но такое уже за рамками нашего урока.

Запросы к каталогу

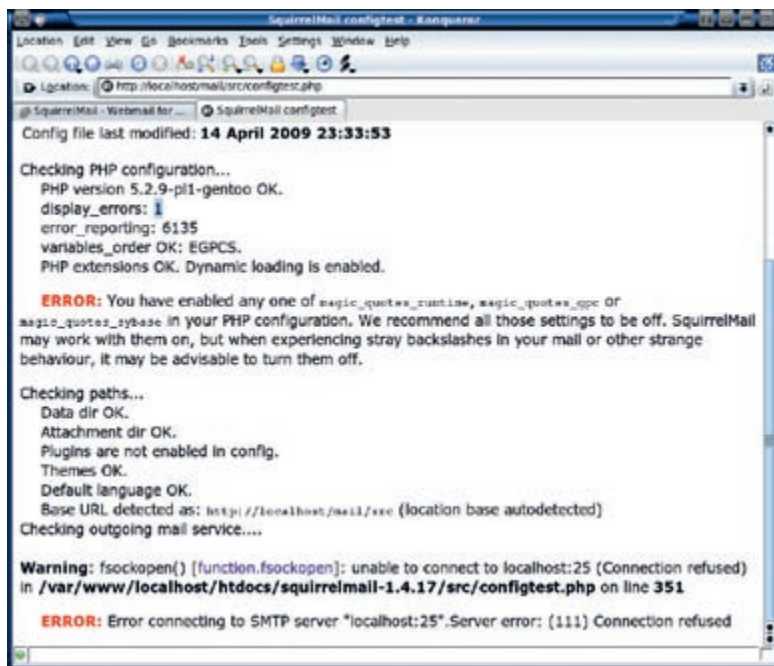
Теперь создадим каталог, где будут храниться фотографии; он не должен находиться в корневом каталоге web-сервера. В первой статье этой серии мы решили использовать в качестве корневого каталога (DocumentRoot) *Apache* директорию **/var/www/[имя_хоста]/htdocs**. Если вы последовали этому примеру, создайте рядом каталог для данных:

```
mkdir /var/www/localhost/gallery/g2data
chmod 777 /var/www/localhost/gallery/g2data
```

Таким образом, к фотографиям нельзя будет обратиться напрямую по URL: это нарушило бы требования безопасности *Gallery*.

Распакуйте скачанный архив и скопируйте получившийся каталог **gallery2** в каталог **htdocs** web-сервера. Покончив с подготовительными шагами по настройке базы данных и каталогов, запустите установку *Gallery*, открыв в браузере ад-

»



➤ Пусть *SquirrelMail* проверит текущую конфигурацию после запуска скрипта настройки.

рес <http://ваш.хост/gallery2/install>. Вы пройдете через несколько страниц с настройками, на каждой из которых будет собственная справка и возможность прервать установку. Главное решение, которое нужно принять – это выбор между стандартной и много-сайтовой установкой. В последнем случае на один сервер можно установить несколько экземпляров *Gallery*, что удобно, если нужно получить несколько независимых галерей. Это также отлично подходит для создания многопользовательской системы, поскольку в одной *Gallery* может быть несколько отдельных альбомов с собственными правами доступа. Сейчас мы воспользуемся стандартной установкой.

Дойдя до страницы с настройками базы данных, введите имя пользователя и пароль, заданные вами ранее. Не поддавайтесь искушению отделаться пользователем `root` и его паролем, даже если установщик предлагает их по умолчанию, потому что у этого пользователя есть права на изменение всех таблиц. Создание отдельного пользователя – не только хорошая привычка, но и мера, позволяющая ограничить возможный ущерб в будущем, если в *Gallery* будут найдены уязвимости.

Пользователь-администратор, имя и пароль которого у вас также запросят, отличается от пользователя базы данных, имя и пароль которого мы только что ввели. В *Gallery* есть учетные записи (с паролями), определяющие, кому что разрешено делать в каждом альбоме. Администратор может делать все, и, конечно, будет лучше использовать для него менее очевидное имя, чем «`admin`» (по умолчанию). Завершив эту скучную часть, выбирайте модули для установки. Если места на диске хватает, можно добавить их все. (А если места мало, то куда вы денете фотографии?) Модули и подключаются, и отключаются в административной панели *Gallery*, поэтому в излишках нет ничего страшного: наоборот, с ними будет немного проще экспериментировать. Можно активировать все установленные модули, но если вы устанавливали все подряд, то лучше активировать их по мере необходимости. Если *Gallery* будет доступна через Интернет, почитайте руководство по безопасности и только потом завершите установку.

Добавим фотографии

Пора загружать фотографии. Можно сделать это с ходу, но, наверное, лучше сначала раскидать снимки по альбомам. Для этого кликните по ссылке `Add Album` [Добавить альбом] и задайте путь

и имя (остальные настройки можно не трогать, если вам не терпится), затем кликните по ссылке `Add Item` [Добавить фотографию]. Здесь можно столкнуться с парой ограничений: может появиться предупреждение о невозможности создания миниатюр, или что максимальный размер загружаемого файла – 2 МБ. Чтобы исправить первое, кликните по ссылке `Site Admin` [Администрирование сайта] и затем выберите `Plugins` [Модули]. Для создания миниатюр из фотографий *Gallery* использует стороннюю утилиту, и лучший выбор – *ImageMagick*. Кликните на ссылке `Install` [Установить], чтобы она заработала (это не совсем точное название: пакет-то уже установлен, просто не активирован).

Максимальный размер загружаемого файла – это настройка PHP, и, таким образом, исправить ситуацию можно по-разному. Если вы передаете файл через Интернет на удаленный сервер, то, наверное, лучше изменить размер изображения до его загрузки. Если сервер локальный, просто увеличьте максимальный размер файла: найдите файл `php.ini` где-то в `/etc` (обычно в `/etc/php/apache/`) и измените параметр `upload_max_filesize` на более подходящее значение, скажем, 5 МБ. После этого потребуются перезапустить *Apache*. Есть и третий вариант: выбрать другой метод загрузки. Это может быть особенно кстати, если у вас много фотографий.

Если вы включили модуль `Archive Upload` [Загрузка архива], то сможете загружать zip-архив с изображениями вместо одной картинки. Выберите zip-файл, как если бы это была одна фотография, *Gallery* распакует его и добавит содержимое в текущий альбом. Если zip-файл содержит каталоги, для каждого из них будет создано по альбому.

Загрузка из локальных каталогов

Если у вас есть прямой доступ к серверу, то проще воспользоваться модулем для загрузки снимков из локального каталога. Он не входит в нашу версию *Typical*, поэтому зайдите в `Admin > Plugins` [Администрирование > Модули] и выберите вкладку `Get More Plugins` [Дополнительно]. Следуя подсказкам, загрузите модуль `Add Item` (и любые другие модули, которые захотите) в разделе `Import` [Импорт]. Теперь пройдите по ссылке `Web/Server` [Web/Сервер] в нижней части меню `Admin` и укажите каталог, предназначенный для загрузки. Все его содержимое будет добавлено, но он должен находиться вне корневого каталога сервера; например, неплохой выбор – `/var/www/localhost/gallery2/uploads`.

Скопируйте фотографии в этот каталог командой `scp`, если они находятся на другом компьютере. Кликнув по ссылке `Add Items` в своем альбоме, вы увидите несколько новых закладок. На закладке `From Local Server` [C локального сервера] укажите каталог загрузки, нажмите `Find Files` [Найти файлы] и отметьте файлы, которые намерены загрузить.

Другие языки

Хотя многие web-приложения написаны на PHP, это далеко не единственный язык. PHP был специально написан для встраивания в web-страницы, и это естественный выбор для решения подобных задач; но и у других языков есть свои преимущества. На Perl писали web-скрипты еще до того, как появился PHP, и пишут до сих пор. Ruby, особенно с каркасом *Ruby on Rails*, также популярен у некоторых разработчиков. Другие предпочитают Python, возможно, с каркасом *Django* (LXF105–108). Во многом выбор языка зависит от предпочтений разработчика или разработчиков, потому что для конечного пользователя разница невелика. Впрочем, популярность PHP можно объяснить и простотой, с которой он приспособляется под ваши нужды.

Скорая помощь

Не пытайтесь импортировать все фотографии сразу: вдруг вам захочется изменить настройки? Сначала добавьте пару штук и посмотрите, как они выглядят.

Вам явно ни к чему ситуация, когда весь канал уходит на загрузку картинок, поэтому кликните по ссылке Edit Album [Редактировать альбом] и установите один или несколько целевых размеров. Это размер изображения, отправляемого в ответ на запрос загрузки. *Gallery* изменит размер изображений по запросу и закаширует новые версии для увеличения производительности в дальнейшем. В секции Maintenance [Сопровождение] области Admin есть функция моментального создания всех миниатюр и изображений с измененным размером, что опять же экономит вам несколько драгоценных секунд.

На самом деле мы здесь лишь поверхностно коснулись возможностей *Gallery*. Следующей гаванью должна стать секция Users, где можно определять пользователей или группы и управлять доступом. Будьте поосторожнее с правами гостевых пользователей, если *Gallery* будет доступна через Интернет.

Затем одним глазом загляните в раздел Modules. Там масса интересных опций, с которыми можно поиграть, особенно если вы установили полный архив со всеми доступными модулями.

Ваша почта в сети

В первых двух частях этой серии мы настроили web-сервер и почтовый сервер IMAP, а теперь можем объединить их с помощью почтового приложения. Вариантов полно; большинство написано на PHP, но есть и на других языках, например, на Perl. Один из самых распространенных почтовиков на PHP — *SquirrelMail*.

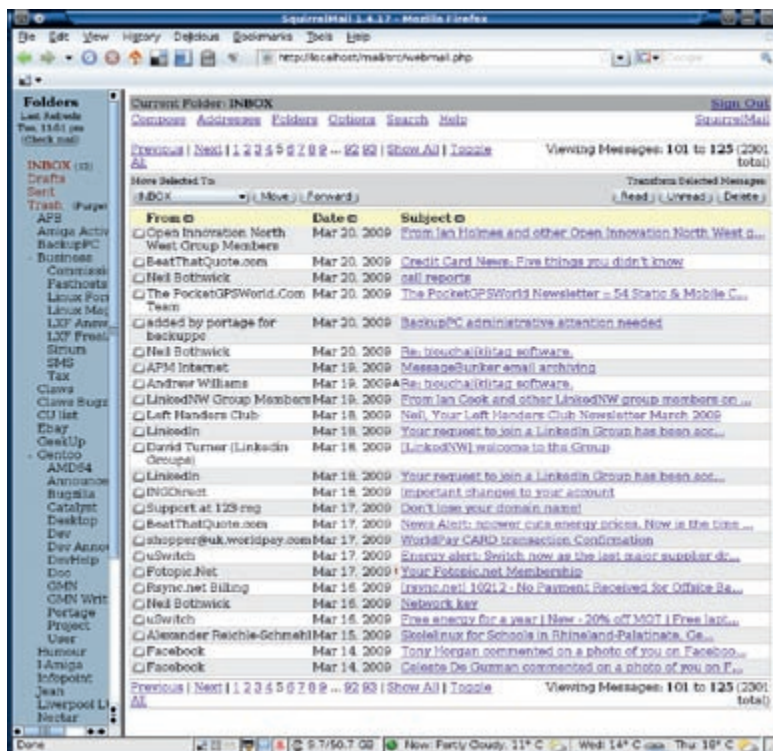
Как и все web-приложения, *SquirrelMail* можно установить с помощью менеджера пакетов, если у вас есть доступ с правами root, или из архива с сайта <http://www.squirrelmail.org>. Текущая версия — 1.4.18, поэтому скачайте с сайта архив [squirrelmail-1.4.18.tar.gz](http://www.squirrelmail.org). Пакет содержит подробные инструкции по установке, но для начала нужно создать два каталога за пределами DocumentRoot. Их владельцами должны быть пользователь и группа, от имени которых запущен сервер, поэтому установите пакет следующими командами, изменив пути в соответствии с вашей установкой:

```
mkdir -p /var/www/localhost/mail/{data,attachments}
chown apache: /var/www/localhost/mail/{data,attachments}
chmod 730 /var/www/localhost/mail/attachments
Теперь распакуйте архив SquirrelMail командами
tar xf squirrelmail-1.4.18.tar.gz -C /var/www/localhost/htdocs
ln -s squirrelmail-1.4.18 /var/www/localhost/htdocs/mail
```

Вторая команда добавит директорию **mail** как символическую ссылку на каталог **squirrelmail-x.y.z**. Благодаря этому можно установить более позднюю версию *SquirrelMail* вместе с более ранней и переключить символическую ссылку, когда вы убедитесь, что новая версия работает правильно. Тем самым уменьшается время простоя для пользователей. Другой вариант — создать подходящий каталог и перенести содержимое **squirrelmail-x.y.z** туда. Если этот сервер предназначен только для web-почты, файлы можно поместить прямо в **htdocs**. Теперь в оболочке, а не в браузере нужно запустить скрипт настройки. Под пользователем root выполните команды

```
cd /var/www/localhost/htdocs/mail
config/conf.pl
```

Нажмите D и выберите свой IMAP-сервер (у тех, кто следовал нашим предыдущим урокам, это *Dovecot*). Многие из этих настроек можно оставить в покое, но нужно зайти в Server Settings [Настройки сервера] и убедиться, что параметры серверов SMTP и IMAP указаны верно. Если оба сервера работают на том же компь-



➤ Вся моя почта без почтового клиента в пределах видимости. Установка *SquirrelMail* означает, что письма можно читать откуда угодно через Интернет.

ютере, что и *SquirrelMail*, оставьте их в виде **localhost**, в противном случае задайте их согласно настройкам сети. Если вы не запусти-

ли собственный SMTP-сервер (пока мы об этом не говорили), возможно, придется указать здесь сервер провайдера. В секции General Options [Общие настройки] приведите настройки каталогов данных

и вложений в соответствие с созданными вами. Остальные параметры пока оставьте как есть, сохраните свои настройки и выйдите из скрипта. Затем откройте в браузере адрес <http://ваш.хост/mail/src/configtest.php>. Так вы проверите, что настройки работают; если есть ошибки, то запустите **config/conf.pl** еще раз, чтобы исправить их, а в случае проблем с PHP отредактируйте файл **php.ini** (**configtest.php** выдаст ошибку, если в **php.ini** включена любая из опций **magic_quotes**).

Готово к работе

Теперь смело входите, кликнув по ссылке внизу тестовой страницы или открыв в браузере адрес <http://ваш.хост/mail>. Появится страница входа в систему; введите имя пользователя и пароль для почтового ящика. Ссылка Options [Параметры] в верхней строке позволяет изменять настройки показа почты. Это опции для отдельных пользователей; а если вам нужны глобальные изменения, запустите скрипт **config/conf.pl** еще раз.

Теперь у нас есть web-браузер, умеющий показывать ваши фотографии и работать с почтой. Вы также должны были получить основные навыки установки и настройки других web-приложений, так что можете спокойно ударяться в безумные эксперименты. Через месяц мы вставим последний кусочек мозаики с почтой, настроив SMTP-сервер, способный фильтровать спам и вирусы. **ЛЭР**

» Через месяц Установим SMTP-сервер для получения и сканирования почты.

Python: Сеть на

Часть 1: Недовольны, что контент раскидан где попало и как попало?
Ник Вейч приступает к объяснению, как подчинить сеть вашим целям.



» Получить идентификатор вашей местности можно прямо у онлайн-службы Yahoo, просто скопировав открытый ею URL.

ций – ну да, можно потратить время на распутывание данных или работы API, но зачем, если есть кому предоставить и данные, и руководство по их использованию?

Пошарив там и сям, мы решили, что нас устраивает служба погоды Yahoo. Она проста и обладает достаточной документацией, чтобы начать без особых усилий. Еще чуток порывшись, мы обнаружили <http://developer.yahoo.com/weather>, где предоставлено множество деталей и пара примеров использования службы. Очко в пользу Yahoo!

Метод Yahoo состоит в добавлении идентификатора местности в конец URL. В ответ служба генерирует RSS-ленту данных о погоде в указанном регионе. Это удобно во многих отношениях, поскольку означает, что даже код писать не придется. Идентификатор местности найти легко – согласно документации, достаточно пройти на главную страницу Yahoo! Weather, ввести город, и URL ваш (иногда, правда, при этом выскакивает название не города, а ближайшей к нему метеостанции).

«St. Petersburg» даст нам <http://weather.yahoo.com/forecast/RSEX0091.html> – значит, код нашей местности RSXX0091. Теперь у нас есть информация о местоположении. Инструкция также говорит, что можно добавлять параметр для выбора температуры по Фаренгейту или по Цельсию, и мы добавили к URL опцию по Цельсию **u=c**. Теперь протестируем URL, чтобы увидеть как он работает (о написании кода речи все еще нет).

Приличный браузер, вроде *Firefox*, способен отобразить RSS-ленту в удобной форме – надо всего лишь ввести ее URL: <http://weather.yahooapis.com/forecastrss?p=RSXX0091&u=c>.

Обработчик ленты

Теперь, зная, что нужная нам информация доступна в виде RSS-ленты, как заполнить ее в скрипт Python и декодировать? Скопировав указанный выше URL в *Firefox* и выбрав в меню Вид > Исходный код, вы увидите кучу информации плюс всякие заголовки



Наш эксперт

Ник Вейч

отныне не табельный номер (00147) – он вольная птица! Ведущий свободный Linux-журналист, отличный рассказчик и сторонник Python, он также более чем знаком с изготовлением всяких мулек на базе скриптов.

Что бы вам ни захотелось узнать, оно, вероятно, найдется в Сети. Это загадочное облако – иногда белое и пушистое, а иногда и полное грязных инсинуаций – содержит знания любого сорта, хотя до них надо еще докопаться. Готовы поспорить, что отыщется даже точная дата крещения «Наболиона Буонапарте», если порыться старательно.

Но не вся информация в сети статична, соединена разумным образом или вообще настолько хороша, насколько должно быть. Вот почему одним из заметных шагов пост-web 2.0 является мэшап [мэшап, mashup – технология построения web-сайта с объединением возможностей группы сторонних web-приложений, см. [http://ru.wikipedia.org/wiki/Мэшап_\(веб\)](http://ru.wikipedia.org/wiki/Мэшап_(веб)), – прим. пер.] – преобразование web-содержимого в новые удивительные формы. Добро пожаловать в мир панк-данных.

Куда дует ветер

Для начала нашего путешествия сделаем что-нибудь попроще, типа изменения фона рабочего стола в зависимости от погоды. Приложения и виджеты, предоставляющие такой сервис, уже имеются, а значит, данные должны быть легко добыть в Интернете. Быстрый поиск по словосочетанию «weather API data» непременно выдаст множество ссылок. Наши критерии: API должны быть понятны, охватывать как можно большую часть мира и иметь приличную документацию. Под них подпадают лишь пара из предложенных источников, в основном в связи с документа-

ВАШ ВКУС

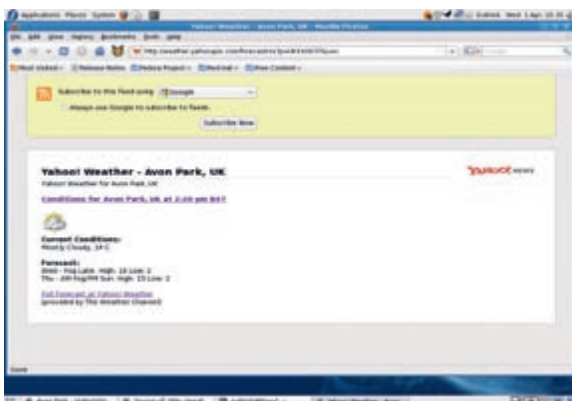
и прочее. Можно создать обработчик, который возьмет эти сырые данные и выудит из них соответствующие куски – но, как обычно и бывает, кто-то его уже создал. Для Python имеется библиотечный модуль под названием *Feedparser*, создающий объект Python из RSS-потока. Проще всего установить его через менеджер пакетов: другие способы малость утомительны.

После установки *Feedparser* наконец-то настает время кодирования. Сначала запустите Python из командной строки, чтобы понимать, с чем мы имеем дело. Вы окажетесь в интерактивной оболочке Python, где мы и введем:

```
>>> import feedparser
>>> url = "http://weather.yahooapis.
com/forecastrss?p=RSXX0091&u=c"
>>> data = feedparser.parse(url)
>>> data
```

Результатом действия последней строки будет выдача вереницы символов, которая представляет собой ленту. К счастью, хотя она и похожа на набор случайных данных, пронизанных частоклоком скобок, на деле это хорошо структурированный объект. Чтобы убедиться в этом, попробуйте ввести в оболочке Python следующее:

```
>>> for x in data:
...     print x
...
feed
status
version
encoding
bozo
headers
etag
href
namespaces
```



» Убедитесь, что передаваемый URL выдает RSS-ленту, через *Firefox*, который ее тут же отобразит.

Регулярные выражения

Регулярные выражения заполнили все вокруг. Они бывают и простенькими, и дьявольски сложными. Вкратце, регулярное выражение – это группа знаков, определяющая некую конфигурацию символов в строке. Имеются также специальные знаки, типа местоимителей, которые подразумевают любой символ. Кроме того, есть еще списки, группы и даже операторы для объединения соответствий – можно, например, создать регулярное выражение, полностью описывающее все базы для поиска.

При создании шаблона лучше всего воспользоваться инструментом, который поможет вам проверить его корректность – даже один неверный символ чреват катастрофой! Среди лучших подобных инструментов – онлайн-построитель регулярных выражений <http://gskinner.com/RegExr/>. Вставьте в него пример текста и протестируйте свои навыки создания шаблонов. Не помешает и обратиться к документации: <http://docs.python.org/library/re.html>.

entries

Это цикл, пробегающий по всем объектам внутри *data*. Одно из замечательных свойств Python – легкость работы с объектами, и даже возможность заставить их рассказать о себе. Например, пусть мы хотим точно знать, с каким типом объектов работаем:

```
>>> type(data)
<class 'feedparser.FeedParserDict'>
```

Ну, отчасти нам это помогло. Если вы уже сталкивались с Python, то, вероятно, слышали про объекты-словари (ассоциативные массивы) – они просто хранят данные в виде пар **ключ = значение**. Объекты, список которых мы только что выводили, являются в данном случае ключами. Обратившись к документации *Feedparser*, мы получим несколько более внятное представление, что хранят такие ключи, поскольку они стандартны для объектов *Feedparser*. Из них для нас важен ключ **entries**. Он хранит объект-список отдельных записей RSS-ленты, которые и составляют ее реальное содержимое.

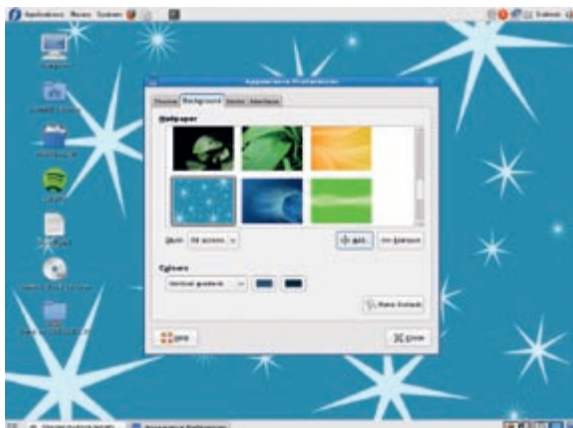
Списки Python нумеруются с индекса 0, поэтому, чтобы сослаться на объект первой записи, следует использовать:

```
>>> data.entries[0]
{'updated': u'Thu, 28 May 2009 2:00 am MSD', 'yweather_
condition': u'', 'updated_parsed': ...
>>> for x in data.entries[0]:
...     print x
...
updated
yweather_condition
updated_parsed
links
title
summary_detail
geo_lat
```

Скорая помощь

Python требователен к синтаксису. Он требует выделять отступами блоки многострочных выражений, функций и условных операторов. Количество пробелов в отступе роли не играет: главное, чтобы оно сохранялось внутри блока.

» Пропустили номер? Узнайте на с. 103, как получить его прямо сейчас.



» Кого же не согреет милый снегопад на SVG-обоях рабочего стола? Они могут быть вашими.

summary
guidislink
title_detail
link
geo_long
yweather_forecast
id

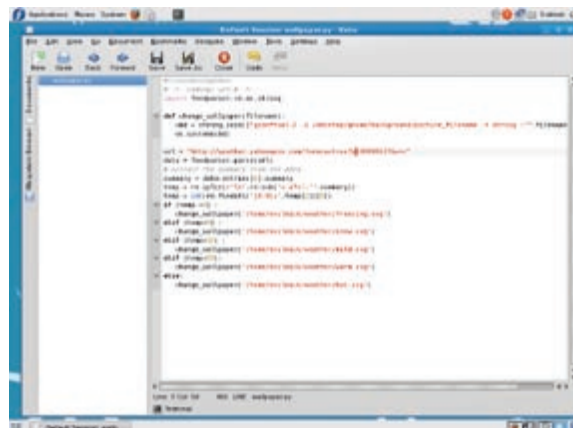
И вновь мы имеем объект-словарь с парами ключей и значений. На сей раз они определяются самой XML-структурой ленты, поэтому для данного объекта нет модуля с голубой каемочкой; зато его элементы документированы на сайте Yahoo. После недолгого изучения становится ясно, что полезна нам будет сводка (summary) текущих погодных условий, включая температуру.

Налицо небольшая проблема: необходимые данные отнюдь не сосредоточены в одном поле нам на радость. Эта часть ленты – HTML-код для отображения на web-странице, а нам нужен только текст.

Регулярные выражения

От регулярных выражений (также известных как «регекспы» [regex]) никуда не денешься – они будут все чаще встречаться на нашем пути переустройства мира. Регулярные выражения – это просто способ выполнения поиска и замены, и хотя они смахивают на криптограммы, разобраться в них не так уж трудно. В данном случае нам надо всего лишь избавиться от нудных HTML-тэгов. Мы, вероятно, и без того извлекли бы нужные данные, но подобный опыт пригодится нам в будущем, если мы захотим приспособить наш скрипт к работе с различными источниками.

Мы импортируем Python-модуль `re` (он включен в качестве одной из стандартных библиотек, так что ничего загружать и устанавливать не придется) и привлечем его к работе с этим текстом. На нашем уроке нет места на детальное объяснение работы регулярных выражений, а за краткими сведениями об-



» Сберегите свой рассудок — для написания кода Python пользуйтесь IDE; это сильно упрощает обнаружение ошибок.

ратитесь ко врезке Регулярные выражения на стр. 67 или статье из **LXF80**: http://wiki.linuxformat.ru/index.php/LXF80:Регулярные_выражения.

Выражением мы хотим охватить все, что обрамлено знаками «больше» и «меньше», используемыми для обозначения HTML-тэгов. Это очень легко; выражение будет выглядеть как `<.+?>`, за которым следует шаблон любого символа, повторенный любое количество раз, то есть `.+?` и в конце `>`. `+` – то же, что и `+`, но это «ленивое» соответствие: оно отвечает наискратчайшей корректной строке. А нам того и надо – все, что находится между `<>`; других вариантов нет.

```
>>>summary = data.entries[0].summary
>>>import re
>>>pattern = '<.+?>'
>>>temp = re.sub(pattern, '', summary)
>>>temp
u'\nCurrent Conditions:\nLight Rain Shower, 12 C\nForecast:\n
Thu – Partly Cloudy. High: 25 Low:12\nFri – Mostly Sunny. High:
19 Low:
9\n\nFull Forecast at Yahoo! Weather\n(provided by The Weather
Channel)'
```

Теперь текст очищен от тэгов HTML; но в нем остались переводы строк. Вообще-то есть стандартный модуль для преобразования такой строки в список, но у нас уже загружен модуль `re`, и мы вполне можем им обойтись. Для поиска символов новой строки надо объяснить Python, что мы хотим использовать чистое (raw) значение строки – поместив перед ней символ `r`.

```
>>> temp = re.split(r'\n', temp)
>>> temp
[u'', u'Current Conditions:', u'Light Rain Shower, 12 C', u'Forecast:',
u'Thu – Partly Cloudy. High: 25 Low: 12', u'Fri – Mostly Sunny. High:
19 Low: 9', u'', u'Full Forecast at Yahoo! Weather', u'(provided by
The Weather Channel)']
>>>temp[2]
u'Light Rain Shower, 12 C'
```

Как видите, мы предположили, что необходимую строку содержит третий элемент полученного списка. Чтобы добыть из него температуру, воспользуемся другим регулярным выражением, отбирающим из строки исключительно цифры.

```
>>>temp = re.findall(u'[0-9]+', temp[2])[0]
>>>temp
u'12'
>>>>> temp = int(temp)
>>> temp
12
```

Почему Python?

Можно, естественно, писать скрипты и приложения для обработки web-данных на любом другом языке; так почему же мы выбрали Python, а не C#, например? На то есть веские причины. Python – язык простой и понятный, код на нем легко пишется и (что, вероятно, еще важнее) легко читается. Он обладает

прекрасными возможностями для работы с текстом (которым по большей части и являются наши данные), он кросс-платформенный и снабжен бездной полезных библиотек для web-служб и протоколов. Используя Python и пару библиотек, можно мигом создать рабочее приложение или скрипт.

Финальный шаг – употребить встроенное в Python преобразование типов, превратив строку со значением температуры в целое число: числа проще сравнивать. (В реальном скрипте мы можем, эффективности ради, объединить некоторые из приведенных этапов, но это не даст большого выигрыша для приложения, а ясность пошаговой записи будет утрачена.) Осталось только создать то, что мы посулили в начале урока: обои рабочего стола, меняющиеся в зависимости от погоды.

Перемена погоды

Как переделать обои? А для этого в Python есть вызов внешних команд. В Gnome можно изменить заставку командой **gconftool-2**, устанавливающей переменную окружения, в которой хранится размещение и имя файла обоев. Оно бы и достаточно. Но поскольку у нас возможны варианты, лучше оформить это в виде функции.

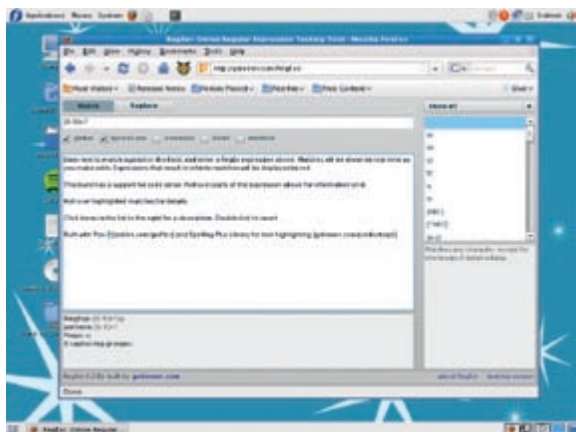
Несложная функция не особо напрягает вам мозг. Python обладает простой конструкцией: описание функции и ее параметров, за которым следует текст с отступом. Ее даже можно создать в интерактивной оболочке:

```
>>> def change_wallpaper(filename):
...     cmd = string.join(["gconftool-2 -s
... /desktop/gnome/background/picture_filename -t string
... \"", filename, "\""], '')
...     os.system(cmd)
...
>>> change_wallpaper('plop.jpg')
```

Первая строка функции создает команду оболочки, а вызов **os.system** выполняет ее. Вызов функции меняет переменную окружения, и загружается новое изображение. Подставленное нами имя файла **filename** – просто параметр; в реальности, хорошо бы хранить ваши изображения где-нибудь внутри вашего домашнего каталога, допустим, в подкаталоге с именем **weather**, и называть их согласно погодным условиям.

Сейчас Gnome допускает в качестве обоев SVG-изображения, и вы можете создать такую изысканную масштабируемую картину для вашего рабочего стола.

Не знаем, как по-вашему, но мы думаем, что здесь должно быть пять изображений: freezing [мороз], cold [холод], mild [средне], warmish [тепло] и hot [жарко]. Диапазоны температур выбирайте в соответствии с вашим климатом, а мы присвоим следующие: ниже 0°C – мороз, 0–8 – холод, 9–15 – средне, 16–25 – тепло, выше – жарко. В некоторых языках для реализации проверки предусмотрена конструкция case/switch. В Python ее нет, и мы нагородим огород из конструкций if/elif/else, вот так:



» Регулярные выражения не всякому по плечу; нам поможет служба <http://gskinner.com/RegExr/>.

Форматы web-данных

Мы осознали, что есть масса web-приложений, способных доставить нам данные. Но вы ведь не ждали, что все будет просто, не так ли? Протоколов или способов предоставления данных немало, и некоторые сайты (например, Flickr) даже предусматривают более одного варианта. Еще одно неудобство – разнородность в их использовании на сайтах. В нашей серии уроков мы постепенно все это рассмотрим.

```
>>> if (temp < 0):
...     change_wallpaper('/home/evilnick/weather/freezing.svg')
... elif (temp < 9):
...     change_wallpaper('/home/evilnick/weather/snow.svg')
... elif (temp < 16):
...     change_wallpaper('/home/evilnick/weather/mild.svg')
... elif (temp < 26):
...     change_wallpaper('/home/evilnick/weather/warm.svg')
... else:
...     change_wallpaper('/home/evilnick/weather/hot.svg')
...
```

Объединив все это в один скрипт, мы получим нечто подобное следующему (остается только разжиться изображениями):

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import feedparser, re, os, string
def change_wallpaper(filename):
    cmd = string.join(["gconftool-2 -s /desktop/gnome/
background/picture_filename -t string \"", filename, "\""], '')
    os.system(cmd)
    url = "http://weather.yahooapis.com/
forecastrss?p=RSXX0091&u=c"
    data = feedparser.parse(url)
    # extract the summary from the data
    summary = data.entries[0].summary
    temp = re.split(r'\n', re.sub('<.+?>', '', summary))
    temp = int(re.findall('[0-9]+', temp[2])[0])
    if (temp < 0):
        change_wallpaper('/home/evilnick/weather/freezing.svg')
    elif (temp < 9):
        change_wallpaper('/home/evilnick/weather/snow.svg')
    elif (temp < 16):
        change_wallpaper('/home/evilnick/weather/mild.svg')
    elif (temp < 26):
        change_wallpaper('/home/evilnick/weather/warm.svg')
    else:
        change_wallpaper('/home/evilnick/weather/hot.svg')
```

Конечно, даже при буйном воображении это не полноценное приложение, а небольшой сценарий, но его вполне можно взять за основу. На данном уроке мы взяли данные из сети и автоматически совместили их с контекстом рабочего стола. Мы увидели, как просто выглядят RSS-ленты и как работать с объектами в Python; постигли ужасы регулярных выражений; и рассмотрели совершение из Python системных вызовов для выполнения внешних команд. Вот так мы и можем, исследуя сервисы всемирной паутины, брать что угодно и ставить себе на службу.

Скрипт легко расширить (например, позволив пользователям выбирать местоположение) или преобразовать его в апплет. Покамест он завершился бы с ошибкой при отсутствии подключения к Интернету, что не есть хорошо, но на следующих уроках мы изучим, как с этим справиться. Оставайтесь с нами! **LSX**

» **Через месяц** Создайте себе чат-бота, чтобы разыграть друзей.

VirtualBox: Python

Графический интерфейс и диалоговые окна — далеко не единственный способ взаимодействия с виртуальными машинами *VirtualBox*. Николай Иготти покажет, как делать это из программ на Python.



Наш эксперт

Николай Иготти разрабатывает *VirtualBox* в компании Sun Microsystems, а в свободное время разбирается, почему в нем не запускается какая-нибудь экзотическая ОС.

В широком спектре доступных решений для виртуализации продукт *VirtualBox* (www.virtualbox.org) от Sun Microsystems занимает достойное место. Это высокопроизводительное решение с открытым кодом, работающее на Linux, Windows, Mac OS X, Solaris и FreeBSD, позволяет запускать самые различные гостевые ОС со скоростью, близкой к скорости реального оборудования. Одним из интересных свойств данного продукта является полная открытость всех интерфейсов, то есть возможность взаимодействовать с *VirtualBox* из приложений, написанных на популярных языках программирования. Официально поддерживаются C, C++, Java, Perl и Python, но можно использовать и любой другой язык, для которого реализован вызов SOAP-методов.

Я разрабатывал поддержку Python в *VirtualBox*, и хотел бы подробнее рассказать о различных тонких моментах и интересных элементах функциональности. Чтобы разговор был более предметным, рекомендую скачать последнюю версию *VirtualBox* 2.2 для вашей операционной системы, а также SDK *VirtualBox* с www.virtualbox.org/wiki/Downloads (требуется бесплатная регистрация). Желательно также установить хотя бы одну виртуальную машину, чтобы нам было чем управлять.

Как это работает

Внешние интерфейсы *VirtualBox* описаны в документации к SDK (на английском языке), а XML-файл `bindings/VirtualBox.xidl` предоставляет полную информацию об устройстве API. Кроме этого, может быть интересно исследовать исходные коды примеров, предоставляемые с SDK.

Доступ к внешним интерфейсам технически осуществляется двумя способами: через непосредственный вызов методов COM, XPCOM (компонентной технологии, происходящей от Mozilla) или через SOAP (то есть специально сформированные HTTP-запросы, обрабатываемые сервером *vboxwebsrv*). Первый вариант гораздо быстрее, и предоставляет доступ к более широкой функциональности: например, уведомления об изменении состояния. К сожалению, он требует, чтобы вызывающий процесс и *VirtualBox* выполнялись на одной машине, что может быть не всегда удобно. Второй способ позволяет использовать удаленные сервера, но имеет ограничения по функциональности и производительности, а также требует запуска специального приложения на каждом из серверов.

Интерфейсы для Python разрабатывались таким образом, чтобы по возможности скрыть это различие и использовать один и тот же код для различных сценариев доступа. Тем не менее, данная статья будет в основном посвящена Python и XPCOM.

В качестве примера нетривиального приложения, использующего *VirtualBox* из Python, рассмотрим интерактивную оболочку, поставляемую в комплекте с SDK. Один и тот же сценарий (`shellcommon.py`) может работать как через SOAP, так и на локальной машине. Вся раз-

ница в инициализационном коде (`vboxshell.py`), который отличается для COM, XPCOM и режима web-сервиса (SOAP). Если вы работаете в Linux или Solaris, перед использованием необходимо установить переменную `VBOX_PROGRAM_PATH` следующим образом:

```
export VBOX_PROGRAM_PATH=/opt/VirtualBox
```

Здесь `/opt/VirtualBox` — это название директории, в которую установлен *VirtualBox* в вашей системе. Также, если SDK находится в `/opt/VirtualBox-sdk`, переменная `PYTHONPATH` должна быть установлена как

```
export PYTHONPATH=/opt/VirtualBox-sdk/bindings/xpcom/python:$VBOX_PROGRAM_PATH
```

чтобы Python мог найти необходимые модули. После настройки окружения можно просто запустить оболочку командой

```
python /opt/VirtualBox-sdk/bindings/xpcom/python/samples/vboxshell.py
```

и вы увидите приглашение, как на рисунке. Чтобы получить список команд, известных оболочке, наберите `help`. Можете поэкспериментировать в свое удовольствие.

Расширяем оболочку

Как можно видеть, сама по себе оболочка достаточно функциональна и полезна, но нам сейчас будет интересна возможность ее расширения. Предположим, к примеру, что вам необходимо написать команду, которая создает виртуальную машину со смонтированным ISO-образом и запускает ее (скажем, для целей автоматизированного тестирования какого-нибудь дистрибутива). Назовем эту команду *makevm*.

Откроем `shellcommon.py` в редакторе и добавим строку `'makevm': ['Make VM with given ISO image', makevmCmd]`,

к массиву `commands`. Она просто регистрирует нашу команду в списке известных оболочке.

Теперь надо реализовать действия, выполняемые при вызове *makevm*, то есть написать тело функции `makevmCmd()`, на которую мы ссылаемся выше.

```
def makevmCmd(ctx, args):
    # Имя новой машины
    name=args[1]
    # Путь к файлу образа (можно сделать частью команды)
    hddPath= "/tmp/MyUbuntu/hdd.vdi"
    # Путь к образу ISO содержащему дистрибутив
    isoPath=args[2]
```

Переменная `ctx` содержит контекст исполнения, откуда можно получить объекты класса *VirtualBox* и *Session*, а также доступ к константам. Пример использования контекста можно найти в коде `shellcommon.py`. Для команды *makevm* нам надо получить доступ к объекту `vb`, представляющему собой экземпляр запущенной виртуальной машины *VirtualBox*.

```
vb=ctx['vb']
```

Далее следует найти или создать подключить к *VirtualBox* ISO-образ, а также определить (или, при необходимости, создать) виртуальный жесткий диск, на который мы будем устанавливать нашу гипотетическую систему:

» Оболочка *VirtualBox*: с виду невзрачная, но мы дополним ее новыми возможностями.



управляет

```
isoImg=vb.findDVDImage(isoPath)
if not isoImg:
    isoImg=vb.openDVDImage(isoPath, None)
hddImg=vb.findHardDisk(hddPath)
if not hddImg:
    hddImg=createHardDisk("VDI", hddPath)
progress=hddImg.createBaseStorage(4*1024,
    ctx['ifaces'].HardDiskVariant.Standard)
progress.waitForCompletion(-1)
hddImg=vb.openHardDisk(hddPath,
    ctx['ifaces'].AccessMode.ReadWrite)
```

Как можно видеть, генерация образа жесткого диска происходит в несколько этапов: сначала мы создаем устройство вызовом `createHardDisk()`, указав желаемый тип (VDI, родной для *VirtualBox* формат) и путь в файловой системе хост-ОС, затем выделяем под него 4 ГБ методом `createBaseStorage()` и ждем завершения этой операции. Наконец, винчестер регистрируется в системе методом `openHardDisk()`.

Ключ на старт!

Давайте будем для определенности считать, что мы тестируем сборки Ubuntu. Виртуальные машины с предустановками для этого дистрибутива будем создавать в каталоге `/tmp/MyUbuntu`. Для этих целей нам потребуется сессионный объект — `Session`.

```
session = ctx['mgr'].getSessionObject(vb)
mach = vb.createMachine(name, "Ubuntu", "/tmp/MyUbuntu", "")
vb.registerMachine(mach)
# Запомнить уникальный идентификатор машины
id=mach.id
print "created machine with UUID",id
```

Этот код не должен вызывать затруднений. Мы получаем сессионный объект через контекст, создаем виртуальную машину, расположенную в `/tmp/MyUbuntu`, с указанным пользователем именем и типом гостевой ОС 'Ubuntu', и регистрируем ее. Наконец, мы запоминаем уникальный идентификатор машины и выводим его на консоль.

Полученная нами виртуальная машина не имеет нужной периферии — исправим этот факт, смонтировав наш DVD-образ и подключив к ней жесткий диск. Обратите внимание на первую строку — перед изменением виртуальной машины необходимо получить на нее блокировку, чтобы исключить конфликты одновременного обращения.

```
# Получить блокировку на доступ к машине
vb.openSession(session, id)
mach=session.machine
mach.DVDDrive.mountImage(isoImg.id)
mach.attachHardDisk(hddImg.id, "IDE", 0, 0)
# Сохранить внесенные изменения
mach.saveSettings()
# Закрыть сессию
session.close()
```

В данном случае мы подсоединяем жесткий диск к виртуальному контроллеру IDE. Можете выбрать SATA или SCSI — это не принципиально.

Теперь все готово к работе — осталось только запустить виртуальную машину, терпеливо дожидаясь ее остановки и завершить сеанс.

```
progress = vb.openRemoteSession(session,
    id, "gui", "")
progress.waitForCompletion(-1)
session.close()
return 0
```

Последняя строка уведомляет оболочку, что по завершении команды ей необходимо продолжить свою работу. На первый взгляд выглядит непросто, но если вдуматься, мы только что написали код, который создает виртуальную машину, виртуальный жесткий диск, монтирует образ DVD и запускает ее. И все это — примерно 20 строчек.

И много чего еще

Давайте протестируем созданную нами команду. Запустите оболочку и введите

```
vbox> makevm Test /ISO/Linux/Ubuntu/ubuntu-8.10-desktop-i386.iso
```

Вы увидите виртуальную машину, готовую установить новую ОС. Более того, мы можем использовать команду оболочки `guest` и начать установку, не переключая окна.

```
vbox> guest Test console.keyboard.putScancode([0x5a,0x1c,
    0x5a,0x1c])
```

Здесь мы эмулируем двойное нажатие на клавишу Enter путем отправки гостевой системе соответствующих скан-кодов. Подробнее про скан-коды клавиатуры можно прочесть по адресу: <http://www.quadibloc.com/comp/scan.htm>. Домашнее задание: измените команду так, чтобы перед установкой автоматически выбирался русский язык.

Вы также можете эмулировать любые движения мыши; например,

```
vbox> guest Test console.mouse.putMouseEvent(20,20,0,0)
```

сдвинет мышь на 20 пикселей вправо и вниз. Полный список методов взаимодействия с гостевой ОС можно получить в документации к интерфейсу `IConsole`, идущей в комплекте с SDK *VirtualBox*.

Но и это еще не все. Оболочка позволяет выполнить любой код Python, использующий API *VirtualBox*: для этих целей предназначена команда `eval`. Например,

```
eval for m in getMachines(ctx):
```

```
print
```

```
m.name,m.getNetworkAdapter(0).MACAddress
```

напечатает MAC-адреса первой сетевой карты всех виртуальных машин.

Как можно видеть, достаточно простые и интуитивно понятные вызовы функций позволяют выполнять весьма сложные операции над виртуальными машинами, и использовать все мощь *VirtualBox*.

На текущий момент управление *VirtualBox* из Python возможно на хостах Linux, Solaris и Mac OS X; начиная с 3.0, будет реализована поддержка Windows. Еще раз обращаю ваше внимание на то, что аналогичные интерфейсы можно использовать через SOAP на всех платформах, которые поддерживаются *VirtualBox*.

Полный код примера в формате патча для *shellcommon.py* вы найдете на прилагаемом к журналу диске. В заключение хочу сообщить, что на все вопросы, связанные с использованием *VirtualBox* (и, в частности, API Python) вам с удовольствием ответят на форуме forums.virtualbox.org. LXF



➤ Готовая виртуальная машина с Ubuntu — одной командой: при желании можно даже не нажимать.

GDB: Избавимся

Один из основных принципов открытого ПО – «больше глаз – меньше ошибок»; так чего же вы ждете? **Андрей Боровский** познакомит вас с необходимым инструментарием.



Наш эксперт

Андрей Боровский
Программирует для Linux уже примерно 2920 дней, 1460 из которых он провел, не покидая командной строки GDB.

При первом знакомстве отладчик *GNU Debugger (GDB)* может напугать программистов, привыкших к встроенным отладчикам графических интегрированных сред разработки. На самом деле все не так страшно. Выучив несколько простых команд, вы сможете сделать с помощью отладчика GNU все, что вы могли бы сделать в плане отладки в средах Microsoft или Borland. Выучив еще несколько команд, вы сможете делать такое, что пользователям графических IDE и не снилось. И хотя в наше время безалкогольного шампанского и бескофеинового кофе в Linux появились свои графические IDE со встроенными функциями отладки (*Eclipse*, *Qt Creator*), изучение возможностей *GDB* все равно будет вам полезно, поскольку «за кулисами» указанные среды вызывают именно его.

Выдумывать специальные примеры для отладки – дело неблагодарное, поэтому мы рассмотрим этот процесс на примере программы *Cuneiform*, «доводкой» которой я занимаюсь уже некоторое время (*LXF118*). Не пугайтесь, вам не придется вникать в логику работы *Cuneiform*. Все выбранные нами примеры самоочевидны.

Приступаем к отладке

Чтобы извлечь из *GDB* максимум, программу, предназначенную для отладки, следует скомпилировать с дополнительной отладочной информацией. Ее добавлением управляет ключ **-g** команд **gcc** и **g++**. Файл программы или библиотеки, скомпилированный с добавлением отладочной информации, занимает больше места, чем обычный, однако вам вовсе не обязательно перекомпилировать модуль по окончании отладки. Удалить отладочную информацию из исполняемого файла можно в любой момент с помощью утилиты **strip**.

Подготовив двоичные модули *Cuneiform*, мы запускаем отладчик с помощью команды **gdb**. Перед нами появляется приглашение командной строки: **(gdb)**. Далее в примерах команд мы будем ука-

зывать его, чтобы отличать команды интерактивного режима отладчика от других. Интерактивный режим *GDB* похож на таковой в командной оболочке *Bash*. Более того, в *GDB* задействованы многие полезные функции командной оболочки: например, автоматическое завершение команды с помощью клавиши табуляции, преобразование символа `~` и история команд. Есть даже свой аналог сценариев оболочки. Для выхода из *GDB* служит команда **quit**.

Мы должны указать отладчику имя программы, которую собираемся отлаживать. Это можно сделать во время запуска *GDB*:

```
$ gdb cuneiform
```

А можно и после запуска *Cuneiform* с помощью команды **exec-file**:

```
(gdb) exec-file cuneiform
```

В любом случае, *GDB* лишь подготовит среду для отладки программы, но не запустит ее на выполнение. Вообще, следует помнить, что отладчик всегда находится в одном из трех режимов: отлаживаемая программа работает, отлаживаемая программа приостановлена и программа не выполняется. Большую часть команд отладчика можно вводить в последних двух режимах. Для запуска программы служит команда **run**:

```
(gdb) run rf2.bmp -l rus_fra -o out.txt
```

Все аргументы, которые мы указываем в команде **run**, передаются непосредственно программе, запускаемой на отладку. После ввода приведенной выше команды мы увидим:

```
Starting program: /usr/local/bin/cuneiform rf2.bmp -l rus_fra -o out.txt
```

```
Cuneiform for Linux 0.7.0 (multilang)
```

```
Program exited normally.
```

Текст, появляющийся между строками «Starting program...» и «Program exited normally.», представляет собой консольный вывод отлаживаемой программы. Пока что отладчик не сообщил нам каких-либо полезных сведений о ней, но даже от запуска приложения в этом режиме может быть толк. Если во время выполнения произойдет ошибка сегментации, отладчик, помимо прочего,

Помощь придет

Для получения справки по работе с *GDB* вовсе не обязательно выходить из интерактивного режима – достаточно ввести команду **help** или **help имя_команды**. Например,

```
(gdb) help exec-file
```

распечатает справочную информацию о команде **exec-file**. Получать справку о конкретной команде имеет смысл, если вы знаете, что именно вам нужно. Если же вы – новичок в работе с отладчиком, вам прежде всего понадобится информация о том, какие вообще команды существуют в системе. В интерактивном режиме отладчика GNU программисту доступны сотни команд (большая часть которых никогда ему не понадобится).

Для удобства навигации в справочной системе GDB они сгруппированы по нескольким разделам:

- » **aliases** – псевдонимы других команд
- » **breakpoints** – команды, управляющие точками останова
- » **data** – команды, позволяющие манипулировать данными программы в процессе отладки
- » **files** – команды для работы с файлами
- » **internals** – внутренние команды GDB
- » **obscure** – команды для продвинутых пользователей
- » **running** – команды, управляющие запуском отлаживаемой программы

- » **stack** – команды, управляющие просмотром стека
- » **status** – команды (тысячи, тысячи их), которые сообщают различные сведения об отлаживаемой программе, отладчике и системе
- » **support** – разные полезные команды, управляющие отладчиком
- » **tracepoints** – команды, которые управляют трассировкой
- » **user-defined** – команды, определенные пользователем.

Например, для того, чтобы получить сведения о командах, управляющих точками останова, введите:

```
(gdb) help breakpoints
```

» Пропустили номер? Узнайте на с. 103, как получить его прямо сейчас.

ОТ ОШИБОК

покажет нам состояние стека процедур на момент ее возникновения. Также, если программа зависнет и мы завершим ее выполнение с помощью Ctrl+C, нам будут выданы сведения о том, в каком месте программы была прервана ее работа, например:

```
Program received signal SIGINT, Interrupt.
0xb7d8c8b4
in IntervalsBuild (y=979) at /home/andrei/bazaar/cuneiform-
multilang/cuneiform_src/Kern/rblock/sources/c/ltexcomp.c:144
144 while (x < nWidth && (pLine[x] & BlackMask) == 0)
```

Здесь мы видим имя функции, во время выполнения которой была прервана работа программы (то есть **IntervalsBuild**), значение параметров, с которыми она была вызвана (в нашем случае — один параметр, **y**, со значением 979), имя файла и номер строки исходного текста, соответствующего прерванному фрагменту, и саму строку кода.

Очень часто даже этой информации достаточно для того, чтобы вызвать озарение у программиста, ищущего ошибку. Но **GDB** способен показать нам гораздо больше. Мы можем просмотреть значения переменных, доступных нам в данном контексте (как минимум, это **x**, **nWidth**, **pLine**, и **BlackMask**). Для этого воспользуемся командой **print**:

```
(gdb) print x
```

Результат:

```
$1 = 681
```

Команда:

```
(gdb) print pLine[x]
```

Результат:

```
$2 = 0 '\0'
```

Таким образом мы узнаем, что на момент прерывания переменная **x** содержала значение 681, а элемент массива **pLine[681]** — символ **'\0'**. Если мы забыли или не знали, какой тип имеет переменная **pLine**, мы можем узнать это, не заглядывая в исходники:

```
(gdb) ptype pLine
```

```
type = unsigned char *
```

Но и это еще не все. Можно изменить значение переменной:

```
(gdb) set x=0
```

а затем возобновить выполнение программы с помощью команды **continue**.

Точки останова

Поскольку современные микропроцессоры работают очень быстро, комбинация клавиш Ctrl+C далеко не всегда останавливает программу именно там, где нам нужно. Как правило, мы хотим, чтобы ее выполнение было прервано в заранее определенном месте. Для этого и служат точки останова (breakpoint). Наберите

```
(gdb)break copy_text
```

Эта команда создает точку останова в начале функции **copy_text** (ленивые могут ввести просто **b copy_text**). В ответ на ввод команды **GDB** скажет:

```
Breakpoint 1 at 0xb7f57f35: file /home/andrei/bazaar/cuneiform-
multilang/cuneiform_src/Kern/puma/c/spcheck.cpp, line 85.
```

Это значит, что отладчик нашел интересующую нас функцию и строку кода, соответствующую точке останова. Теперь мы снова запускаем программу **Cuneiform** на выполнение с помощью команды **run**. Когда процессор достигнет точки останова (если это случится), выполнение программы будет прервано, и на экране

появится следующее сообщение:

```
Breakpoint 1, copy_text (word=0x9452338) at
/home/andrei/bazaar/cuneiform-multilang/cuneiform_
src/Kern/puma/c/spcheck.cpp:85
85 word->text = (Word8 *) malloc((word->wlen +
1)*sizeof(Word8));
Current language: auto; currently c++
```

Мы снова видим имя функции, выполнение которой было приостановлено, имя и значение переданного ей параметра и строку исходного текста, которая будет выполнена далее. Команда

```
(gdb) ptype word
```

сообщает нам следующее:

```
type = struct _SPWord {
  Bool32 is_latin;
  CSTR_rast begin;
  CSTR_rast end;
  Word8 *text;
  int wlen;
} *
```

Мы видим, что переменная **word** представляет собой указатель на экземпляр структуры **_SPWord**. Чтобы узнать значения полей структуры, наберите

```
(gdb) output * word
```

Команда **output** делает то же, что и команда **print**, но при этом не утруждает себя излишним форматированием. Результат выполнения команды выглядит так:

```
{is_latin = 1, begin = 0x93e7008, end = 0x93c2a20, text =
0x1cc08f3f <Address 0x1cc08f3f out of bounds>, wlen = 6}
```

Обратите внимание, что отладчик любезно сообщил нам о том, что поле **text** указывает на невыделенную область памяти. Эта информация может очень пригодиться при поиске ошибок, но не в данном случае. Поле **text** просто еще не инициализировано (это произойдет в следующей строке).

Если вам хочется изучить работу программы подробнее, начиная с этого момента, к вашим услугам команды **step** и **next**. Они выполняют операции, соответствующие одной строке исходного текста (когда это возможно), и снова приостанавливают выполнение программы. Серия вызовов **step** или **next** позволяет вам производить отладку построчно. В нашем примере, в результате выполнения команды

```
(gdb) step
```

мы переходим к следующей строке:

➤ Запуск программы из-под отладчика и Ctrl-C позволяют получить о ней море полезной информации, даже если вы ни о чем не спрашиваете.

Скорая помощь

Собирая программу с отладочной информацией, не забудьте отключить и оптимизацию (-O0). Код оптимизированной программы не совсем похож на ваши исходники, и это может привести к трудноразрешимым проблемам.

```
86 CSTR_rast rast = word->begin;
```

Разница между командами **step** и **next** заключается в том, что **step** «заходит» в каждую функцию, вызов которой встречается у нее на пути, и проходит ее построчно, тогда как **next** «перепрыгивает» вызов функции, выполняя его за один шаг. Команда **step** по смыслу соответствует командам **Trace into/Step into** интегрированных отладчиков Borland и Microsoft, тогда как команда **next** соответствует их командам **Step over**. Запуск приостановленной программы осуществляется командой **continue**. Вместо полных имен команд **step**, **next** и **continue** можно использовать их однобуквенные псевдонимы – **s**, **n** и **c** соответственно. Если вы хотите, чтобы программа сделала сразу 5 шагов, можете просто скомандовать

```
(gdb) s 5
```

Если же построчной отладки вам недостаточно, перейдите на уровень отслеживания выполнения отдельных инструкций процессора с помощью команд **stepi** и **nexti**.

Точки останова не обязательно должны располагаться в начале функций. Например, если мы скомандуем

```
(gdb) break 97
```

следующая точка останова будет создана в строке 97 файла `/home/andrei/bazaar/cuneiform-multilang/cuneiform_src/Kern/puma/c/spcheck.cpp` (поскольку он является текущим). Если вы хотите создать точку останова на заданной строке в другом исходном файле, команда должна выглядеть так:

```
(gdb) break имя_файла:номер_строки
```

Можно также указывать для точки останова определенный адрес.

Следует, однако, учитывать, что не каждой строке текста программы соответствуют операции микропроцессора. Если мы попытаемся создать точку останова в строке, для которой отладчик не может найти исполняемый код, она будет создана на одной из следующих строк, иногда в довольно неожиданном месте. Например, если мы попытаемся создать точку останова в строке 155 в следующем фрагменте:

```
151 if (!strchr(".,:;!?\\" «» %", nxt->vers->Alt[0].Code[0])) {
152     rast->vers->Alt[0].Code[0] = ' ';
153     rast->vers->Alt[0].Code[1] = 0;
154     rast = CSTR_GetPrev(rast);
155     continue;
156 } else {
157     i++ ... }
```

отладчик разместит ее в строке 157. Дело в том, что строке 155 (как и строке 156) нельзя сопоставить машинный код. Первая строка после строки 155, для которой это можно сделать – строка 157, но она находится в другом логическом блоке. Очевидно, что это совсем не то, чего мы хотели. Самое неприятное заключается в том, что отладчик не покажет нам номер строки, на которой на самом деле была создана точка останова, до тех пор, пока эта точка не будет вызвана. Правильным решением может быть создание новой точки останова в строке 154, но в этом случае программа будет приостановлена до вызова функции **CSTR_GetPrev()**. Если нам нужно отследить результат выполнения **CSTR_GetPrev()**,

точку останова следует создавать внутри этой функции (можно, конечно, остановиться до вызова **CSTR_GetPrev()**, а затем пройти по функции серией команд **step**).

Помимо локаций, команда **break** позволяет задать для точки останова

условие. Им может быть любое выражение, возвращающее значение, приводимое к типу **int** и действительное в контексте данного кадра отладки. Вы можете создать несколько точек останова в одной строке (это имеет смысл для условных точек останова).

Если вы не хотите больше прерывать выполнение программы на текущей точке останова, очистите ее с помощью команды **clear**. Любую точку останова можно удалить с помощью команды **delete**, указав в качестве аргумента номер точки останова (он присваивается автоматически при выполнении команды **break**). Если вы заранее уверены, что точка останова понадобится вам только один раз, создайте однократную точку останова с помощью команды **tbreak** (ее синтаксис такой же, как и у **break**).

Предъявите ваши данные

С некоторыми командами, позволяющими заглянуть в содержимое переменных отлаживаемой программы, мы уже знакомы. Когда выполнение программы приостановлено, мы можем просмотреть содержимое любой переменной, доступной в данном контексте. Очень часто в процессе отладки бывает необходимо отследить, как меняется значение некоторой переменной по ходу выполнения программы. Чтобы не вызывать каждый раз команду **print**, мы можем создать контрольную точку доступа к данным (**watch**).

Рассмотрим практический пример. В программе *Cuneiform* есть функция **make_tokens**, которая в ходе своей работы перебирает элементы некоего связанного списка. Указатель на текущий элемент списка хранится в локальной переменной **rast**. В ходе отладки программы мне требовалось посмотреть, как меняются значения **rast**, для чего я решил создать контрольную точку доступа к данным. Такие точки осуществляют мониторинг обращения к определенному адресу памяти на аппаратном уровне. Это означает, между прочим, что контрольную точку доступа нельзя определить до тех пор, пока наблюдаемая переменная не будет создана, то есть, в нашем примере, до тех пор, пока не будет вызвана функция **make_tokens**. Таким образом, прежде чем устанавливать контрольную точку доступа, мне необходимо остановить программу в начале вызова **make_tokens**:

```
(gdb) break make_tokens
```

```
(gdb) c
```

```
Breakpoint 16, make_tokens (line=0x957a108, words=0xbf867b98)
at /home/andrei/bazaar/cuneiform-multilang/cuneiform_
src/Kern/puma/c/spcheck.cpp:140
```

```
140 CSTR_rast rast=CSTR_GetFirstRaster(line);
```

Теперь можно установить контрольную точку доступа к переменной **rast**:

```
(gdb) watch rast
```

```
Hardware watchpoint 17: rast
```

Обратите внимание, что значение **rast** изменяется в той самой строке (140), на которой было приостановлено выполнение программы. Это значит, что уже следующий вызов **next** приведет к изменению содержимого переменной:

```
(gdb) n
```

```
Hardware watchpoint 17: rast
```

```
Old value = (CSTR_rast) 0x0
```

```
New value = (CSTR_rast) 0x957a19c
```

Отладчик сообщает нам, что значение **rast** изменилось с 0x0 на 0x957a19c. Ниже показан дальнейший результат пошагового выполнения функции:

```
142 bool wb = true;
```

```
(gdb) n
```

```
143 int wc = 0;
```

```
(gdb) n
```

```
144 for(rast =
CSTR_GetNext(rast);rast;rast=CSTR_GetNext(rast))
```

```
(gdb) n
```

Назад, в будущее!

В ряде случаев бывает удобным проходить программу не только в прямом порядке, но и в обратном, «оборачивая время вспять». Некоторые ошибки очень сложно вызвать, и если она все же случилась, появляется большое желание развернуть ход выполнения программы в обратном направлении и поглядеть, что же к ней привело. Такой возможно-

стью обладает отладчик *UndoDB* (www.undodb.com). Увы, это не свободное ПО, но если вы работаете над открытым проектом, то сможете получить лицензию бесплатно. Приятнее всего то, что *UndoDB* является надстройкой над *GDB* и просто добавляет новые команды, не изменяя старые, так что вам не придется переучиваться.

Hardware watchpoint 17: rast

Old value = (CSTR_rast) 0x957a19c

New value = (CSTR_rast) 0x95776d0

Мы видим, что при входе в цикл **for** значение **rast** (которая является его управляющей переменной) тоже изменилось. Если бы значение **rast** было случайно перезаписано где-то между двумя явными обращениями к переменной, отладчик сообщил бы нам об этом.

Установленная нами контрольная точка существует столько же, сколько и переменная **rast**, то есть до выхода из функции **make_tokens**: при следующем вызове **make_tokens** ее придется создавать снова. Для удаления контрольных точек доступа служит уже известная нам команда **delete**.

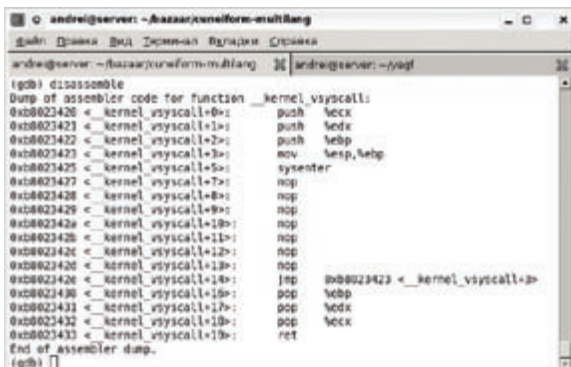
Отслеживать изменение содержимого переменной с помощью комбинации команд **watch** и **step/next** не очень-то удобно. У отладчика GNU есть другой, лучший способ. Команда **awatch** будет останавливать выполнение программы всякий раз, когда значение заданной переменной считывается или записывается. Так же, как и **watch**, команда **awatch** стремится использовать аппаратные ловушки для перехвата обращений к конкретному адресу и может выявлять неявные обращения к нему.

Как я сюда попала?

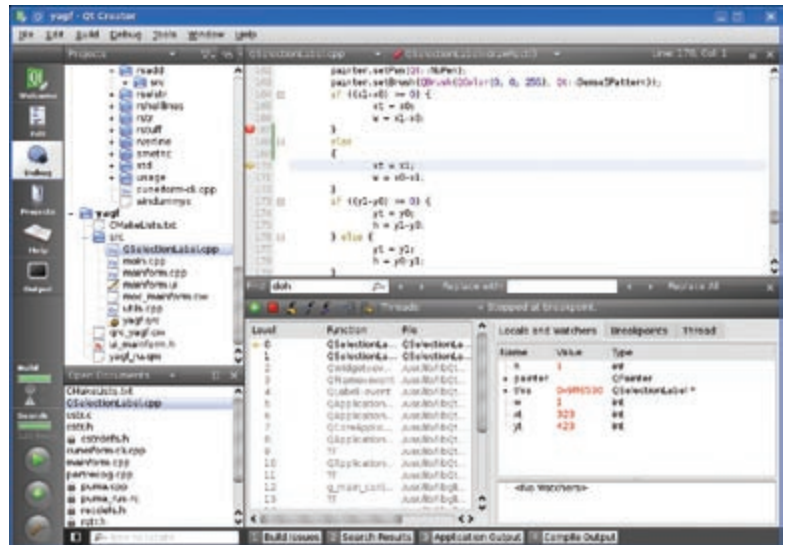
Иногда программисты тоже пытаются найти ответ на этот sacramентальный вопрос. Если в программе во время отладки происходит нечто, требующее вашего внимания, вам может понадобиться не только информация о функции, при вызове которой случилось страшное, но и сведения о полном состоянии стека вызовов. Команда **backtrace** позволяет узнать, какими путями программа попала в данную точку. Посмотрим результат ее выполнения для многоадресной функции **copy_text** (предполагается, что выполнение программы **Cuneiform** приостановлено в ее начале):

```
(gdb) backtrace
```

```
#0 copy_text (word=0x983d348) at
/home/andrei/bazaar/cuneiform-multilang/cuneiform_
src/Kern/puma/c/spcheck.cpp:85
#1 0xb7f213ff in make_tokens (line=0x9789278,
words=0xbff78a88) at /home/andrei/bazaar/cuneiform-
multilang/cuneiform_src/Kern/puma/c/spcheck.cpp:172
#2 0xb7f2158e in mix_lines (ruseng=0x9789278, local=0x97af118,
rus=0x9794578) at /home/andrei/bazaar/cuneiform-
multilang/cuneiform_src/Kern/puma/c/spcheck.cpp:369
#3 0xb7f1eeaf in MultilangRecognizeStringsPass1 () at
/home/andrei/bazaar/cuneiform-multilang/cuneiform_
src/Kern/puma/c/partrecog.cpp:425
#4 0xb7f1fb55 in Recognize () at /home/andrei/bazaar/cuneiform-
multilang/cuneiform_src/Kern/puma/c/partrecog.cpp:798
#5 0xb7f226f7 in PUMA_XFinalRecognition () at
/home/andrei/bazaar/cuneiform-multilang/cuneiform_
src/Kern/puma/main/puma.cpp:600
```



➤ Заглядываем в машинный код. А вы знаете, как выполняются системные вызовы в вашей системе?



➤ Qt Creator (мы говорили о нем в LXF119) — отладка доверена GDB. Обратите внимание, где расположена точка останова и где программа остановилась на самом деле.

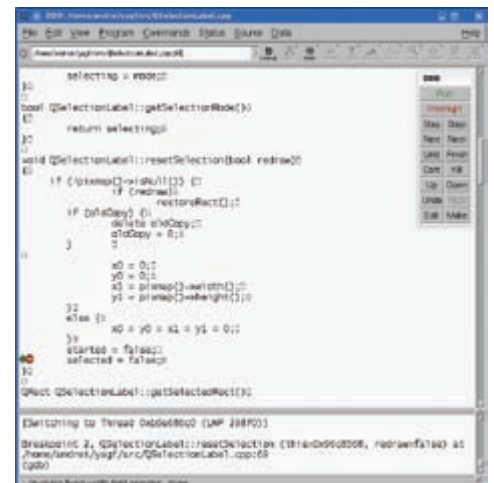
```
#6 0x0804ad18 in ?? ()
#7 0xb7463685 in __libc_start_main () from
/lib/tls/i686/cmov/libc.so.6
#8 0x08049fb1 in ?? ()
```

В переводе на русский язык сказанное отладчиком звучит так: «Функция **PUMA_XFinalRecognition** (), вызванная из функции верхнего уровня, вызвала функцию **Recognize** (), которая вызвала функцию **MultilangRecognizeStringsPass1** (), которая вызвала функцию **mix_lines** (), которая вызвала функцию **make_tokens** (), которая и вызвала функцию **copy_text()**». Но это не все. Благодаря команде **backtrace** мы можем узнать не только адреса функций и значения их параметров (они приведены в скобках), но и локации вызовов в исходных текстах программы (**copy_text** вызывается в нескольких разных местах функции **make_tokens**), и нам, разумеется, полезно знать, из какого именно места она была вызвана на этот раз). Не могу не отметить, что команда **backtrace** помогает не только при отладке кода, но и при изучении работы программы, написанной другими людьми. Фактически у нас перед глазами «живая» цепочка вызовов функций в сложной программе, снабженная значениями их аргументов и другими полезными сведениями.

Графический отладчик DDD

Из всех графических отладчиков, использующих **GDB**, мы рассмотрим только **DDD**. Такая честь выпала ему потому, что этот отладчик наиболее прозрачно интегрирован с **GDB**. Основные команды графической оболочки соответствуют командам **GDB**, а в нижней части главного окна **DDD** мы видим фрагмент консоли **GDB**, в которой отображаются вывод отладчика GNU и его командная строка. Таким образом, если возможностей **DDD** вам не хватает, вы всегда можете обратиться к отладчику **GDB** напрямую. Но главное преимущество **DDD** по сравнению с «чистым» **GDB** — это возможность легко ориентироваться в исходных текстах программы. LXF

➤ Отладчик **DDD** — тонкий слой графики поверх **GDB**.



JavaFX: Добавим

Часть 2: На одном только языке далеко не уедешь. **Антон Черноусов** познакомит вас с объектами, из которых строятся настоящие приложения с графическим интерфейсом.



Наш эксперт

Антон Черноусов
Самый голодный Java-разработчик города Иркутска. Обогащает опасные бритвы.

На предыдущем уроке мы подробно рассмотрели базовые механизмы языка сценариев JavaFX Script, а сегодня сосредоточимся на возможностях JavaFX в области создания графического интерфейса пользователя. После прочтения этой статьи вы разберетесь с объектами-примитивами и применением эффектов, а также, в заключение, узнаете про обработку событий, генерируемых пользователем.

Мы не будем приводить полные исходные тексты учебных примеров, ограничившись лишь интересующими нас частями. Как и в прошлый раз, вы сможете найти все обсуждаемые программы на **LFXDVD**: советую вам прямо сейчас распаковать архив с диска, чтобы было удобнее следить за мыслью. Готовы? Тогда приступим.

Stage — основа GUI

Главный объект, с которого начинается создание пользовательского интерфейса — это **Stage**. Для работы с ним в вашем сценарии JavaFX Script необходимо подключить библиотеку **javafx.stage** с помощью ключевого слова **import** сразу после объявления пакета — например, так:

```
package ru.golodnyj.lxf.fx;  
import javafx.stage.*;
```

» **Рис. 1.** Работа со свойством **opacity** у объекта **Stage**.



Вы можете подключить как весь пакет с помощью символа *****, так и отдельные классы — указанием их канонического имени, скажем, **javafx.stage.Stage**. Следующий простой сценарий (файл **FirstStage.fx** в примерах на диске) создает пустое окно приложения и выводит его в конкретную позицию на экране:

```
Stage {  
    title: "This is Stage"  
    x : 100 // Start point to Stage  
    y : 200  
    width: 300 // The width and height  
    height: 400  
}
```

В этом листинге при создании объекта **Stage** устанавливается заголовок **title**, позиции по горизонтали и вертикали **x**, **y**, а также ширина и высота **width**, **height**.

Однако при таком задании параметров главного окна программы будет невозможно обратиться к его свойствам во время выполнения приложения. Для решения этой проблемы достаточно присвоить объект **Stage** некой переменной, как, например, это сделано в сценарии **OpacityStage.fx**:

```
var s : Stage = Stage {...}
```

API объекта **Stage** достаточно скуден по своим возможностям, но, внимательно изучив его (документацию на английском языке можно найти по адресу <http://java.sun.com/javafx/1/docs/api>), легко обнаружить замечательное свойство **opacity**, которое варьируется в пределах от 0.0 до 1.0 и задает прозрачность окна. Например, значение **opacity**, равное 0.5, создает приятный полупрозрачный эффект, который можно видеть на рис. 1. К сожалению, возможности загрузить API для локального просмотра нет, и мне кажется, что с точки зрения Java-разработчика новый способ представления документации значительно уступает проверенному временем JavaDoc.

При загрузке приложения можно выполнить некоторые подготовительные операции и лишь потом отобразить интерфейс. В этом вам поможет свойство **visible** объекта **Stage**, делающее его видимым или невидимым.

» **Месяц назад** Мы знакомились с синтаксисом JavaFX Script и узнали о связывании.

красок!



Графические примитивы

Безусловно, само по себе пустое окно никому не интересно. Фактически, все компоненты пользовательского интерфейса в JavaFX представляют из себя «узлы» — **Node**, то есть являются некоторым расширением класса `javafx.scene.Node`. Существует несколько типов узлов:

- » примитивы;
- » текст;
- » изображения;
- » медиа-контент;
- » Swing-компоненты.

Узлы размещаются в корневой области, которая называется «сценой» и представлена объектом **Scene**. Объект **Scene** содержит переменную **content**, в которую и помещаются узлы, как, например, в сценарии **FirstScene.fx**:

```
var scene1 : Scene = Scene {
    content: [a, b, c]
}

var s : Stage = Stage {
    scene: scene1
    ...
}

function run(){
    println("Start the App");
    s.visible = true; // make the Stage visible
    s.toFront();
}
```

Результат работы программы можно видеть на рис. 2. В **FirstScene.fx** создаются три графических объекта **a**, **b** и **c**: это прямоугольник (**Rectangle**), окружность (**Circle**) и дуга (**Arc**). Все три наших объекта унаследованы от **Node** и являются графическими примитивами. Для работы с ними необходимо подключить пакет `javafx.scene.shape.*`.

Пакет **shape** содержит всевозможные графические примитивы, каждый из которых имеет очень богатый API. Например, если мы рассмотрим уже созданный нами объект **Arc**, то среди его атрибутов можно найти поле **type**, которое определяет внешний вид примитива и может принимать три различных значения:

- » **ArcType.ROUND** — окружность с центральным углом как на рисунке 2 (по-моему, здесь есть что-то знакомое... да это же Пэкмен!);
- » **ArcType.CHORD** — вместо «скушанного» сектора будет хорда;
- » **ArcType.OPEN** — выглядит как **ArcType.CHORD**, но без линии хорды, соединяющей края центрального угла.

Конечно, создавать отдельные фигуры хорошо, но гораздо удобнее использовать их в совокупности и управлять ими как группой элементов: в JavaFX предусмотрен и такой сценарий. Существует специальный объект **Group**, который является узлом и позволяет хранить в себе другие объекты **Node**. **Group** предоставляет специальный атрибут — последовательность **content**, предназначенный, как и у **Scene**, для хранения **Node**-объектов.

В качестве примера рассмотрим создание несложной картинкой, представленной на рис. 3. Для отрисовки такого изображения необходимо два элемента: это прямоугольник с закругленными краями (**Rectangle**) и текст (**Text**). Несколько забегая вперед, отметим, что текст выводится с помощью объекта **Text**, который также является наследником **Node**.

Как видно из приведенного ниже кода **FirstGroup.fx**, **Node**-объекты перечисляются через запятую, то есть заносятся в последовательность **content**, а доступ к ним можно получить через переменную **d**.

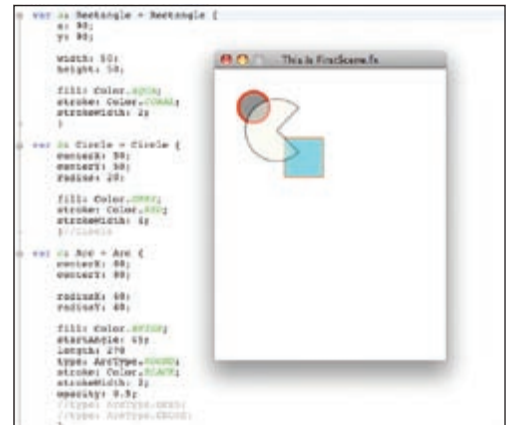
```
var d: Group = Group {
    content: [
        Rectangle {
            x: 30; y: 15; width: 150; height: 20;
            arcWidth: 20; arcHeight: 20; stroke: Color.GRAY;
            fill: LinearGradient {
                startX: 0.0, startY: 0.0, endX: 0.0, endY: 1.0,
                proportional: true
                stops: [
                    Stop {offset: 0.0 color: Color.WHITE},
                    Stop {offset: 1.0 color: Color.GRAY}
                ]
            },
        Text {...}]
}
```

Градиенты

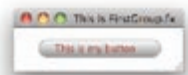
Наибольший интерес в приведенном выше фрагменте представляет свойство **fill** (заливка), а точнее, использованный для нее объект **LinearGradient**, который (кто бы мог подумать?) представляет собой линейный градиент. Переменные объекта **LinearGradient**: **startX**, **startY**, **endX** и **endY** определяют координаты начальной и конечной точек градиента, и таким образом задают его направление и размер охватываемой им области. Градиент можно размещать в любом направлении.

Хитрая переменная **proportional** при установке ее значения в **true** масштабирует переменные **startX**, **startY**, **endX** и **endY** относительно объекта, на который накладывается градиент (при этом значения координат должны находиться в диапазоне от 0,0 до 1,0). При установке **proportional** в **false** величины **startX**, **startY**, **endX** и **endY** имеют смысл абсолютных значений в пикселях.

Последовательность **stops** — это совокупность объектов **Stop** (опорных точек), задающих распределение цветов вдоль градиента. Таких точек может быть больше двух. Поле **offset** объекта **Stop** определяет точку, где градиент будет иметь некий заданный цвет, как число в диапазоне от 0,0 до 1,0 (то есть в долях от протяжен-



» Рис. 2. Различные типы узлов, созданные нами в сценарии **FirstScene.fx**.



» Рис. 3. Наша «почти кнопка» на самом деле собрана из двух примитивов.



► Рис. 4. Варианты радиальной заливки: без повторения (слева), стандартная периодическая (справа) и зеркальная (на переднем плане).

ности градиента). Цвет точек градиента определяется значением атрибута **color**. Кроме определенных цветов, можно задать свой оттенок в стандартной шестнадцатеричной нотации — это делается с помощью метода **web** объекта **Color**, как в примере ниже:

```
stops: [
    Stop {offset: 0.0 color: Color.
    WHITE),
```

```
    Stop {offset: 0.5 color: Color.DARKBLUE},
    Stop {offset: 1.0 color: Color.web («#ffffff»)}
]
```

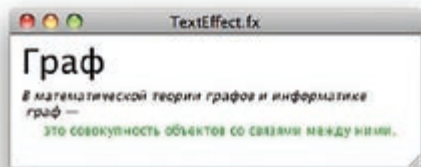
Фактически существует три вида градиентов: **Solid** (сплошная заливка одним цветом), **LinearGradient** (линейный) и **RadialGradient** (радиальный). У последних двух типов есть возможность указать периодичность заливки с помощью свойства **cycleMethod**: без повторения (**CycleMethod.NO_CYCLE**), прямой цикл градиента (**CycleMethod.REPEAT**), прямой-обратный (**CycleMethod.REFLECT**). Какой смысл скрывается за этими значениями, можно видеть на рис. 4.

Работа с текстом

В целом, графических возможностей JavaFX с лихвой хватает для реализации любого способа представления информации, но вернемся к отображению текста. Как уже было отмечено ранее, **Text** — это объект типа **Node**, что позволяет работать с ним так же, как и с другими графическими объектами. Однако работу с его содержимым (то есть собственно текстом — буквами, знаками препинания и так далее) иногда следует отделять от отображения, что демонстрируется в сценарии **TextEffect.fx** и на рис. 5 (определение графа любезно предоставлено Википедией: ru.wikipedia.org):

```
var text = «В математической теории графов и информатике\п
граф —»;
Stage {
    title: "TextEffect.fx" width: 200 height: 230 visible: true
    scene: Scene {
        fill: Color.WHITE
        content: {
            VBox { spacing: 10 content: {
                Text { y: 35 x: 10 content: "Граф" font: Font { size: 30 } },
                Text { y: 15 x: 10 content: text font: Font { oblique: true } },
                Text { y: 5 x: 30 content:
                    "это совокупность объектов со связями
                    между ними." fill: Color.GREEN },
            } }
        }
    }
}
```

Основными атрибутами для объекта **Text** являются его положение **x**, **y** и содержание **content**. С помощью атрибута **font** можно в значительной степени повлиять на внешний вид текста. Объект **Font** определяет характеристики надписи: размер шрифта **size**, начертание **oblique** (наклонное), цвет заливки **fill**, имя гарнитуры **name** и так далее.



► Рис. 5. Расшифровка терминов средствами JavaFX выполняется не хуже, чем в настоящем электронном словаре.

Эффектная трансформация

Помимо прочего, графические объекты JavaFX поддерживают самые различные эффекты и преобразования (трансформации). Что скрывается за этими словами? Ответ на этот вопрос понятен любому, кто работал с графическими редакторами, такими как *GIMP* или *Adobe Photoshop*. Эффект — это визуальное изменение, которое строится на базе текущего представления объекта на сцене. Для задания эффекта у объектов типа **Node** существует специальный атрибут **effect**.

Прежде чем использовать эффекты, необходимо подключить пакет **javafx.scene.effect**. Классическим эффектом является отбрасываемая от объекта тень, словно он является трехмерным и действительно может заслонять собой падающий свет. Тень может быть реализована с помощью объекта **DropShadow** или **InnerShadow**. Например, если вы внимательно просматривали исходные тексты сценария **FirstGroup.fx**, упоминавшегося нами ранее, то могли заметить внутри объекта **Group** пару закомментированных строк:

```
effect: DropShadow { offset X: 5 offset Y: 5 color: Color.DARKGRAY }
effect: InnerShadow { offset X: 5 offset Y: 5 color: Color.DARKGRAY }
effect: Blend {
    topInput: DropShadow {
        offset X: 4 offset Y: 4 radius: 6
        color: Color.DARKGRAY }
    bottomInput: DropShadow {
        offset X: -4 offset Y: 4 radius: 6
        color: Color.DARKGRAY }
}
```

Первая строка накладывает на объект **Group** эффект отбрасываемой наружу тени, в основе которой лежит цвет **Color.DARKGRAY** (темно-серый). Если оставить только вторую строку, то с помощью объекта **InnerShadow** будет создана тень, отбрасываемая внутрь объекта — иллюзия вдавленной под поверхность кнопки. Механизм эффектов реализован очень просто, и JavaFX даже позволяет комбинировать несколько эффектов — как, например, в последнем случае с **Blend**. Визуальный эффект от него интереснее, чем от двух предыдущих — скомпилируйте и запустите скрипт **FirstGroup.fx**, предварительно раскомментировав соответствующие строки.

Эффекты не ограничены только работой с тенями. Есть, например, любопытный эффект **Reflection**, который позволяет создать отражение (ну не могла же Sun Microsystems обойтись без него в JavaFX, ведь этот эффект у всех ассоциируется с Web 2.0!). Можно сделать небезосновательное предположение, что число доступных эффектов в следующих версиях JavaFX будет увеличиваться.

В отличие от эффектов, возможностей преобразования графических объектов с помощью трансформаций наверняка больше не станет, так как уже реализованы следующие:

- перенос (**Translate**);
- вращение (**Rotate**);
- масштабирование (**Scaling**) — пропорциональное изменение размеров;
- деформация (**Shear**) — эффект от применения похож на сдвиг рисунка по осям — например, наклон, как в *наклонном тексте*.

Помимо перечисленных, в JavaFX есть и другие преобразования, но мы сейчас не будем останавливаться на них подробно.

События

Если вернуться к нашей «кнопке», то сразу станет понятна ее неполноценность. Для создания настоящего элемента управления необходимо реализовать обработку событий. На мой взгляд, по сравнению с объектами **Swing** в Java, сделать это в JavaFX значительно проще. У объектов **Node** заранее определено более десятка всевозможных событий, которые обрабатываются в том случае, если написана соответствующая функция.

Для примера рассмотрим скрипт **FirstGroup2.fx**, который является небольшой модификацией **FirstGroup.fx**. Разница заключается в том, что объект **Group** содержит следующий дополнительный код:

```
effect: DropShadow(offsetX:5 offsetY:5 color: Color.DARKGRAY)
onMousePressed: function( e: MouseEvent ):Void {
    e.node.effect = InnerShadow(offsetX:5 offsetY:5 color: Color.DARKGRAY)
}
onMouseReleased: function( e: MouseEvent ):Void {
    e.node.effect = DropShadow(offsetX:5 offsetY:5 color: Color.DARKGRAY)
}
onMouseClicked: function( e: MouseEvent ):Void {
    numOfClick++;
}
```

Кроме того, в теле скрипта определены еще две переменные:

```
var numOfClick = 0;
var a : Text = Text {
    content: bind numOfClick.toString();
    x: 30;
    y: 80;
}
```

Фактически, для нашей кнопки назначен эффект падающей тени и определены функции для обработки трех событий: нажатие клавиши мыши (**onMousePressed** – меняет эффект с тени, падающей наружу, на падающую вовнутрь), отпускание клавиши мыши (**onMouseReleased** – возвращает все как было) и обработка клика (**onMouseClicked** – увеличивает значение переменной **numOfClick** на единицу). Переменная **a** отображает число щелчков, которое было сделано мышью по кнопке. Обратите внимание на связывание (конструкция **bind**) – мы говорили о нем в **LXF119**.

Онлайн-магазины

Оглушительный успех iTunes App Store как магазина приложений заставил крупных поставщиков взглянуть на это направление как на новую возможность распространения своих продуктов и удержания клиентов.

Еще в мае был открыт магазин приложений Ovi (<https://store.ovi.com/>), разработанный Nokia. Доступ к ресурсу осуществляется прямо с мобильных телефонов Nokia. Так же, как и iTunes Store, сервис от финской компании позволяет загружать приложения, игры, видеоролики, подкасты.

Отметим, что сервис Ovi был русифицирован и стал доступен российским пользователям без задержек, а наиболее активные

владельцы телефонов Nokia уже попробовали его в действии.

Незадолго до открытия сервиса Ovi Джонатан Шварц [Jonathan Schwartz], исполнительный директор компании Sun Microsystems, анонсировал запуск Java App Store: видеообращение по этому поводу можно посмотреть в официальном блоге http://blogs.sun.com/jonathan/entry/will_java_be_the_world. Сервис будет предоставлять возможность распространения ПО, написанного с помощью Java и JavaFX, как для мобильной платформы, так и для обыкновенных ПК. Официальное открытие магазина состоялось на конференции JavaOne 2009.

И напоследок...

На данном уроке мы познакомились с некоторыми возможностями JavaFX по части пользовательского интерфейса: создали графический объект и реализовали обработку событий для него. Этого вполне достаточно для несложных приложений, но в более серьезных случаях бывает необходимо применять компоненты Swing и Java-классы. Я решил оставить этот вопрос вам на самостоятельное изучение: в конце концов, если вы уже знаете Swing или Apache POI, то разобраться с тем, как вызвать их из JavaFX Script, не составит для вас труда. Основа заложена – дополнительную информацию можно почерпнуть из документации и других ресурсов, перечисленных в прошлом номере. Если ваша программа выиграет приз в JavaFX Galaxy, дайте нам знать!

SunSPOT и JavaFX

На конференции JavaOne, проходившей с 1 по 5 июня в Сан-Франциско (США), технологии JavaFX уделялось повышенное внимание. Среди участников конференции у нас имелся свой агент – **Денис Магда**, Sun Campus Ambassador из города Благовещенска Амурской области. Мы с радостью предоставляем вам его отчет о мероприятии.

Linux Format (LXF): Денис, скажите, как просто-му студенту удалось побывать на JavaOne 2009? Кого для этого нужно убить?

Денис Магда (ДМ): Убивать никого не пришлось [смеется]. Все дело в том, что мое увлечение технологиями Sun и реализация небольшого приспособления с использованием SunSPOT, Java и JavaFX заинтересовало боссов Sun Microsystems.

LXF: Вы имеете в виду, сенсорно... выговорить не могу?

ДМ: Да, да. Сенсорно-моторный интерфейс управления, по сути – это перчатки с кнопками и контроллерами SunSPOT. SunSPOT'ы позволяют определить изменение положений рук с помощью встроенных датчиков (акселерометров) и обеспечить беспроводную передачу данных на ПК. Кстати, управление контроллером осуществляется с помощью программы, написанной на Java, и приложение, принимающее данные на стороне ПК, тоже написано на Java. Весь код доступен по лицензии GPL на сайте проекта <https://sensormotor-gloves.dev.java.net/>. Там же можно посмотреть видеодемонстрацию.

LXF: А каким образом используется в вашем проекте JavaFX?

ДМ: О, я думал, как сделать красочную демонстрацию своего устройства, и написал демо на JavaFX. На сцене размещены несколько объектов с мультимедийным содержанием (картинки и видео), и с помощью перчаток можно выполнять различные операции: запускать ролики, увеличивать или уменьшать объекты, вращать их и так далее. Но для JavaOne пришлось создать гораздо более интересную демонстрацию.

LXF: Отлично. И, конечно, ее исходные тексты тоже можно загрузить с сайта проекта?

ДМ: Нет, боюсь, их там пока нет, но как только я освобожусь, думаю, выложу обе «демки»: и старую, и новую. Пока нет и официального видео с JavaOne, но можно посмотреть любительские съемки, например, здесь: http://www.youtube.com/watch?v=QKkHjSO_cAo. Вообще, JavaFX на JavaOne было уделено огромное внимание и среди стендов, и среди выступающих. В числе новинок – релиз JavaFX 1.2 и специальная среда разработки, причем это не NetBeans, и еще многое, о чем просто нельзя рассказать – это надо видеть.

LXF: Читателей нашего журнала волнует такой вопрос: когда же будет версия JavaFX под Linux?

ДМ: Скажу коротко, я ее уже видел... и вы тоже можете. Просто зайдите по адресу <http://javafx.com/downloads/all.jsp>: отсюда можно загрузить JavaFX 1.2 для Linux, Solaris, Mac OS X и Windows.



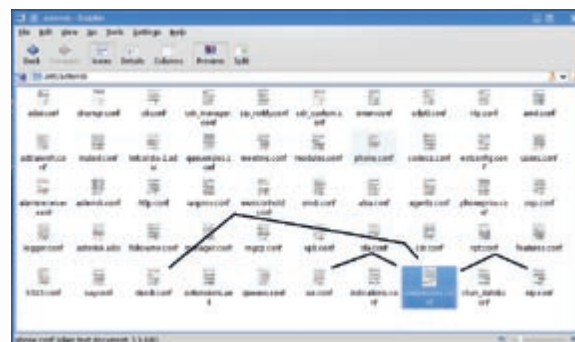
➤ Выступление Дениса Магды на JavaOne 2009 имело заслуженный успех.

LXF: Спасибо вам, Денис, за уделенное время и приятные новости!

ДМ: И вам тоже спасибо. **LXF**

Asterisk: Телефон

Нырните с **Максом Бруксом** в мир телефонии с открытым кодом, *Asterisk* и стаю орущих мартышек.



» Соединение файлов в схему набора: похоже на паука, но совсем не кусается.

Наш эксперт

Макс Брукс

давний фанат Slackware, пользователь Gentoo и профессиональный разработчик LAMP. Он создал VoIP-сеть на основе Debian для маленькой конторы из 20 работников.

Кто из нас не желал бы послать в ответ на рекламный обзвон визг стаи мартышек, или эхом повторить то, что вам сказали? Симпатична ли вам идея, чтобы каждый член вашей семьи имел личный добавочный номер и голосовую почту? А может, вы предпочитаете создать белый список людей, чьи звонки вас осчастливят даже после 10 вечера?

Asterisk метко назвали звездой Open Source в быстро расширяющейся вселенной VoIP-коммуникаций. До мелочей настраиваемая и порой соответственно сложная, это также идеальная система для экспериментов, и она умеет делать все вышеперечисленное. Этот учебник безболезненно проведет вас через создание собственной домашней PBX VoIP [мини-АТС, — прим. пер.] с *Asterisk*. Начав с простой базовой системы, вы получите выбор, какие функции нужно будет добавить.

Приступим

Для запуска *Asterisk* прежде всего необходим компьютер — к счастью, очень скромный на счет ресурсов: сойдет любое старье с чердака (или, как в моем случае, из-под кровати); а не то установите его на ваш медиа-сервер Linux, притаившийся в прихожей. Правда, для подключения вашей мини-АТС к имеющейся телефонной линии при установке

на медиа-сервер может потребоваться полноразмерная PCI-карта, но это единственная неприятность. Если вы не собираетесь этого делать, медиа-центр — идеальный выбор.

Вариантов установки *Asterisk* несколько; вы можете развернуть специально созданный дистрибутив, типа *Asterisk@Home*, или просто установить пакет для вашей ОС. Я буду исходить из того, что вы добавляете пакеты на свой медиа-сервер и не хотите переустанавливать Linux, так как только недавно заставили все красиво работать. В Fedora 10 (который я использую здесь), этот трюк проделает простое

```
sudo yum install asterisk
```

Закливание для Ubuntu —

```
sudo apt-get install asterisk asterisk-sounds-main asterisk-sounds-extra
```

установит все необходимые файлы, включая звуковые. Если вы используете графическую утилиту установки, не помешает добавить все доступные пакеты для *Asterisk*.

Базовая система

Итак, куда мы двинемся? Сперва создадим простую конфигурацию *Asterisk*, пригодную в качестве основы для последующего обвешивания нужными вам функциями. Эта базовая схема включает всего один SIP-телефон, аппаратный либо программный, и вы сможете звонить по нему через Интернет при помощи VoIP-провайдера.

Провайдеров VoIP на свете хватает — и хороших, и плохих, и совсем никудышных. Хорошие дадут вам возможность использовать IAX, родной VoIP-протокол *Asterisk*, который лучше работает

«Почти все настройки *Asterisk* следуют тем же простым правилам.»

» **Месяц назад** Мы ловили пожирателей трафика с помощью хитрого *Trickle*.

ПОД КОНТРОЛЕМ



в NAT-окружении — с ним вы можете столкнуться, если занимаетесь этим дома. Через прокси, протокол IAX также поддерживает другие решения с открытым исходным кодом. Поддержка буфера колебаний задержки в IAX2 также более зрелая, чем в SIP-версии. Если ваш VoIP-провайдер не позволяет использовать IAX, или если качество линии плохое, я бы порекомендовал связаться с ближайшим LUG, чтобы разузнать про что-нибудь приличное поблизости. Чем меньше будут хаять, тем лучше.

Для вашей системы Asterisk можно применить чисто программный SIP-телефон, если в компьютере есть микрофон и динамик. Можно приобрести для этого специальную гарнитуру, но будет гораздо лучше вложиться в покупку сугубо аппаратного SIP-телефона — например, одной из базовых моделей Linksys или Grandstream: надо ли вам, чтобы компьютер постоянно занимался приемом и передачей звонков? Тем более, если у вас не по компьютеру на каждого человека, желающего поговорить. В более дорогих моделях аппаратных SIP-телефонов есть функции, которые нужны только в офисе, например, поддержка нескольких линий, поэтому хорошенько подумайте, прежде чем тратить лишние деньги.

Система настройки Asterisk довольно проста: файлы конфигурации сохраняются в `/etc/asterisk`, и для каждого протокола и схемы набора (dialplan) есть отдельный конфигурационный файл. Самые важные из них — `extensions.conf` и `extensions.ael`. В них вы определяете схему набора — иными словами, помещаете туда то, что должно происходить, когда звонок идет в систему или из нее. Различаются эти файлы синтаксисом. Расширение `.ael` — сокращение от Asterisk Extension Language, этот язык допускает более глубокую настройку и легче читается, чем сложный файл `.conf`. Оба файла при необходимости могут быть использованы вместе. Asterisk устанавливается вместе с настройками по умолчанию и множеством полезных комментариев в файлах конфигурации, и это первый пункт, где стоит искать подсказку, если вы застряли.

Пляшем с телефоном

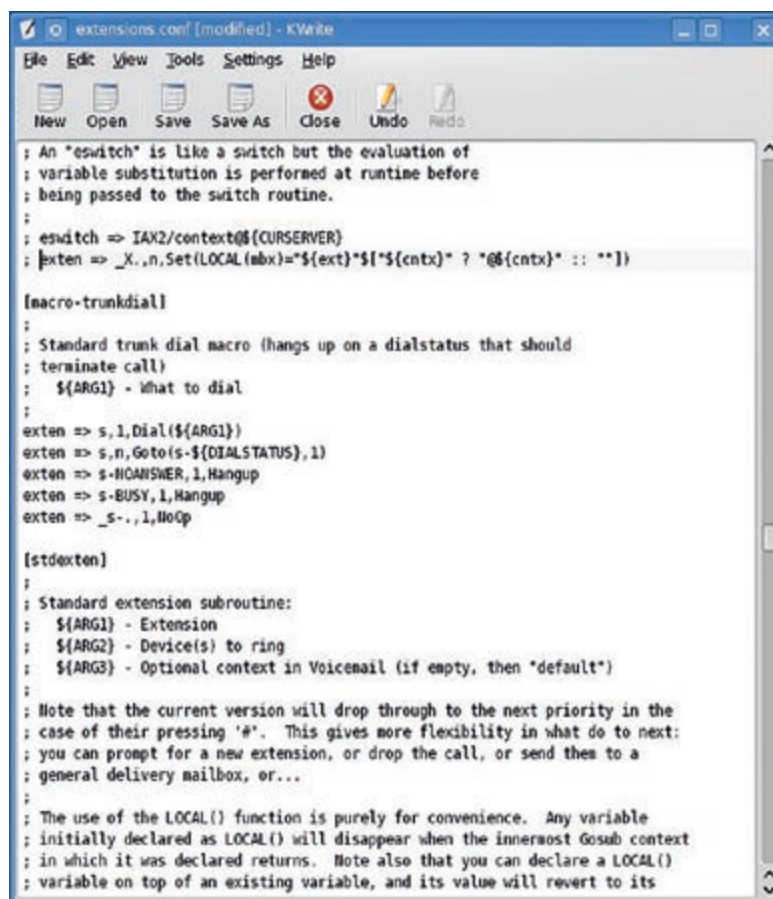
Прежде чем окончательно погрязнуть, разберем несколько общих вещей в файлах конфигурации: ; обозначает комментарий; опции, определяемые в начале файла, применяются ко всем нижележащим частям, пока не встретится явная перезапись. Звучит сложно, но на практике имеет немалый смысл.

Чтобы сообщить Asterisk о вашем SIP-телефоне, добавьте запись в `/etc/asterisk/sip.conf`. Вы должны сделать это от имени root, так как по умолчанию файлы Asterisk принадлежат суперпользователю. Предлагаю воспользоваться простым редактором командной строки, типа *Nano*, но, конечно, если вы исповедуете религию *Vi* или *Emacs*, изменять им незачем.

```
$ su -
```

```
$ nano /etc/asterisk/sip.conf
```

Прокрутив страницу вниз и увидев кучу доступных опций, можно прийти в недоумение и даже слегка струхнуть; но не бойтесь. Упомянутые пункты довольно просты и легко понимаемы,



➤ На первый взгляд — жуть, но если присмотреться поближе, то все проясняется.

что и ценно, потому что почти все файлы конфигурации Asterisk следуют тем же простым правилам.

[general]

Это отметка начала раздела, который в файле конфигурации Asterisk называется контекстом. Здесь может быть несколько опций, таких как

context=default ; Контекст по умолчанию для входящих звонков

bindport=5060 ; UDP-порт для связи (стандартный порт SIP для незашифрованных сессий UDP
; и TCP это 5060)

; bindport это локальный UDP порт, который Asterisk будет слушать

Смысл темен? Не беда — позже все прояснится. Существуют два слегка различных синтаксиса добавления записей в `sip.conf`. Первый использует шаблоны и гораздо больше подходит для систем с большим количеством записей `sip.conf`, поэтому

мы его пропустим и перейдем к способу попроще. Вот пример записи в **sip.conf**:

```
[myhouse]
type=friend ; Может быть "friend" (peer+user), "peer"
или "user"
context=incoming
host=dynamic ; Вы можете задать здесь IP-адрес
username=username ; Имя пользователя на устройстве SIP
secret=blah ; пароль на устройстве SIP
nat=no ; Нет NAT между телефоном и Asterisk
canreinvite=yes ; разрешить передачу трафика RTP, ми-
нуя Asterisk
```

Можно скопировать и вставить данный пример в конец (но не в начало – при создании конфигурации в *Asterisk* это имеет значение) файла **sip.conf**, просто изменив переменные **username** и **secret** на действительные, и этого должно быть достаточно.

Приведенный пример должен быть понятен сам по себе или по крайней мере с комментариями: по существу, вы задали SIP-устройство с параметрами его конфигурации. При желании вы могли бы изменить переменную **context** в **[general]** на

```
context=incoming
```

а затем удалить эту строку из вышеуказанной конфигурации. Запишите ваши изменения в файл и закройте его.

Надеемся, вы нашли себе VoIP-провайдера, поддерживающего протокол IAX; если нет, следуйте инструкциям, но редактируйте файл **sip.conf**, добавляя в него сведения о SIP-услугах. Пример записи в **iax.conf** выглядит так:

```
[provider] ; Контекст с именем provider
type=peer
username=userid
secret=blah
host=iax.provider-example.net ; имя хоста вашего постав-
щика услуг
```

Это напоминает синтаксис **sip.conf** (на самом деле, почти все файлы конфигурации *Asterisk* выглядят похоже), где основным дополнением будет опция **host**, которую должен предоставить поставщик услуг. Отредактируйте и сохраните файл, как и раньше.

Совершаем звонок

Возможно, вы заметили, что мы фактически не сказали *Asterisk*, что делать, получив вызов, или как установить соединение через свеженастроенного VoIP-провайдера. Как я уже намекал, это волшебство происходит в файле **/etc/extensions.conf** за счет использования так называемых приложений схемы набора. Мы сей-

час рассмотрим очень простой пример, и позднее вы сможете добавить некоторые интересные функции. Нам нужны два основных контекста в **extensions.conf**:

```
[outgoing]
exten => _0[1-9].1,Dial(IAX2/userid@
provider/44${EXTEN:1})
exten => _00.1,Dial(IAX2/userid@provider/${EXTEN:2})

[incoming]
exten => s,1,Answer()
exten => s,n,Dial(SIP/myhouse)
exten => s,n,Hangup()
```

Если у вас есть опыт программирования, то вышеприведенный текст будет вам более понятен. Основной шаблон для определения расширения выглядит следующим образом:

```
exten => name,priority,application()
```

Во входящих расширениях существует специальное имя, **s** (обозначает старт), соответствующее всем номерам, направляемым в контекст, и особый приоритет, **n**, который принимает предыдущий приоритет плюс один и сбрасывается в каждом новом расширении. Не так уж сложно разобраться, что входящее расширение отвечает на все направленные на него вызовы, затем набирает SIP-телефон, настроенный в контексте **myhouse**, а после завершения вызова вешает трубку.

Неплохо будет добавить в вашу схему набора отладочный номер, чтобы проверить, работает ли она хотя бы частично. Звуковой файл **ss-weasels** находится в пакете **asterisk-sound-extra** в вашей соответствующей установке.

```
exten => 8888,1,Answer()
exten => 8888,n,Playback(ss-weasels)
exten => 8888,n,Hangup()
```

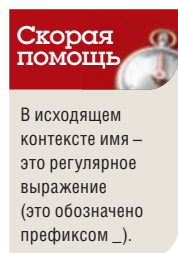
Подключение к линии

Подключение вашего компьютера с новехоньким *Asterisk* к имеющейся телефонной линии, возможно, единственная сложная операция с устройством *Asterisk*. У нас недостаточно места, чтобы описать все детали, но мы включили ссылки на дополнительную информацию в конце этого учебника. Для подключения существующей телефонной линии к вашему компьютеру с *Asterisk* вам необходимы конкретные аппаратные средства. Такую аппаратуру делает целый ряд компаний, и наиболее широко известна из них Digium, которая финансирует развитие *Asterisk*, так что работа их карт с *Asterisk* гарантируется. Но стоят они дорого, и такая цена может показаться вам неоправданно высокой, потому что существуют решения типа Atcom AX-100P, который раз в двадцать дешевле.

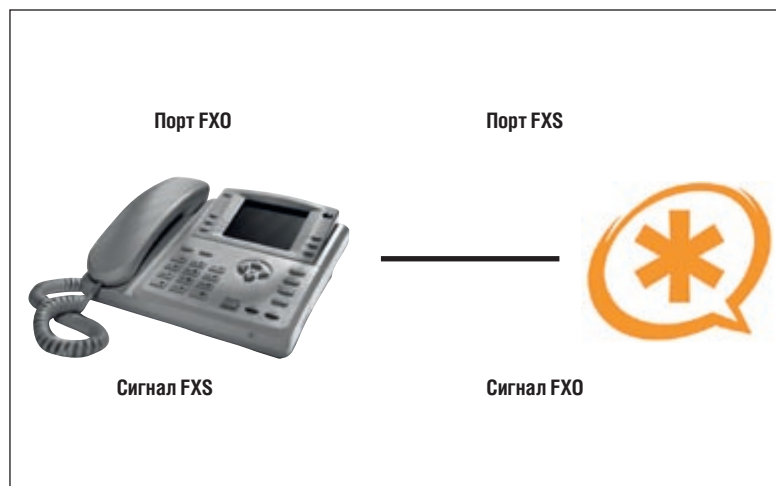
Основная сложность этой задачи зависит от видов используемой связи. На вашей карте PSTN можно найти два типа аппаратных портов или устройств: FXO и FXS. Устройство FXS подключается к порту FXO на существующем аналоговом телефоне и использует связь FXO. И наоборот, устройство FXO подключается к устройству FXS или розеткам, а использует связь FXS.

Вплоть до версии *Asterisk* 1.6 драйвер ядра для устройств Digium назывался Zaptel, а затем по соображениям авторских прав его переименовали на Dahdi. Так как Dahdi – всего лишь название, то при возникновении трудностей заодно поищите в Google насчет проблем, связанных с Zaptel.

Чтобы использовать и установить Dahdi, надо будет установить драйвер (модуль ядра) для вашей карты. Правильный модуль ядра для карты Digium найдется в списке на <http://docs.tzafir.org.il/dahdi-linux>. Затем понадобится настроить использование правильных способов связи, как описано выше. Полезную информацию для этого можно найти в **/proc/dahdi** и использовать ее для редактирования файлов конфигурации в **/etc/dahdi**. Используйте инструмент **dahdi_test** для проверки вашей конфигурации.



В исходящем контексте имя – это регулярное выражение (это обозначено префиксом _).



➤ Это противно интуиции, но устройство FXO использует связь FXS, и наоборот.

```
max : bash <2>
File Edit View Scrollback Bookmarks Settings Help
root@darkEarth:/home/max #echo "200 => 1234,Your Name,email@address" >> /etc/asterisk/voicemail.conf
root@darkEarth:/home/max #
```

Добавляем голосовую почту

Добавление голосовой почты – штука несложная; нужно изменить три конфигурационных файла. С двумя первыми все просто до изумления, а третий чуть сложнее. Файл конфигурации голосовой почты находится в стандартном `/etc/asterisk/`, и предсказуемо называется **voicemail.conf**. Прежде всего, укажите номер почтового ящика в **sip.conf**:

```
mailbox=1234@default ; ящик 1234 В контексте голосовой почты "default"
```

Затем нужно задать почтовый ящик в **voicemail.conf**, поэтому запустите следующую команду в терминале *Bash* от root:

```
echo "200 => 1234,Your Name,email@address" >> /etc/asterisk/voicemail.conf
```

Если у вас несколько телефонов, то умно будет обозначить почтовые ящики пользователей их же внутренними добавочными номерами, а затем использовать эти номера для переключения голосовой почты динамически в схеме набора, вместо того, чтобы каждый раз прописывать подобные строки.

```
exten => 777,1,VoiceMailMain()
exten => 777,n,Hangup
; Набор 777 позволит вам прослушать свои голосовые сообщения
[macro-voicemail]
exten => s,2,Goto(s-$DIALSTATUS),1)
exten => s-NOANSWER,1,VoiceMail(u${MACRO_EXTEN})
exten => s-NOANSWER,2,Hangup()
exten => s-BUSY,1,VoiceMail(b${MACRO_EXTEN})
exten => s-BUSY,2,Hangup()
exten => _s-.,1,Goto(s-NOANSWER,1)
```

Мартишки против обзвонов

А сейчас – повеселимся: создадим «черный» список, запретив кое-кому доставать вас звонками. На словах не особенно впечатляюще, но это полезная вещь. Лица из черного списка будут перенаправляться на звуки, издаваемые стаей обезьян, а телефон при этом звонить не будет. Признаюсь, идею про орущих обезьян я придумал не сам: этот звуковой файл поставляется с *Asterisk* в дополнительном звуковом пакете. Соответствующие строки были добавлены в текст ниже:

```
[incoming]
exten => s,1,LookupBlacklist(j)
exten => s,n,Dial(SIP/myhouse)
exten => s,n,Hangup()
exten => s,102,Goto(blacklisted,s,1)
[blacklisted]
exten => s,1,Answer()
```

```
exten => s,n,Playback(tt-monkeys)
exten => s,n,Hangup()
```

В *Asterisk* нет встроенного приложения для «белых» списков, в отличие от «черных». Это более расплывчатая концепция. Мы используем «белый» список людей, которые могут звонить вам в определенное время, например, только с мобильных телефонов детей после 10 вечера. Предположим, номеру 01225 888888 разрешается звонить в любое время:

```
[incoming]
exten => s,1,GotoIfTime(9:00-22:00|mon-fril*!)?whitelist,s,1)
exten => s,2,Answer()
exten => s,n,Dial(SIP/myhouse)
exten => s,n,Hangup()
[whitelist]
exten => s,1,GotoIf("${CALLERID(num)}" = "01225888888")?incoming,s,2)
```

Советы, секреты и починка

Я покамест не охватил параметры командной строки *Asterisk*. Для запуска командной строки *Asterisk* выполните

```
asterisk -c
```

для возобновления связи –

```
asterisk -r
```

Остановка *Asterisk* отличается от выхода из оболочки, поэтому здесь необходима команда возобновления. Ввод

```
asterisk> stop now
```

в командной строке *Asterisk* остановит сервер *Asterisk*, а команда *exit* просто выведет вас из оболочки.

Вас, вероятно, интересует, зачем вообще *Asterisk* командная строка. Причина номер один – поиск неисправностей и развитие. Вы можете собрать из командной строки важную информацию о текущем состоянии вызова с номера, а также обновить файлы конфигурации без перезапуска *Asterisk*, набрав

```
asterisk> module reload
```

Поздравляем, теперь у вас полная власть над своим телефоном! **LXF**

» В тихом омуте черти водятся!

Скорая помощь

Символ >> добавляет вывод команды *echo* к указанному после нее файлу, символ > заменит или создаст файл.

Что дальше?

Asterisk – чрезвычайно сложная и настраиваемая система. У нас просто не хватит здесь места, чтобы рассказать вам об *Asterisk* все: на это потребуется целая книга, и на самом деле такая уже написана. Она называется *Asterisk: the Future of Telephony* [*Asterisk: будущее телефонии*], издана O'Reilly и имеет версию в PDF-формате, доступную на www.asteriskdocs.org.

Еще одна важная ссылка (я нашел ее при разработке системы *Asterisk*) – это вики на www.voip-info.org/wiki-Asterisk.

» **Через месяц** Как настроить web-сервер на максимум безопасности.

ОТВЕТЫ

Есть вопрос по Open Source? Пишите нам по адресу answers@linuxformat.ru!

» В этом месяце мы отвечаем на вопросы про:

- | | |
|-----------------------------------|------------------------------|
| 1 Обновление пользователей Ubuntu | 6 Базы паролей |
| 2 Устранение паники ядра | 7 Настройку дисплея |
| 3 Subversion | 8 Battle for Wesnoth |
| 4 LIRC | 9 Установку Linux на нетбуки |
| 5 Защиту данных | ★ UNR |

1 Верните мои записи!

В Раньше у меня на /dev/sda1 был установлен Ubuntu 8.10, а в /home на /dev/sda2 было четыре учетных записи пользователей. Затем я решил скачать Ubuntu 9.04 и заменить 8.10 на /dev/sda1. Когда установка закончилась, я хотел восстановить три другие учетные записи, но Ubuntu сообщил мне, что они (их домашние папки) уже существуют, и их нельзя добавить снова. Не подскажите ли, как восстановить эти три учетные записи?

Кай Сиум [Cae Sium]

О Тут проявился норов инструмента управления пользователями Ubuntu: он отказывается создавать учетную запись, если домашняя директория с таким именем уже существует. Можно переименовать старую дирек-

торию, пересоздать пользователя, изменить идентификаторы пользователя и группы (UID и GID), удалить директорию нового пользователя, а затем вернуть прежнюю на место – но при использовании якобы «облегчающего труд» графического интерфейса это адски много работы, причем придется повторять все тот же нудный процесс для каждого пользователя.

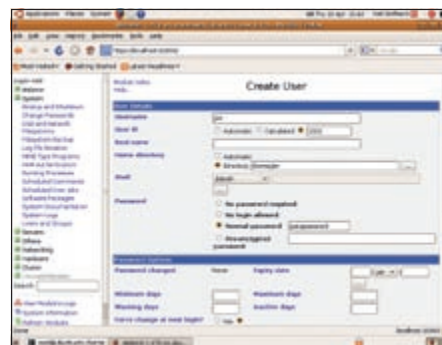
Гораздо легче сделать это с помощью инструмента командной строки *useradd*, который не создает пользовательскую директорию по умолчанию. Во-первых, найдите старые UID и GID Ваших пользователей – это можно сделать с помощью *ls*. Попробуйте выполнить следующее:

```
sudo ls -l /home
```

В результате должно получиться нечто вроде

```
drwxr-xr-x 2 1001 1001 4096 2009-04-16 11:07 fred
drwxr-xr-x 2 1002 1002 4096 2009-04-16 15:23 jim
drwxr-xr-x 30 nelz nelz 4096 2009-04-16 11:02 nelz
```

Третий и четвертый параметры содержат UID и GID. Вы заметите, что для пользователя с именем *nelz* вывод покажет имена пользователя и группы, а для двух других – пользователей, которые еще не созданы – числовые значения, приписанные им. Теперь пересоздайте этих пользователей командой *useradd* – например, так:



» Вездесущий Webmin создает и администрирует пользователей в любом дистрибутиве.

```
sudo useradd --home /home/fred --uid 1001
--user-group fred
sudo passwd fred
```

Создается пользователь *fred* с домашней директорией */home/fred* и UID 1001, а затем будет запрошен пароль для него. Ключ *--user-group* создаст индивидуальную группу для этого пользователя, согласно правилу Ubuntu по умолчанию. Если Вы хотите, чтобы все пользователи являлись членами одной и той же группы (например, *users*), используйте команду

```
sudo useradd --home /home/fred --uid 1001
--gid users fred
```

Повторите это для каждой учетной записи, а затем проверьте с помощью *ls -l*, что все домашние директории принадлежат правильным группам и пользователям. Если это не так, Вы можете исправить ошибку:

```
chown -R fred: ~fred
```

Хотя этот процесс решит Вашу текущую проблему, нет гарантии, что такое же не произойдет при очередном обновлении. Но это не повод отказываться от переходов на новые версии дистрибутива: лучше сделать копию */etc/passwd*. Тогда при следующем обновлении Вы можете просто скопировать соответствующие строки из этого файла на свои места в новой установке, хотя пароли задать все равно придется.

Альтернативный инструмент для выполнения этой задачи – утилита *Webmin*. Ее нет в стандартных репозиториях Ubuntu, но Deb-пакет можно загрузить и установить с www.webmin.com/download.html. Затем откройте в браузере <https://localhost:10000> и примите сертификат безопасности (это Ваша собственная программа, работающая локально, так что сертификат действительный, что бы ни говорил браузер); и, наконец перейдите в Система > Пользователи и группы, для настройки Ваших пользователей. **ПХ**

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.



Владимир Житомирский

Председатель Совета директоров Правового бизнес-бюро «Граф Маевский» и убежденный линуксоид; помогает нам в качестве юриста..



Майк Сондерс

Майк был одним из создателей прототипа LXF – Linux Answers. Его специальности – программирование, оконные менеджеры, скрипты инициализации и SNES.



Грэм Моррисон

Когда он не обзаведает кучи программного обеспечения и не халтурит с MythTV, Грэм готов дать ответ касательно любого оборудования и проблем виртуализации.



Юлия Дронова

Когда ее компьютер не занят выполнением команды *emerge*, Юлия использует его, чтобы модерировать ЛинуксФорум.

Куда посылать вопросы

Пишите нам по адресу: answers@linuxformat.ru или спрашивайте на форуме: www.linuxformat.ru

2 Без паники!

В Мой монитор Samsung 2343BWX не работает с Ubuntu 8.10. При загрузке с Live CD процесс останавливается, и я получаю пустой экран, а два из трех индикаторов в верхней части клавиатуры начинают мигать. Далее ничего не происходит. Мигает тот индикатор, что с буквой А, и тот, что с символом стрелки. Сперва я думал, что проблема в видеокарте, и купил Nvidia GeForce 6200, но смена адаптера не помогла. Потом я попытался проверить монитор на других моих компьютерах (на них стоит OpenSUSE 11.1, Mandriva 2009 и Fedora 10), но там он работал. Этот монитор также работает при загрузке Ubuntu 8.04. Как вы думаете, в чем тут проблема?

Джим Беккер [Jim Becker]

О Весьма маловероятно, что проблема именно с монитором, Джим. Можно предположить, что Вы увидите те же симптомы, даже если замените его. Ключ к решению — мигание светодиодов Caps Lock и Scroll Lock клавиатуры, указывающее на «панику ядра» [kernel panic]. Такое бывает, когда ядро сталкивается с ошибкой, которую ему не одолеть — это похоже на «голубой экран смерти», присущий некой проприетарной ОС, но обычно более информативно. А именно: встретив неразрешимую ошибку, ядро печатает отладочную информацию на экран и останавливает работу. Мигающие светодиоды на клавиатуре являются дополнительным индикатором в случае, когда нельзя увидеть экранный вывод, как в Вашей ситуации.

Сообщения об ошибке не видно потому, что Ubuntu, как и большинство других дистрибутивов, скрывает его за заставкой. Для удаления заставки и просмотра сообщения об ошибке, запустите компьютер с CD и нажмите F6, чтобы в меню появилась строка с загрузочными параметрами. Уберите из нее слова **quiet** и **splash** и нажмите Enter для продолжения загрузки. Ваш компьютер будет по-прежнему виснуть из-за паники ядра, но на этот раз Вы увидите, что оно имеет Вам сообщить. Вставьте эту информацию в любимую поисковую систему. Если ошибка не связана с каким-нибудь не в меру загадочным оборудованием, кто-нибудь, скорее всего, уже имел дело с этой проблемой и опубликовал ее решение.

Хотя, прежде чем впрягаться поисковую систему, стоит попробовать несколько стандартных вещей. Во-первых, отключите все лишние USB-устройства. Вполне очевидно, что без клавиатуры и мыши не обойтись, но любые сканеры, принтеры, аудиокарты и внешние устройства хранения данных могут отдохнуть. Это не означает, что Вы не сможете пользоваться ими: просто проблемы ядра на Live CD могли быть из-за них. Установив систему, Вы сможете опять подключить свои устройства.

Еще одна рядовая проблема — неверная реализация APIC (расширенный программируемый контроллер прерываний) на Вашей материнской плате. Видите ли, некоторые изготовители охотно работают по минимуму, обеспечивая лишь работу в Windows, а не выполняя требования спецификации. Этот вопрос решается путем отключения

поддержки APIC ядром. Нажмите F6, чтобы получить загрузочные параметры, как описано ранее, удалите **quiet** и **splash**, чтобы видеть происходящее, и замените их на **noapic**. Если это сработает, можно добавить опцию **noapic** в загрузочное меню во время установки, чтобы использовать ее всегда, но заодно проверьте, нет ли обновления BIOS для Вашей материнской платы. Даже у новых материнских плат бывают обновления, появившиеся за время их путешествия к прилавку, и они нередко исправляют проблемы с APIC. **НБ**

3 Вдвойне subversивно

В С интересом прочел вашу недавнюю статью про Subversion (LXF118). У меня есть старая машина с Ubuntu 8.04 Server, а на ней рабочая установка Subversion. В настоящее время она имеет один репозиторий в /svn, но я затеял эксперимент с наличием двух отдельных репозиториях с общей родительской директорией.

Я создал репозитории с помощью svnadmin, и они вроде отлично работают, если я использую svn checkout file:///полный/путь/до/репо/, но если я использую svn://относительный/путь/к/checkout, то я должен сообщать svnserve, где находится корень двух репозиториях. Поиск в Интернете, я обнаружил, что могу передать корень в svnserve как параметр — а как это сделать при автоматическом запуске svnserve при загрузке системы?

Энди Мьюз [Andy Muse]

О Оба репозитория находятся в одной и той же директории? Тогда используйте с svnserve опцию --root и включите имена репозиториях в запрос. Например, если Вы используете /полный/путь/к/репо1 и /полный/путь/к/репо2, запустите svnserve так:

```
svnserve --root /полный/путь/к/
```

Затем обращайтесь к ним как svn://репо1/путь/к/файлу или svn://репо2/путь/к/другому/файлу.

Альтернатива — запуск отдельного сервера для каждого репозитория, на разных портах или именах хостов, чтобы их не путать. Например, команды

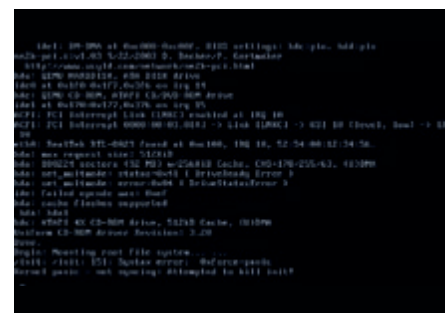
```
svnserve --daemon --root /полный/путь/к/репо1
```

```
svnserve --daemon listen-port=3691 --root /полный/путь/к/репо2
```

должны запустить один сервер на стандартном порте (3690), а другой на 3691. Запрос к второму серверу будет с использованием URI вида svn://hostname:3691/путь/к/файлу. Это работает, но хлопотно для пользователей второго репозитория, так как они каждый раз должны указывать номер порта. Вы можете использовать различные имена хостов, отредактировав /etc/hosts и сделав второе имя псевдонимом (алиасом) к первому. Строка в /etc/hosts выглядит так:

```
IP-адрес имя_хоста псевдонимы
```

Здесь можно указать любое число псевдонимов (разделяя их пробелами). Добавьте псевдоним для второго репозитория, а затем сделайте это на других компьютерах в сети (если Вы используете локальный DNS-сервер, достаточно внести изменения лишь на нем). Сделав это, можно запустить два экземпляра svnserve с помощью



Встречаясь с непонятной ему ошибкой, ядро «паникует». Это может означать сбой в оборудовании, глючный BIOS или внутренние проблемы.

```
svnserve --daemon --listen-host=myhost --root /полный/путь/к/репо1
```

```
svnserve --daemon --listen-host=newhost --root /полный/путь/к/репо2
```

Теперь Вы можете использовать соответствующие имена хостов для каждого репозитория, забыв о мороке с номерами портов. Также учтите, что Ubuntu не содержит скрипта инициализации для svnserve, поэтому Вы должны поместить эти команды в /etc/rc.local перед последней строкой, содержащей exit 0. **НБ**

4 Ретранслятор LIRC

В Знаете ли вы о существовании ПО, способного получать сигнал LIRC, а затем записывать его обратно для повторной передачи? В противном случае, по силам ли мне написать нечто подобное самостоятельно?

Я хочу принимать сигнал LIRC (с пульта) и передавать его снова (через передатчик). У меня уже настроен LIRC с возможностями передачи и приема, но мне нужен способ соединения приемника и передатчика.

Крис Вайк [Chris Wike]

О Исходя из озвученных запросов, Вам нужно использовать irxex, эта программа умеет выполнять любые требуемые команды. Сюда входит и использование irsend для отправки команд через инфракрасный передатчик. Команды, выполняющиеся для каждой нажатой кнопки, определены в Вашем файле .lircrc и выглядят примерно так:

```
begin
prog = irxex
button = Record
repeat = 3
config = irsend SEND_ONCE remote CODE
end
```

Поле **prog** должно быть установлено в **irxex**, а **button** — это идентификатор кнопки, на нажатие которой нужно реагировать. Чтобы узнать идентификатор каждой конкретной кнопки на Вашем пульте, запустите **irw** в терминале, нажимайте несколько клавиш и просмотрите вывод для каждой из них. Далее, опция **repeat** = предотвращает многократный вызов **prog**, если задержаться на кнопке дольше по времени, чем продолжительность одной передачи. Значение, равное трем, означает, что три последовательных сигнала будут

рассматриваться как единое целое. И, наконец, **config** содержит команду, которую Вы собираетесь запускать – в нашем случае это **irsend**, но тут также могло быть что-нибудь другое.

Чтобы все это заработало, Вы должны иметь запущенным **irexec**. При старте из терминала он будет по умолчанию загружать свои настройки из `~/lircrc`. Чтобы использовать другой конфигурационный файл, передайте его в командной строке. Кроме того, если Вы запускаете **irexec** при старте системы, он не сможет определить Вашу домашнюю директорию, поэтому необходимо указать путь к файлу конфигурации. Вы также должны использовать опцию `--daemon` для запуска программы в фоновом режиме. Таким образом, полная команда для запуска будет иметь вид

```
irexec --daemon /etc/lircrc
```

если Вы храните настройки в `/etc/lircrc`. Поскольку в данном случае **irexec** запускается не из оболочки пользователя, он не будет иметь доступа к Вашему полному окружению, и мы рекомендуем использовать абсолютный путь для любой команды, которую Вы указываете в файле **lircrc**. **ГМ**

5 Ликвидация как класс

В Недавно я прочитал в *Which?* статью о стирании жестких дисков: www.theregister.co.uk/2009/01/08/hard_drive_hammer_destruction. Понимаю, что совет *Which?* разбивать старые диски молотком – типа юмор; а что бы порекомендовал Linux Format? Я обычно запускал *Darik's Boot And Nuke (DBAN)*, но недавно начал использовать для зачистки винчестера **dd**. В информации Great Zero Challenge на <http://16systems.com/zero.php> четко говорится, что это гарантия практической невозможности восстановления данных.

Узнав это, я тут же и перешел на **dd**, а потом две недели гонял программу восстановления диска *Ontrack* и не извлек ни одного байта полезной информации. Однако не считаете ли вы, что для ввода лучше использовать `/dev/urandom` вместо `/dev/zero`?

Дэмиан Ланни [Damian Lunny]

О Это правда, что при удалении файлов их содержимое не исчезает, и журнал *Which?* также прав, заявляя, что единственный способ гарантировать невозможность восстановления данных заключается в разрушении диска. В конце концов, *TestDisk* может легко вернуть содержимое удаленных файлов, и даже перезапись нулями не считается достаточной защитой от аналитического оборудования спецслужб. Тем не менее, разбивать жесткий диск молотком – не самая хорошая идея, при всем удовольствии отвести душу. Даже если Вы избежите осколочного ранения, уничтожение диска таким способом неразумно с точки зрения экологии.

В конечном итоге, насколько далеко следует зайти, определяется ценностью Ваших данных. Вряд ли кто станет тратить свое время и разоряться на дорогое спецоборудование, чтобы восстановить картину Ваших отпускных эскапад (независимо от их свойств). А вот конфиденциальная информация компании – дело другое; впрочем, даже и эти сведения все-таки не всегда оправдывают физического истребления диска. Помните и про такой аспект: держа у себя чу-

жие персональные данные – будь то сотрудники, несовершеннолетние, иждивенцы или любые другие лица – Вы несете ответственность за обеспечение сохранности этой информации в соответствии с Законом о защите данных [в России – Федеральный закон «Об информации, информационных технологиях и о защите информации», – прим. пер.]

Вы упомянули *Darik's Boot And Nuke*, и это отличный инструмент для полного и тщательного удаления всех данных с жесткого диска – гораздо радикальнее, чем простое обнуление. *DBAN*, конечно, проработает дольше, но это лишь свидетельство его аккуратности. Вдобавок Вы завоеуете веру в истинность удаления Вами данных у тех людей, которые разбираются в этом вопросе. Так что, по сути, все зависит от того, насколько важен для Вас этот процесс и сколько времени Вы готовы на него потратить. Обнуление гарантирует добрых 99 % неудач при попытках восстановить данные – спросите себя: может, оно и достаточно?

Если Вы храните конфиденциальные данные, первым делом рассмотрите вариант шифрования файловой системы. Он не только защитит



➤ *MythTV* – одна из многих программ, использующих **LIRC**, и один процесс может поддерживать инфракрасное управление несколькими программами.

Вопрос-победитель (английская версия)

★ Опасения UNR

В Будучи новичком в Linux, я прочел вашу серию про **Aspire One** с большим интересом. А сейчас я хочу установить **Ubuntu Netbook Remix (UNR)**, но боюсь, что не смогу убедить дистрибутив найти встроенную в **Aspire One** карту Wi-Fi, а это для меня будет катастрофой. Не подскажите ли, где взять книгу или журнал с объяснением, как управлять Linux в терминале и настроить операционную систему?

Пол Фармер [Paul Farmer]

О UNR можно запустить с загрузочного USB-брелка, содержащего Live-версию дистрибутива. Live-дистрибутив, хоть на USB-брелке, хоть на CD, стартует прямо с этого носителя и никоим образом не затрагивает программное обеспечение, уже установленное на Вашем компьютере. Таким образом, одно из (многих) преимуществ этой системы – возможность попробовать дистрибутив без малейшего риска. Если Ва-

ше беспроводное оборудование заработает с Live-версией, смело ставьте дистрибутив на диск. Если нет, то Вы ничего не потеряете, кроме нескольких минут Вашего времени.

При инсталляции, в зависимости от имеющегося у Вас места, Вы можете установить UNR параллельно существующей системе и выбирать, какую из них использовать во время загрузки. Это отличный способ опробовать дистрибутивы, и его также применя-

ют те, кто хочет работать в Linux, но вынужден сохранять Windows для некоторых задач. Вам это будет предложено в качестве опции на этапе разметки диска в процессе установки.

Что касается книг и журналов – лучший журнал, который мы можем рекомендовать, уже в Ваших руках. Попробуйте *Rute*, это хороший справочник. Его можно купить в виде книги, читать онлайн (<http://rute.2038bug.com/index.html.gz>) или найти на нашем DVD.

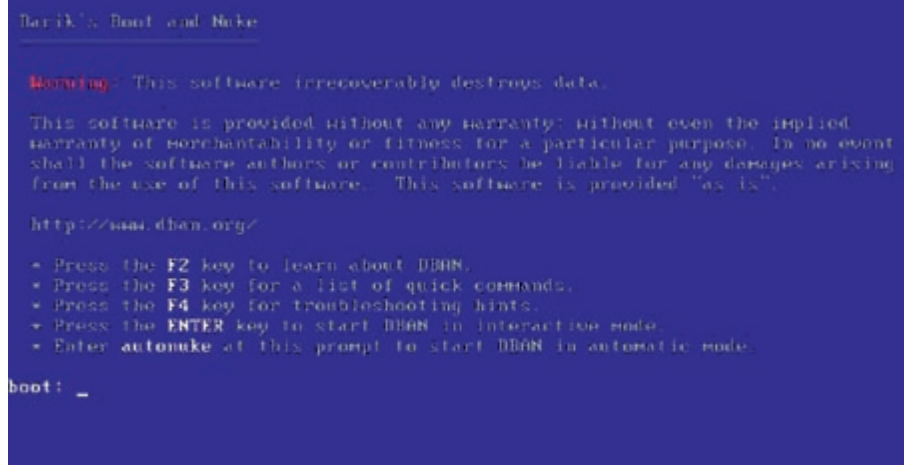
Ваши данные в случае кражи или потери компьютера (воруют и ноутбуки, и настольные машины, так что последние тоже не мешают шифровать), но и добавит лишний рубеж защиты от восстановления после удаления – благодаря тому факту, что любой вариант восстановления предоставит данные только в зашифрованном виде, а чтобы заполучить настоящие, придется еще попотеть. Эффективная защита себя и других – всего лишь обеспечить, чтобы затраты на получение доступа к информации обходились чуть дороже самой информации; в получении 100 % уверенности через разрушение диска крайности нет, если только Вы не сотрудник службы безопасности или член криминального синдиката. **МС**

6 Портативные пароли

В Обдумывая идею сохранения файла с паролями и другой ценной информацией на USB-брелке, зашифрованном с помощью *GnuPG*, чтобы иметь возможность подключить его к любому ПК, а затем расшифровать и прочитать файл. Я хотел бы поставить на брелок и сам *GPG*, чтобы я всегда мог это делать без установки *GPG* на компьютер: тогда на чужой машине не осталось бы следов моих действий. Как это сделать? И вообще, возможно ли такое?

Хью Ченг [Huw Cheng]

Вам придется записать на USB-брелок и свой личный ключ, поэтому единственным средством защиты Ваших данных будет пароль – один пароль, и ничего больше. Приемлем ли для Вас такой риск, решать только Вам – но если Вы решили идти этим путем, за-



» *Darik's Boot and Nuke* не обеспечивает 100% исчезновения данных с жесткого диска, но его качества хватит всем, кто не держит у себя военных тайн.

ведите пароль уникальный и безопасный. Идея Ваша не невозможна, но нужно собрать статически скомпонованный бинарник *GPG* со всеми требуемыми библиотеками в одном файле.

Скачайте исходные коды *GnuPG* с www.gnupg.org, распакуйте их и перейдите (**cd**) в созданную директорию. После этого запустите следующие команды:

```
export CFLAGS="-static"
./configure --enable-static
make
```

Для сборки из исходных кодов Вам понадобится иметь установленными компилятор (*GCC*) и пакет *Autotools*; а если у Вас Ubuntu, придется установить *build-essentials*. Если на этапе *./configure*

всплывают ошибки якобы из-за отсутствия программ и библиотек, но Вы точно знаете, что они есть, проверьте в своем менеджере пакетов наличие версии **-dev** или **-devel** соответствующего пакета. Эти версии содержат заголовочные файлы, не нужные для работы с ПО, но необходимые, когда Вы собираете другое приложение, которое его использует. Весьма вероятно, что наличие заголовочных файлов решит проблему.

Запускать в конце **make install** не обязательно, так как Вы не собираетесь устанавливать эту версию. Вместо этого Ваша новая программа **GP2** будет в директории **g10**. Проверьте, что она скомпонована статически, командой

```
ldd g10/gpg2
```

»

Часто задаваемые вопросы

Шифрование

» Почему я должен шифровать свои файлы? Мне нечего скрывать.

Вы так уверены? Если вы храните на своем компьютере счета или получаете письма по электронной почте, значит, надо защитить пароли или финансовые данные.

» Ну, допустим, у меня есть информация, которую я предпочел бы защитить. Что я должен для этого сделать?

Можете зашифровать файлы с помощью, например, *GnuPG*. Тогда их содержимое будет недоступным, пока вы не дешифруете его с помощью пароля. *GnuPG* – GPL-заменитель *PGP* (Pretty Good Privacy), стандарта де-факто для шифрования файлов.

» Шифровать файлы поштучно – большая морока. Нет ли более простого способа их обезопасить?

Есть такой: это шифрованная файловая система. Она автоматически шифрует файлы при записи и дешифрует при чтении. Вы набираете пароль только один раз, когда монтируется сама файловая система. Если ваш ноутбук будет потерян или украден, никто не сможет прочесть информацию с зашифрованного каталога (обычно это */home*).

» Почему только */home*? А все зашифровать нельзя?

Нет, потому что пропадет доступ к ПО, необходимому для монтирования разделов. Да и какой смысл шифровать системные директории, если там только бесплатно загружаемые пакеты?

» И что мне надо сделать?

Тут есть альтернативы. Ядро Linux содержит *dm-crypt*, используемый для

создания полностью зашифрованных файловых систем. Для управления их созданием и монтированием имеется несколько программ. Я предпочитаю *LUKS* (Linux Unified Key Setup). Другой вариант – *EncFS*. Это файловая система *Fuse* (см. **LXF80**). Вместо шифрования всей файловой системы, она защищает индивидуальные файлы: их имена и содержимое. Это не самый безопасный вариант, поскольку размеры файлов и их права доступа все-таки видимы, но он более удобен.

» А как насчет резервных копий?

Если вы шифруете всю файловую систему, нужно смонтировать ее перед снятием копий; для защиты резервных копий зашифруйте их с помощью *GnuPG*. Когда вы используете *EncFS*, сама файловая система стандартна, зато содержимое файлов выглядит как мусор. Можете использовать любую программу для сохранения их резервных копий, файлы останутся зашифрованными. Это удобно, если

резервные копии хранятся на удаленном сервере.

» Спасет ли меня вся эта абракадабра, если я потеряю свой ноутбук?

Увы, нет, если его похитят спящим (*suspend*) при неразмонтированной зашифрованной файловой системе. *EncFS* автоматически выполняет размонтирование файловой системы, если она какое-то время не используется – но только при отсутствии открытых файлов. Если для шифрования раздела, например, */home*, вы используете *dm-crypt*, нужно выйти из системы и размонтировать */home* перед переводом компьютера в спящий режим.

» Где я могу узнать больше?

Массу информации об *encfs* можно найти на www.arg0.net/encfs, а по *GnuPG* – на www.gnupg.org. А чтобы почитать более подробно о *LUKS*, обратитесь к www.flyn.org/projects/luks-tools/index.html.

которая должна подтвердить, что это не динамический исполняемый файл. Теперь можете скопировать его на свой USB-брелок, но еще раз советуем сперва подобрать «сильную» парольную фразу для Вашего ключа.

Поскольку такое использование *GnuPG* означает размещение личных данных на легко теряемом и подверженном краже устройстве, будет также правильно создать для него отдельную пару ключей. Или попробуйте альтернативу: например, *Ccrypt* (<http://ccrypt.sourceforge.net>). Он использует 256-битное шифрование AES, поэтому достаточно безопасен, а его библиотеки найдутся на любой машине. Он содержит команды *ccencrypt* и *ccdecrypt*, которые совершают действия, отраженные в их названии, вместе с *ccat*, отображающей содержимое зашифрованного файла без записи расшифрованной версии на диск. Так, Вы можете вытащить только Ваши банковские пароли с помощью команды

```
ccat passwords.cpt | grep mybank
```

не оставляя следов незашифрованной информации на устройстве, а главное – на компьютере, к которому оно подключено. **НБ**

7 Экран мал, окна велики

Начал использовать *Aspire One* с *Ubuntu 8.10*, но у меня возникли проблемы с размером экрана. Я был абсолютно не в состоянии настроить почту в *Evolution*, поскольку кнопка перехода ко второй странице была ниже границы экрана. Пришлось сдаться и установить *Thunderbird*. Мало того, я и *Inkscape* использовать не могу. Соответствующие элементы управления в верхней и нижней части экрана у меня только

переводят на страницу вверх или вниз. Нет ли способа заставить окна программ уместиться на экране?

Ральф Мессе [Ralph Massie]

К сожалению, бывают недружелюбные к нетбукам программы – с минимальными размерами окон по умолчанию, не влезаящими на небольшой экран. Но это не беда! Есть целый ряд шагов, способных решить эту проблему. Иногда и сам размер окна по умолчанию слишком велик, и достаточно нажать на кнопку *Распахнуть [Maximize]*, чтобы окно совпало с экраном.

Другой шаг – уменьшить размер меню или панелей задач, предоставив более широкий простор для самой программы. Аналогично можно обрабатывать каждую программу в отдельности – благодаря мелкому шрифту и иконкам она потребует для эффективной работы меньшего размера окна. Некоторые программы также имеют полноэкранный режим, когда границы окна и гаджеты вообще убраны из вида. Обычно на нетбуках мы так запускаем *Firefox* – остается больше места для web-содержимого.

Конечно, все это предполагает, что Вы в состоянии добраться до элементов управления настройками, что возвращает нас к исходной проблеме. Окна допускают частичное выдвигание



➤ Если вам настолько понравился *Battle for Wesnoth*, что вы хотите написать о нем книгу – свяжитесь с разработчиком.

за экран, и Вы можете перетащить окно вверх, чтобы добраться до кнопки внизу. Перетаскивая окно за строку заголовка, этого сделать нельзя, потому что заголовок-то Вам и нужно убрать с экрана. Вместо этого, кликните мышью в любом месте и, удерживая клавишу *Alt*, перетащите окно куда надо.

Комбинируя приведенные способы, Вы сумеете заставить большинство программ сработать с дисплеем *Aspire One*. **НБ**

8 Wesnoth — дело тонкое

Прошу вашего совета по вопросу, к которому, сказать откровенно, не знаю, с какой стороны подступиться. Я – молодой литератор, пишу в жанре фэнтези. Вот уже несколько лет с увлечением играю в *Battle for Wesnoth*, давно созрела идея написать несколько романов по этой игре, но я не знаю, какие соглашения мне нужно заключить, чтобы не нарушать авторских прав ее создателей. И кого вообще считать правообладателем, учитывая, что мир Веснота создавался большим количеством людей?

Вера Сучкова

Сам *Wesnoth* выпущен по лицензии *GPL v2.0*, и, казалось бы, все должно быть легко. Ан нет! Задавая этот вопрос, Вы, фактически, говорите о переработке, и тут начинают сложности.

С одной стороны, игра – это программа, а программы в смысле защиты приравнены к литературе. С другой стороны, игра – это не только программа, но и сюжет, который сам по себе литература, притом без всякой *GPL*. А есть еще и персонажи, которые также защищены авторским правом, есть совместное постановление Верховного и Высшего арбитражного судов – там все очень жестко.

Так что выход у Вас один: связаться с Дэвидом Уайтом [David White] и заключить лицензионный договор на переработку произведения, включая и использование персонажей. Количество соавторов у Дэвида пусть Вас не смущает – это не Ваши заботы. **ВЖ**

Краткая справка про...

Nice

Помните, как мама учила вас вежливости? Она хотела, чтобы вы были дружелюбнее к другим детям и не отнимали у них все игрушки. То же самое применимо к программам, запускаемым на вашем компьютере. У вас конечное число циклов работы процессора, выделяемых на все ваши приложения; если одна программа заберет львиную долю их себе, пострадают другие программы. *Linux* поставляет с командой, заставляющей приложения вести себя любезнее, и она называется *nice*. Вы можете запустить ее так:

```
nice любая_программа
```

или

```
nice -n 5 любая_программа
```

Число после *-n* – это приращение «любезности» программы. По умолчанию программы запускаются с «любезностью» **0** (чем выше число, тем «любезнее» запускаемая программа); программы уступают программам

с меньшей любезностью, но будут иметь приоритет над программами с большей любезностью. Цифра в команде *nice* добавляет значение к существующему: так, дважды запустив *nice -n 5*, получим любезность, равную **10**.

Наивысшая любезность равна **19**, и она специально предназначена для использования процессора, только когда он никому не нужен. Программы, запущенные с таким параметром, могут работать очень медленно, но на реакции системы это никак не скажется. Обычный пользователь может только увеличивать значение любезности, но при наличии доступа *root* можно также вводить отрицательные значения. Сломать систему очень легко, поэтому «строить» программы, заставляя их быть любезнее, разрешается только суперпользователям.

Хотя *nice* помогает в загрузке процессора, она ничего не может поделать с программами, требующими частого доступа к диску и блокирующими процессор ожиданием ввода/вывода. Решением данной проблемы является *ionice*, которая делает со вводом/выводом то же, что *nice* с процессором.

Большой вопрос Как установить Linux на нетбук?

9 Установка без CD-привода

Недавно я приобрел компьютер Eee Vox, намереваясь установить на него Linux. Однако у него нет привода CD-ROM. Как перенести ISO-образ Linux на USB-брелок и установить дистрибутив с него?

Дональд Ридделл [Donald Riddell]

С появлением нетбуков все больше дистрибутивов предлагают опцию переноса инсталлятора на USB-брелок. Настольные компьютеры обеспечивают еще больше вариантов: можно, например, начать с образа CD/DVD и создать свой личный установочный USB. Тем, кто лишен подобной роскоши, Ubuntu предоставляет Ubuntu Netbook Remix (UNR) в качестве образа для копирования непосредственно на USB-брелок (<https://wiki.ubuntu.com/UNR>). Это не просто удобно: UNR еще и оптимизирован под ресурсы нетбука. Скачав его, переместите файл на USB-брелок командой

```
dd if=ubuntu-9.04-netbook-remix-i386.img of=/dev/sdX
```

где /dev/sdX – Ваше USB-устройство. Используйте путь к самому устройству, а не отдельно-

му разделу типа /dev/sdb1. На устройстве, которое должно иметь размер не менее 1 Гб, все старые данные будут стерты, поэтому убедитесь, что на нем нет ничего важного. Теперь можете загрузиться с этого USB-брелка и попробовать UNR, прежде чем установить его.

PCLinuxOS также предусматривает установку на USB-брелок, так что загрузитесь с Live CD и выберите System > Make LiveUSB из меню. Следуя инструкциям на экране, убедитесь, что указали правильное устройство для записи, и загрузочный Live-дистрибутив запишется на USB-брелок.

Если Вы хотите установить Debian, процедура будет другая. Не пытайтесь влихнууть на USB-брелок все файлы базовой установки Debian, проект Debian Eee PC (<http://wiki.debian.org/DebianEeePC>) использует в качестве отправной точки сетевой установочный Debian CD. Этот минимальный образ, размером около 16 Мб, содержит достаточно данных, чтобы подключиться к сети и запустить инсталлятор. Тогда все необходимое для установки будет загружаться по требованию. Загрузите файл образа, следуя по

ссылке How to Install на странице Debian Eee PC, и скопируйте его на USB-брелок командой

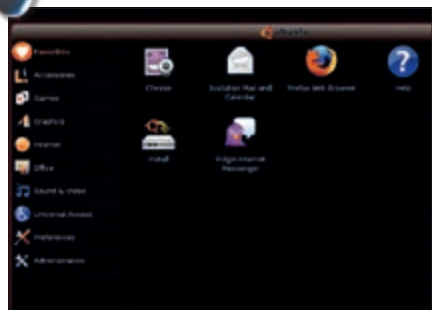
```
dd if=debian-eeepc.img of=/dev/sdX bs=4096
```

Опция bs задает копирование данных крупными блоками (4096 байт вместо 512), что ускоряет процесс. Кстати, этот метод пригоден также и для UNR. Как и в случае с образом UNR, все данные, имеющиеся на Вашем USB-брелке, пропадут; но учтите, в этом случае не обойтись без сетевого подключения. Порядок здесь такой же, как при использовании CD или DVD Debian, разве что добавится время на скачивание пакетов.

Другой вариант, работающий с большинством CD-инсталляторов – Unetbootin. Эта программа создает загрузочный USB-брелок с ISO-образа CD или DVD. Вы можете добыть Unetbootin с <http://unetbootin.sourceforge.net>, если его нет в менеджере пакетов Вашего дистрибутива; имеется также версия для Windows – это полезно, если у Вас есть оптический привод только на компьютере с Windows. Для некоторых дистрибутивов программа может даже сама скачать ISO-образ.



Шаг за шагом: Готовим live USB-брелок для нетбука



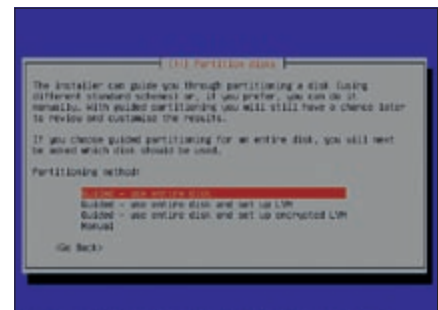
1 Ubuntu

UNR спроектирован для нетбуков, и его можно запустить прямо с USB-брелка, чтобы убедиться в работоспособности оборудования до установки.



2 PCLinuxOS

PCLinuxOS также отлично запускается на нетбуках и имеет утилиту для создания загрузочного Live USB-брелка, и вы можете опробовать или установить дистрибутив.



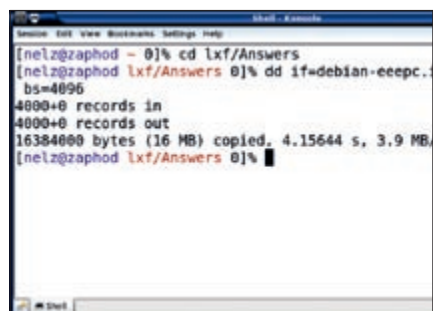
3 Debian Lenny

Сетевой инсталлятор Debian, может, и не особо красив, зато хорошо делает свое дело, и существует специальная версия для Eee PC.



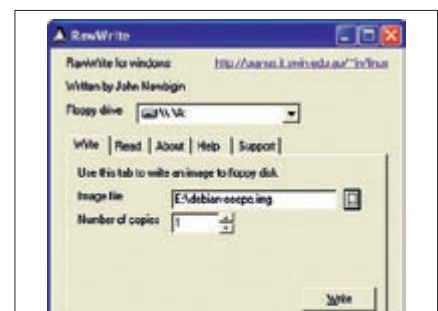
4 Unetbootin

Можно создать USB-версию установочных CD или DVD большинства дистрибутивов при помощи Unetbootin – пригодится, если у вашего дистрибутива нет собственных USB-вариантов.



5 Копируйте образ...

Используйте dd для переноса образа на USB-брелок, но убедитесь, что имя выходного устройства выбрано верно, не то угробите полезную информацию.



6 ...или копируйте в Windows

Если компьютер с CD-приводом у вас есть только под Windows, просто загрузитесь с Live CD и используйте dd или RawWrite с <http://www.chrysocome.net/rawwrite>. LXF



Лучшие новинки
открытого ПО на планете

LXF HotPicks



Энди Хадсон

Когда Энди не притворяется, что отлаживает почтовые сети, он ныряет среди коралловых рифов в поисках жемчужин для HotPicks.

Knowledgeroot » Rawstudio » Grsync » Umit » Toy Cars » Ardentryst » Neverputt » Psi » DiscWrapper » Klavaro » MtPaint

Платформа для обмена знаниями

Knowledgeroot

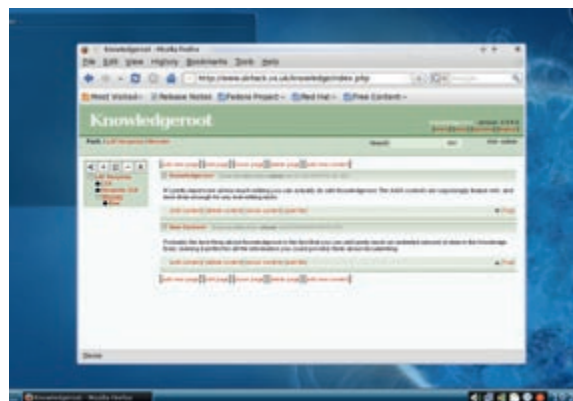
Версия 0.9.8.6 Сайт www.linuxdelta.de

Благодаря Википедии, вики стала стандартным средством для обмена данными. Однако навигация в вики бывает затруднительна, ведь при работе с содержимым приходится полагаться как на поиск, так и на ссылки внутри статей. Для Википедии это, может быть, и здорово, но когда такого рода навигация не годится, стоит обратить внимание на Knowledgeroot.

Это онлайн-приложение основано на принципе базы знаний — репозитория, предназначенного для хранения коллективных знаний отдела или компании, облегчающего сотрудникам задачу поиска и просмотра информации из общего источника.

Knowledgeroot предлагает древовидное отображение содержимого вашей базы знаний. Подобный иерархический подход идеален для отображения знаний, содержащихся в базе; именно так работает с содержанием своих книг сервис Safari Books Online.

Будучи web-приложением, Knowledgeroot использует компоненты, имеющиеся практически в любой LAMP-системе. Его легко запустить и начать работать: просто загрузите файлы в каталог на своем сервере и затем откройте `install.php` в своем web-браузере. Начальная настройка на диво проста и укладывается всего лишь в десяток вопросов. Вы и охнуть не успеете, а программа уже будет установле-



» Знание — сила: поделитесь ею с членами своей команды.

на и готова к использованию. В качестве движка базы данных можно использовать *MySQL* или *PostgreSQL*, но сперва Knowledgeroot представляет собой просто чистый холст. Прежде чем выплескивать на него свой запас знаний, измените стандартные имя пользователя и пароль для администратора.

Обеспечьте безопасность своих знаний

А после этого добавляйте контент. Используя интерфейс Ajax, Knowledgeroot предлагает весьма впечатляющий выбор опций форматирования. Учтите также, что он не полагается на какой-либо язык разметки — вместо этого вам предлагается настоящий интерфейс WYSIWYG, поэтому новые пользователи не будут испытывать с ним никаких затруднений. Вы даже сможете использовать управление доступом, чтобы обезопасить свою базу знаний от чужих глаз. В общем, это отличная альтернатива вики.

Исследуем интерфейс Knowledgeroot

Учетные записи

Здесь редактируются настройки и параметры учетной записи.

Панель инструментов

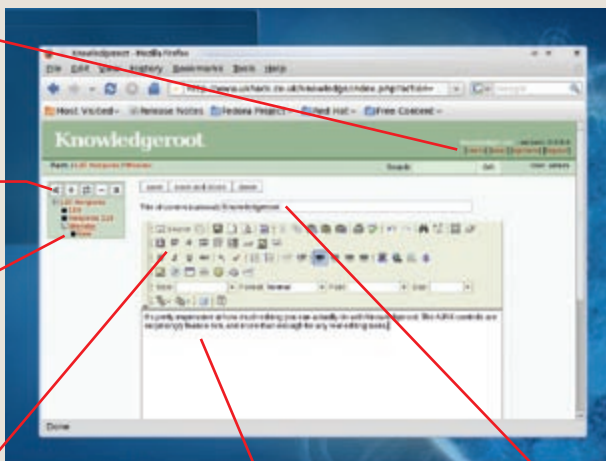
Эти кнопки управляют поведением панели навигации.

Панель навигации

Здесь содержится дерево документов вашей базы знаний.

Инструменты

Функции форматирования, помогающие украсить ваш контент.



Окно содержания

Очень важное место; именно здесь редактируется текст.

Поле заглавия

Введите сюда заглавие для раздела.

Редактор формата RAW

Rawstudio

Версия 1.2 Сайт <http://rawstudio.org>

Камеры-«мыльницы» неплохи для моментальной съемки на вечеринках, но им не сравниться со своими цифровыми SLR-аналогами, когда речь идет о качестве изображения. Возможность использовать форматы RAW – ключевой момент в этом преимуществе, поскольку благодаря им снимки очень подробны и в равной степени подходят как для рядовой обработки, так и для профессионального творчества.

Цель *Rawstudio* – предоставить интерфейс для такого редактирования, с поддержкой разнообразных форматов RAW (да, на свете есть отнюдь не один RAW-формат). Его интерфейс, выдержанный в серых тонах, немного напоминает *Photo-shop Elements*, а это вовсе не плохо.

При импорте изображений, *Rawstudio* позволяет просканировать содержимое директории и автоматически вывести все имеющиеся в ней снимки. Кроме того, можно проводить рекурсивное сканиро-

вание на предмет выбора фото из поддиректорий. Наведя курсор мыши на изображение, вы увидите краткую выжимку из данных Exif, содержащую информацию о том, с какими установками был сделан снимок; это позволит вам быстро отобрать нужные фотографии.

Контролируйте себя

Начиная отсюда, интерфейс предоставляет целый набор регуляторов для редактирования ваших снимков. Они в общем-то стандартны, но изюминка в данной формуле – наличие доступа к трем вкладкам, между которыми можно переключаться и редактировать одно и то же фото тремя разными способами: например, на вкладке А настраивать насыщенность цвета, на вкладке В – контрастность, а на вкладке С менять тона.

Rawstudio также позволяет присваивать фотографиям разные приоритеты, помогая вам сфокусироваться на снимках,



► *Rawstudio* отменно редактирует RAW-файлы.

которые надо срочно обработать, в отличие от тех, что могут подождать. Столь же полезное дополнение – возможность одновременного редактирования нескольких фото с использованием настроек с одной или нескольких вкладок, и *Rawstudio* использует для этого регулярные выражения, что способствует генерированию уникальных файловых имен. И, наконец, умение сохранять изображения в форматах JPEG, TIFF, а также PNG, и менять общий объем фотографий делают это приложение полезным дополнением арсенала фотографа-любителя.

Оболочка для Rsync

Grsync

Версия 0.6.3 Сайт www.opbyte.it/grsync

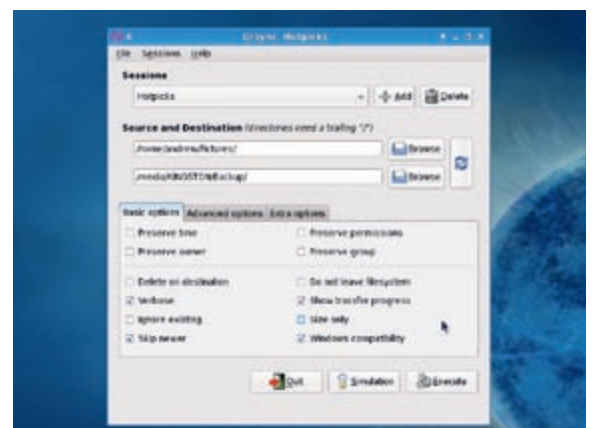
Кажется, нет конца графическим утилитам, основанным на *rsync*, но *Grsync* действительно отличается от предшественников. В Fedora *Grsync* выглядит органичной частью этого дистрибутива, в отличие от некоторых инструментов, которые жаждут выделиться из среды Gnome. Похоже также, что *Grsync* крепко усвоил руководящие принципы интерфейса Gnome, поскольку все действия совершаются на одном экране; правда, можно выбирать между тремя вкладками.

Вкладки

В этих трех вкладках сгруппированы опции разного уровня; так, если вы просто что-то копируете, вам надо побеспокоиться только о первой из них. Остальной интерфейс замечательно лаконичен, и все аккуратно помечено. Настройки снабжены краткими подсказками, появляющимися через секунду после наведения на них

курсора: они уточняют назначение параметра и ключ, который был бы добавлен, если бы вы использовали функцию в командной строке. Хотя опытным пользователям быстрее будет запустить *rsync* из терминала, нельзя не отметить, что благодаря подсказкам новичкам легче освоиться с различными опциями. Было бы неплохо, если бы *Grsync* еще и показывал вам всю команду целиком, помогая выучить синтаксис *rsync*. И это, вкуче с неумением писать в *crontab* (что затрудняет планирование конкретного задания), вызывает некоторое сожаление, потому что во всем остальном это отличная маленькая утилита в опрятной упаковке.

**«Подсказки помогут
новым пользователям
освоиться с опциями.»**



► *Grsync* безопасно и надежно скопирует ваши данные, заодно обучая вас разбираться в опциях.

Еще одна функция, достойная упоминания – возможность эмулировать выполнение задания, чтобы увидеть, правильно ли подобраны настройки: иными словами, протестировать работу до того, как к ней приступить, и убедиться, что все выглядит как надо. Вы будете благодарны за это, если когда-нибудь вам придется восстанавливать данные, которые вы копировали. Можно также активировать журнал – это позволит отследить, почему то или иное задание завершилось с ошибкой.

Инструмент сканирования сети

Umit

Версия 1.0beta 2 Сайт www.umatproject.org

Исследуйте топологию своей сети при помощи удобной графической программы Umit.

Если вы всерьез намерены стать администратором, придется многое изучить про сеть. Мы имеем в виду не подсоединение сетевых кабелей к хабу, а близкое знакомство с сетевой безопасностью и векторами атак.

Nmap – в общем, подходящий сетевой сканер, но разобраться с ним непросто – в частности, потому, что это инструмент командной строки. Цель Umit – облегчить работу в терминале для новых пользователей Nmap: дружелюбным интерфейсом GTK и богатым набором подсказок, которые помогут справиться с задачей сетевого сканирования. При простейшем раскладе вам придется всего-навсего ввести цель (желательно, чтобы это был IP-адрес, но DNS-имя тоже подойдет) и указать тип сканирования. После этого предоставьте Umit сделать все остальное – вы получите полную информацию в окне Nmap Output [Вывод Nmap], а также возможность просмотра дополнительных сведений, как то: открытые в данный момент порты, тип операционной системы и т. п.

В сущности, Nmap Output – просто консоль для вывода информации, получаемой при запуске Nmap с соответствующими параметрами в командной строке. Весьма полезно, что Umit показывает вам команду Nmap, выполняемую для каждого сканирования, позволяя побольше узнать о командной строке в процессе работы. Интерфейс программы построен

по типу мастера: варианты сканирования настраиваются в зависимости от ответов на разнообразные вопросы, а благодаря вкладкам можно задать несколько сканирований и затем сохранить результаты для сравнения в области Compare Results. Там вы быстро заметите разницу между двумя результатами, чему способствует цветовое выделение. А хотите – просмотрите «сырые» данные Nmap или откройте в своем браузере отчет, где отмечены все строки с отличиями.

Блокировка

Прелесть Umit в возможности делать с ним все, что угодно. Он также отлично настраивается: редактируются все аспекты интерфейса, позволяя скрыть или показать любую опцию. Более того, Umit годится и для проведения плановых сканирований, чтобы постоянно следить за вашей сетью: просто настройте требуемый профиль

«Там вы быстро заметите разницу в результатах.»

сканирования, выберите отчет в виде файла или почтового сообщения, и нажмите OK. Umit вставит в расписание соответствующую команду Nmap, используя Cron, и обеспечит вам известное спокойствие,

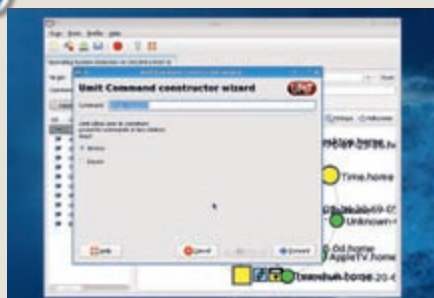
поскольку вы знаете, что состояние вашей сети под постоянным контролем. Это не означает тестирования на предмет проникновения, но предоставляет весьма полезную информацию по решению проблем, особенно если периодически сравнивать результаты сканирования.

Еще один приятный штрих в Umit – возможность создавать и поддерживать инвентаризацию сетевых устройств. Umit приглядит за тем, из чего состоит ваша сеть, и при запуске сканирования можно задать включение новой информации в существующий инвентарный отчет, что позволит отслеживать изменения в перечне устройств.

Здесь есть еще много чего на радость сетевому администратору; наше любимое дополнение – возможность отображения топологии сети и ее графическое представление внутри интерфейса. Коротко говоря, Umit далеко выходит за рамки простого GUI для Nmap: это удобная и полезная самостоятельная утилита.



Шаг за шагом: Настройка сканирования в Umit



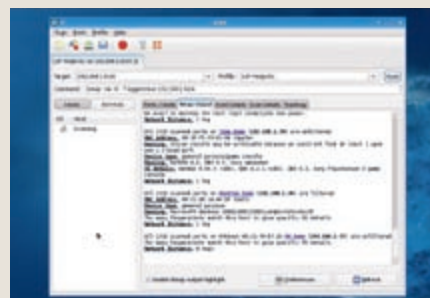
» Приступаем

Для запуска процесса нажмите кнопку Command Wizard [Мастер команд] на панели инструментов. Потом выберите интерфейс: Basic [Основной] или Complex [Сложный].



» Строим сканирование

Выберите параметры, которые хотите включить, и Umit построит на их основе сканирование в Nmap.



» Получаем результаты

Введите целевой сегмент сети для сканирования и спустите Umit с привязи. А теперь просмотрите результаты удобным для себя способом.

HotGames Развлекательные приложения

Приключенческая игра

Ardentryst

Версия 1.7 Сайт <http://jordan.trudgett.com>

Ролевые игры и Linux столь же неразделимы, как жареная рыба и картошка, и неудивительно, что это, пожалуй, наиболее широко представленный в Linux жанр. В этом месяце из общей массы больше всех выделился *Ardentryst*, и мы тут же в него влюбились. Благодаря нехитрым зависимостям (Python и *pygame*) его исключительно легко установить и запустить. Просто зайдите (**cd**) в директорию, куда распаковали zip-файл, и скомандуйте *.ardentryst*.

С ходу бросаются в глаза внимание и силы, затраченные на графику: главное меню великолепно выглядит и искушает вас тут же приступить к процессу. Продвигаясь в игре, трудно не восхищаться визуальными эффектами; хотя они и относительно просты, но перемещение объектов и старательно разработанный фон добавляют игре ощущения приключения

и удовольствия. Система управления отличается достойной простотой — ведут вас кнопки со стрелками, а блок клавиш на клавиатуре используется для командования сражениями и наложения чар; различные их комбинации предоставляют доступ к новым заклятиям и боевым стилям. Постепенно вы набираете опыт, и ваши способности к заклятиям и боевые навыки растут по мере углубления в игру.

Встречи и повороты

Сам по себе сюжет очень прост, хотя попутно предлагается немало побочных квестов. Вы будете встречаться и взаимо-



» Прорубайте себе путь через преграды в этой с любовью сделанной приключенческой ролевой игре.

действовать с периодически возникающими на пути персонажами (они также будут давать вам задания), покупать и продавать предметы в торговых точках, разбросанных по дороге. Вы сможете подбирать эликсиры для восстановления здоровья и маны, и экипироваться новым оружием и броней.

В общем и целом, это хорошо разработанная игра, достаточно глубокая, чтобы доставить удовольствие, и достаточно гибкая. Если вы еще не знакомы с ней, настоятельно рекомендуем уделить ей время.

«Трудно не восхищаться визуальными эффектами игры.»

Гольф

Neverputt

Версия 1.5.1 Сайт <http://neverball.org>

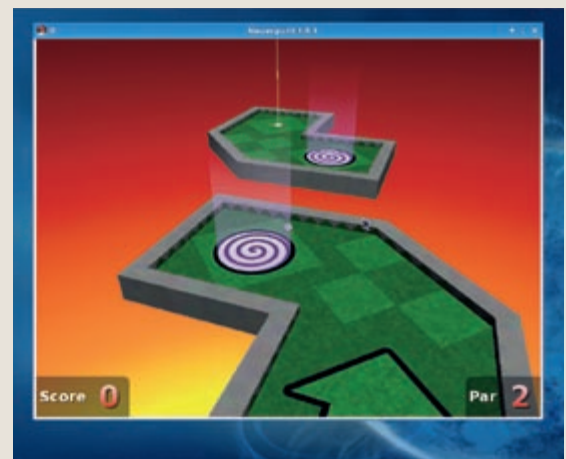
В детстве мы часто выискивали всякие супер-гольф-курсы, чтобы порезвиться во время школьных каникул. Нечего и удивляться нашей радости при известии о *Neverputt* — сумасшедшей игре в гольф, воздвигнутой на игре покрупнее под названием *Neverball*. В ней можно выбирать из нескольких курсов, где лунки варьируются от достаточно простых до клюшколомно сложных.

Основное устройство ввода — мышь: ее перемещения влево и вправо позволяют прицелиться, а перемещения вперед и назад регулируют силу удара.

По части зависимостей, нужно установить SDL, а также *libvorbis*, *libpng* и *libjpeg*.

Верным курсом

Графика *Neverputt* относительно проста, но это не главное, поскольку все внимание уделяется вашим умениям рассчитать угол и силу удара. К счастью, физический движок отличный; мяч рикошетит от стен и скатывается вниз с холма с высокой степенью точности. И это замечательно, поскольку между вами и лунками множество препятствий, и ошибок они не прощают. Немало часов потребует овладение



» Все ощущения сумасшедшего гольфа — и ни одной поломанной клюшки.

«В финале курса для начинающих будут адски сложные лунки.»

основным курсом (в финале уроков для начинающих будет пара адски сложных лунок), и вы ощутите законную гордость своими достижениями. На этой стадии вы наверняка задумаетесь об игре с противниками — ведь нет ничего слаще чувства победы над экс-друзьями на зеленом поле!

Интернет-пейджер

Psi

Версия 0.12.1 Сайт <http://psi-im.org>

Без обмена мгновенными сообщениями использование Интернета в наши дни даже трудно представить, но раньше разные сети не могли общаться между собой. Эта проблема привела к развитию Jabber/XMPP – действительно открытого стандарта, способного соединять друг с другом разные сети и теоретически делающего возможной связь с любым человеком, где бы он ни находился, если он использует сервис на Jabber.

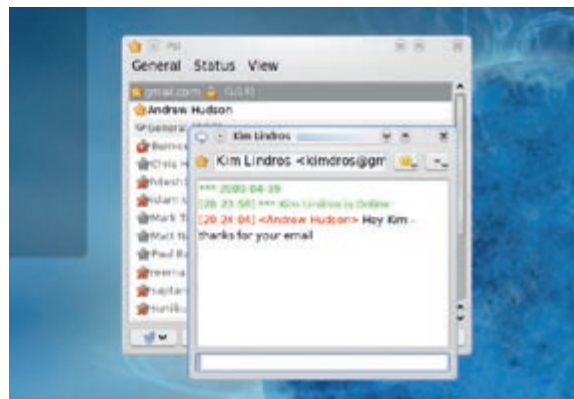
Конечно, большинство IM-клиентов сейчас не зависит от протокола; они умеют соединяться с любой из основных IM-сетей. С другой стороны, *Psi* – сугубо Jabber-клиент, и к этому мы вернемся позднее. Поскольку он разработан для KDE с использованием инструментария *Qt 4*, вам придется обеспечить наличие необходимых библиотек разработки *Qt*, чтобы удовлетворить его зависимости. Одну ошибку мы заметили во время вы-

полнения *./configure* — похоже, *Psi* не нравится, если в названии директории есть пробел. С этой проблемой можно справиться, быстро введя команду *mv*, после чего компиляция займет считанные минуты. Тогда вы сможете запустить *Psi*, удобного пристроившийся к меню Интернет.

Ваш клиент, ваши дела

Загрузив *Psi*, вы осознаете, что он работает только с соединениями Jabber/XMPP, и если многие из ваших друзей сидят в других сетях, то он почти бесполезен. Зато если вы пользуетесь сервисами на основе Jabber, например, Google Talk, к вашим услугам масса полезных опций. В истинном KDE-стиле, они варьируются от общих

«В *Psi* множество полезных опций, достойных изучения.»



➤ Накачанный клиент Jabber, *Psi* даст вам полный контроль над работой в Jabber.

настроек до более детальных параметров приложения и его работы. Достаточно сказать, что здесь полно всего, что нужно, если вы хотите копнуть поглубже. Если нет, *Psi* неплохо справляется с задачей упрощения и без того простых вещей; его внешний вид и работа отличаются лаконизмом и неперегруженностью, и он очень удобен, даже если вас волнует только совместимость с Jabber.

Пожалуй, единственное, что нас огорчило, это неполная реализация поддержки Jingle (голосового элемента Jabber) в *Psi*, хотя работа в этом направлении для клиента Linux уже ведется.

Печать CD-этикеток

DiscWrapper

Версия 1.2.0 Сайт <http://discwrapPer.sf.net>

Возможно, самопальные музыкальные сборники кому-то кажутся далеким прошлым, но их дух жив и поныне – в плей-листах. Создание CD из таких плей-листов тоже может понадобиться, хотя мы тут же столкнемся с неизбежной необходимостью как-то нанести названия треков на диск. Вот было бы здорово избежать этого, создав аккуратную этикетку с перечислением всех треков, а также напечатав что-то на самом CD!

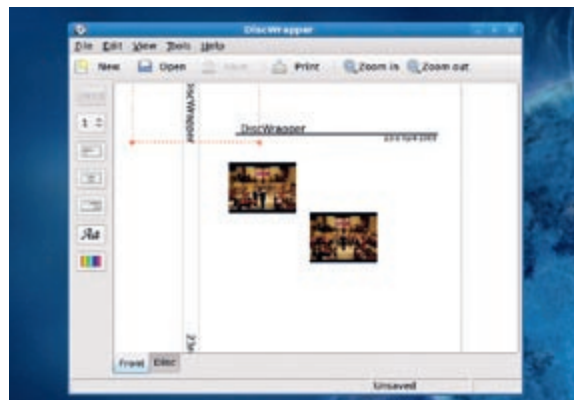
DiscWrapper служит целительным бальзамом для этой саднящей раны, позволяя разрабатывать и печатать вкладыши к коробкам для CD и DVD. Попав в программу, вы увидите, что ее интерфейс очень прост; у вас есть всего лишь чистый холст, несколько вкладок внизу и по несколько кнопок слева и вверху. Но мы бы не назвали его «невзрачным»: он явно рациональный. Вкладки

представляют собой разные части этикетки, над которой вы работаете; здесь предусмотрены области для лицевой стороны, внутренней стороны, оборотной стороны и самого диска, в зависимости от выбора шаблона.

Одетый в слои

Кнопки слева, между прочим, это значки форматирования, а состав инструментов меняется, отражая виды ваших работ. Например, если вы вставляете изображение, появится опция для его масштабирования и смены положения на холсте. Здесь включена библиотека в стиле клипарта, а можно взять и свою картинку.

«Включена библиотека клипарта; а можно взять и свою картинку.»



➤ Простой, но эффективный, *DiscWrapper* позволяет создавать качественные обложки для CD без особых усилий.

Когда вы работаете над наклейкой на сам CD, *DiscWrapper* распределит текст по кругу, а не прямыми строками; но, при желании, эту функцию можно отключить. *DiscWrapper* также использует метатэги, чтобы заглавие и дата были едины для всего проекта – при изменении заглавия на одной вкладке, оно автоматически изменится на всех остальных. Хотя интерфейс незатейлив, *DiscWrapper* фокусируется на том, для чего он, собственно, и разработан: это – простое приложение, чтобы помочь вам справиться с одной из житейских мелочей.

Клавиатурный тренажер

Klavaro

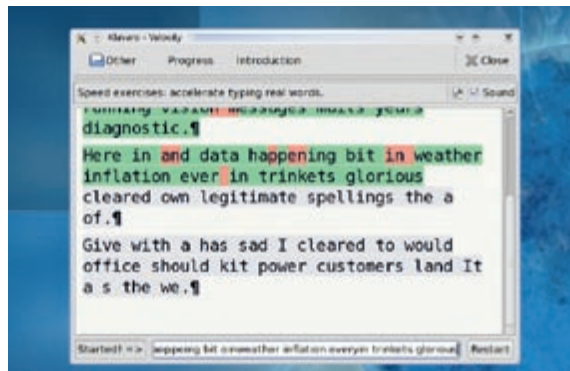
Версия 1.1.9 Сайт <http://klavaro.sourceforge.net>

В LXF93 мы рассматривали *Klavaro*, довольно-таки полезный клавиатурный тренажер, созданный, чтобы увеличить скорость и качество набора текста. Нам приятно констатировать тот факт, что все это время *Klavaro* не стоял на месте: добавилось огромное количество переводов и альтернативных раскладок клавиатуры.

Поскольку в плане зависимостей его требования весьма скромны, вы очень скоро с головой погрузитесь в океан различных клавиатур. Помимо стандартных раскладок для британского (UK) и американского (US) английского, имеются также раскладка Дворжака [Dvorak] и раскладки клавиатуры для большинства языков, включая русский. Упражнения сгруппированы в четыре уровня сложности, от самых основ до скоростного набора произвольных слов. На каждом уровне вы должны одолеть определенные задачи, чтобы продвигнуться дальше – например, в тесте на скорость набора нужно

выдать 97 % правильности при скорости не менее 50 слов в минуту; устрашающее задание. Одна из возможных опасностей – приобретенные с годами плохие привычки, и не исключено, что в *Klavaro* вам придется заново осваивать клавиатуру. Однако мы не сомневаемся, что буквально через пару сеансов работы с этим тренажером вы обретете былую скорость, точность и правильные манеры.

➤ В *Klavaro* множество языков и раскладок клавиатуры.



Редактор изображений

MtPaint

Версия 3.31 Сайт <http://mtpaint.sourceforge.net>

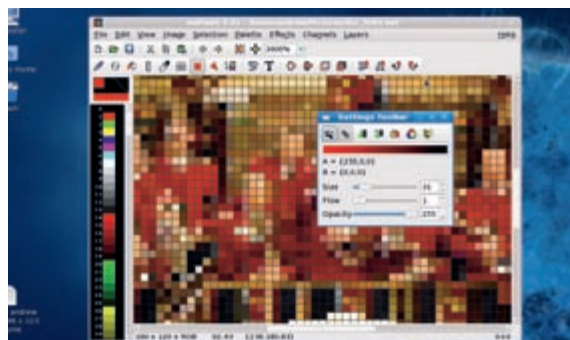
Прошло уже несколько лет с прошлого обзора *MtPaint* (если быть точными, в LXF84), и с тех пор разработка этого удобного редактора изображений не прекращалась. В него вошли несколько новых функций, по большей части благодаря Дмитрию Грошеву, лидеру проекта. Данный релиз включает новый инструмент для подбора палитры и выбора файлов; улучшения в коде также ускорили инструмент Smudge (размазывание). Появилось немало новых эффектов: например, гауссово размывание и экзотично звучащее размывание Кувахары-Нагао [сохраняющее контуры изображения при потере мелких деталей, – прим. пер.].

Однако если раньше *MtPaint* мог собираться с GTK1 или GTK2, то данный релиз – приложение исключительно GTK2, которое уже помахало ручкой архаичному инструментарию, освободив разработчиков от повинности возиться со старым кодом. *MtPaint* не рвется конкурировать

с GIMP, но его функциональность и полезность определенно прогрессируют. Пользователям Windows, переключившимся на Linux, с *MtPaint* будет легко, поскольку он взял за образец пакет *Microsoft Paint*, который будет переработан для Windows 7.

Эти многообещающие изменения позволяют ожидать дальнейших новостей от *MtPaint*; будем следить за ходом развития этого проекта. LXF

➤ Маленький аккуратный редактор изображений с удобными для сиюминутной работы функциями.



Также вышли

Новые и обновленные программы, тоже заслуживающие внимания...

➤ mpg123 1.7.2

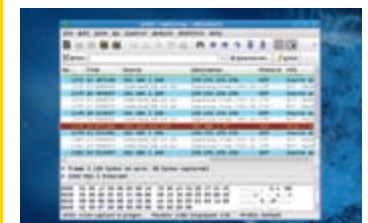
Удобный консольный MP3-плеер, подходящий для старых машин. www.mpg123.org

➤ Wired 0.6

Инструмент редактирования и микширования звука для полупрофессиональных музыкантов и любителей. <http://wired.sourceforge.net>

➤ Wireshark 1.0.7

Дедушка инструментов сканирования сети с графическим интерфейсом. www.wireshark.org



➤ Wireshark помогает не утонуть в опасных водах работы в сети.

➤ Drivel Journal Editor 2.0.4

Редактор блогов для Gnome, сработавшийся с большинством основных блоггинг-платформ. <http://drivel.sourceforge.net>



➤ Обнародуйте свой блог с помощью редактора Drivel на базе Gnome.

➤ CloneZilla 1.2.1-53

Популярная альтернатива Ghost для Linux, особенно полезная, если надо одновременно клонировать диски или разделы дисков на нескольких компьютерах. <http://clonezilla.org>

➤ Liferea 1.4.28

Основанная на Gnome программа для чтения RSS, с множеством различных полезных способов организации новостей каналов. Убойный способ отладить ваш RSS. <http://liferea.sourceforge.net>

➤ DVDStyler 1.7.3 Beta 2

Программа для создания DVD, в стиле iMovie и Premiere Elements, способная работать с несколькими видеоформатами. www.dvdstyler.de

Начните здесь

Диск с двойной загрузкой: два дистрибутива горячей выпечки...



Читайте дальше — вы найдете всю необходимую информацию про DVD! Если вы новичок в Linux, откройте на диске файл `index.html` и перейдите в раздел Справка — там имеются руководства по ОС, в том числе:

- » Что такое Linux?
- » Что такое дистрибутив?
- » Загрузка компьютера с DVD
- » Разбиение жесткого диска на разделы
- » Навигация по файловой системе
- » Учетные записи супер- и обычного пользователя
- » Работа в командной строке
- » Установка программ
- » Помощь онлайн
- ...и еще много полезного!

Майк Сондерс
Редактор диска
mike.saunders@futurenet.com

Дистрибутив Linux

Ubuntu 9.04

И вновь мы преодолели все преграды, чтобы донести до вас расширенную версию Ubuntu — она намного превосходит то, чего можно ожидать от обычного издания на одном CD. Версия на **LXFDVD** включает:

- » Рабочие столы KDE 4.2 и Xfce 4.6.
- » Дополнительные приложения: *AbiWord*, *Gnumeric*, *Thunderbird*, *Scribus*, *Inkscape*, *F-Spot*, *Blender*.
- » Инструменты разработки: *GCC*, *Mono-Develop*, *Anjuta* и другие.
- » Превосходные игры: *Frozen Bubble*, *Neverball*, *Planet Penguin Racer*, *FreeCiv* и *GCompris*.

После установки у вас будет масса нужных программ — не придется ничего скачивать дополнительно! А если вы не любитель KDE, то запросто сможете удалить его: через минуту мы объясним, как.

Эта версия Ubuntu 32-битная, но она будет прекрасно работать и на 64-битных машинах. Если вам нужна 64-битная версия в чистом виде, вы найдете стандартный ISO-образ CD в разделе **Дистрибутивы/Ubuntu** нашего DVD. Остается только записать его на CD-R и загрузиться с него.

Ubuntu работает в режиме Live, и вы сможете опробовать включенное ПО прямо с DVD. Главное, чтобы ваша система соответствовала следующим требованиям: 700 Mrq Intel/AMD-совместимый процес-

сор, 384 МБ ОЗУ и 8 Гб свободного места на жестком диске. (Если хотите установить Ubuntu на более старой машине со 128 МБ или менее ОЗУ, запишите на CD-R образ Alternate ISO из раздела **Дистрибутивы/Ubuntu** нашего DVD и загрузитесь с него.)

После установки вашим рабочим столом по умолчанию будет Gnome, но вы можете переключиться на KDE или Xfce, выбрав про входе другую сессию в меню Settings [Настройки].

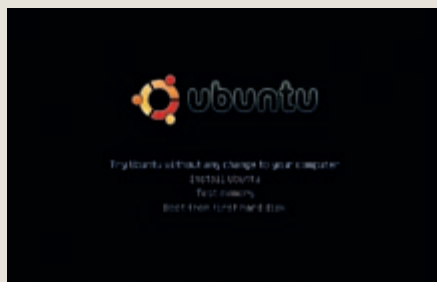
Если вы не используете KDE или какую-либо из его программ, вы, возможно, захотите удалить их, высвободив таким образом место на жестком диске и расчистив меню от излишков. Мы предоставляем быстрый скрипт для избавления от KDE — скопируйте `remove-kde.sh` из раздела **Дистрибутивы/Ubuntu** на DVD в свой домашний каталог и запустите его от имени суперпользователя-root.

Например, вы выставляете **LXFDVD**, жмете **Applications > Accessories > Terminal** [Приложения > Дополнительно > Терминал], и затем вводите

```
cp /cdrom/Distros/Ubuntu/remove-kde.sh
sudo sh remove-kde.sh
```

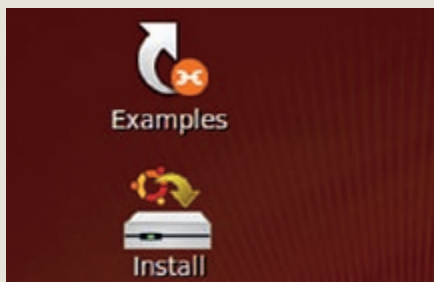
Вам будет предложено ввести свой пароль. Скрипт удаляет более 270 пакетов и высвобождает до 620 МБ места на жестком диске.

Шаг за шагом: Устанавливаем Ubuntu 9.04



1 Загрузка

Запустите компьютер с DVD и нажмите на Enter в появившемся меню (если понадобится изменить порядок загрузочных устройств — информация о том, как это сделать, есть в разделе **Справка/Новичку_в_Linux** на DVD).



2 Рабочий стол

Через некоторое время перед вами появится рабочий стол, где можно опробовать программы. Для копирования Ubuntu на жесткий диск дважды щелкните по Установка [Install] на рабочем столе.



3 Установка

Когда появится программа установки, следуйте подсказкам, чтобы выбрать язык, раскладку клавиатуры и регион. В случае ошибки, нажмите кнопку Back [Назад] и все исправьте.

А как сделать...?

Установив Ubuntu, вы наверняка захотите познакомиться с ним поближе. Вот список стандартных задач и способов их решения...

» **Работа в Web** Нажмите на оранжевую кнопку на верхней панели, чтобы запустить вездесущий *Firefox*.

» **Чат онлайн** Запустите Applications > Internet > Pidgin Instant Messenger [Приложения > Интернет > Pidgin Instant Messenger]. *Pidgin* поддерживает почти все популярные протоколы, в том числе ICQ, Jabber, AIM, Yahoo и MSN.

» **Редактирование документов** Перейдите в Applications > Office [Приложения > Офис], затем выберите текстовый редактор *OpenOffice.org* (под названием *Writer*), редактор электронных таблиц (*Calc*) или презентаций (*Impress*) – каждый из них совместим с документами *Microsoft Office*.

» **Редактирование изображений** Зайдите в Applications > Graphics > Gimp [Приложения > Графика > Gimp], чтобы использовать самое мощное Linux-приложение для редактирования изображений.

» **Воспроизведение музыки и видео** Посетите Applications > Sound & Video [Приложения > Звук и видео] ради знакомства с *Rhythmbox* и *Movie Player*.

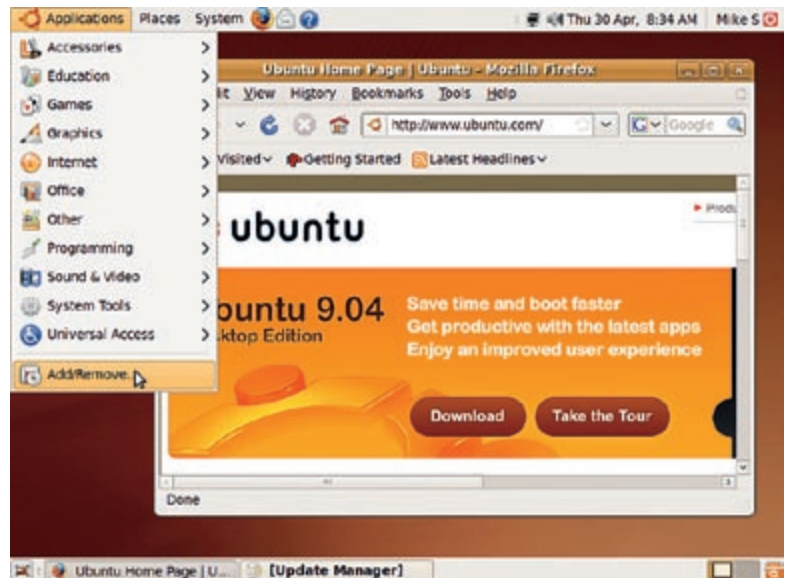
» **Добавление программ** Откройте Applications > Add/Remove [Приложения > Установка и удаление] – вы найдете простой интерфейс для получения дополнительных программ через Интернет; а если вам нужен более продвинутый инструмент, зайдите в System > Administration > Synaptic [Система > Администрирование > Synaptic].

» **Вход в командную строку** Эмулятор терминала доступен через Applications > Accessories > Terminal [Приложения > Дополнительные > Терминал].

» **Настройка системы** В меню System > Preferences [Система > Настройки] и System > Administration [Система > Администрирование] вы найдете утилиты для смены интерфейса, настройки оборудования, управления пользователями и многого другого.

Программа установки Ubuntu отличается простотой, но всегда есть шанс, что появятся проблемы. Если рабочий стол не загружается, попробуйте удалить внешнее оборудование (сканеры или принтеры): они способны вызывать осложнения. Подключить их можно будет потом.

Если вам понадобится помощь онлайн, на www.ubuntu.com/support найдутся указания на документацию и списки ссылки. Можно также зайти на форумы **LXF** на www.linuxforum.ru и собственные форумы Ubuntu на www.ubuntuforums.org. Удачи!



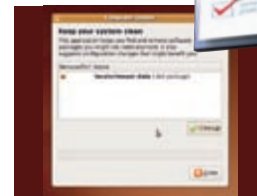
» С годами Ubuntu претерпел метаморфозу, став из грязно-бурого ярко-апельсиновым, что подчеркивает нарядность браузера.

Не пропустите...



Brasero

В меню Applications > Sound & Video [Приложения > Звук и видео] находится *Brasero*, инструмент для записи дисков, который теперь включен по умолчанию.



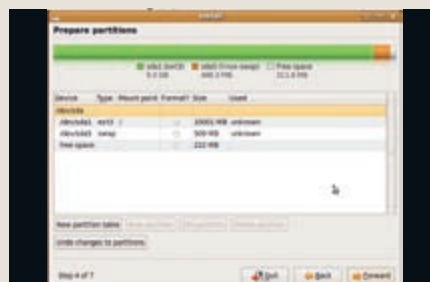
Очистка пакетов

System > Administration > Computer Janitor [Система > Администрирование > Очистка компьютера] поможет удалить ненужные пакеты.



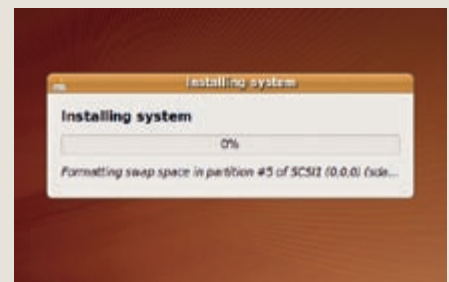
4 Разбиение диска

Затем программа установки предложит поделить жесткий диск на разделы. Можно сжать имеющиеся разделы Windows или Linux, забрать весь диск целиком или разбить его вручную.



5 Разбиение диска вручную

Если вы выбрали ручную разбивку, создайте корневой раздел размером не менее 10 Гб в формате ext3 и 512-Мб раздел подкачки для виртуальной памяти.



6 Учетная запись

Здесь вы создаете пользовательскую учетную запись (в имени и пароле регистр важен!). После этого скопируются файлы Ubuntu, и вы сможете перезагрузиться.

Еще программы

Дистрибутивы, новые приложения, документация

Дистрибутив Linux

Mandriva 2009 Spring

Вот и дошла очередь до Mandriva (бывшего Mandrake): этот дистрибутив привел в мир Linux множество новых пользователей, продемонстрировав им, что переход с Windows вовсе не обязан быть трудным. По сути, Mandriva была одним из пионеров графических программ установки, а ее *Центр управления*, построен-

ный по принципу «все-в-одном», создает истинный комфорт для тех, кого вгоняют в ступор необъятные коллекции инструментов настройки.

Итак, можно уверенно заявлять, что Mandriva – самый дружелюбный к пользователю дистрибутив Linux из ныне существующих: см. наш полный вердикт на стр. 12. Но при всем этом, он вовсе не нацелен только на новичков – его стоит попробовать, если вы уже являетесь по-

же 1,5 ГГц. Чтобы приступить к работе, запустите компьютер, установив в привод LXF DVD, а потом следуйте подсказкам инсталлятора.

Если ваш компьютер не загружается с DVD, откройте [index.html](#) на первой стороне диска в своем браузере и перейдите в раздел *Справка/Новичку в Linux*: там имеются инструкции по изменению порядка загрузки в BIOS. Некоторые компьютеры настроены на запуск с жесткого дис-

ка, а не с оптического привода, но информация на нашем DVD поможет переделать такое поведение.

Дополнительные сведения об этом вы-

«Mandriva отлично поладит с другими ОС на вашем ПК.»

стоянным пользователем Linux и просто хотите сменить систему.

Пам'ятаю о вышесказанном, вы будете рады узнать, что LXF DVD этого месяца – диск с двойной загрузкой: вместе с Ubuntu можно установить и Mandriva 2009 Spring (он же 2009.1) со второй стороны LXF DVD. Для вашего удобства мы разместили на диске Free-редакцию – это полноценный DVD-дистрибутив, но без коммерческих программ и драйверов. Mandriva 2009.1 Free не работает в режиме Live, но для знакомства с системой можно использовать любую виртуальную машину. Минимальные системные требования таковы:

- » 32- или 64-битный процессор Intel/AMD;
- » 256 МБ ОЗУ; рекомендуется 512 МБ;
- » 4 Гб места на жестком диске.

Для базовых задач сойдет 800-МГц процессор, но если надо воспроизводить мультимедийные файлы и играть в игры, рекомендуем процессор не ни-

же 1,5 ГГц. Чтобы приступить к работе, запустите компьютер, установив в привод LXF DVD, а потом следуйте подсказкам инсталлятора.

Мandriva также отлично поладит с другими ОС на вашем ПК. Допустим, у вас стоит Windows; программа установки Mandriva позволит вам сжать раздел Windows (часть диска), чтобы выделить место под файлы Linux, которые должны располагаться на своем разделе. Но перед этим выполните в Windows дефрагментацию диска и сохраните важные данные! После установки, загружаясь с винчестера, вы увидите меню, где предоставляется выбор: запускать Windows или Mandriva.

Учтите, что Mandriva продает версию Powerpack с дополнительными возможностями: программами, мультимедийными кодеками, онлайн-поддержкой и т.п. Загляните на [www.mandriva.ru](#), чтобы узнать больше и, если увиденное вам понравится, приобрести копию.

Настольные приложения

Arista и Gnumeric

Пробуя конвертировать медиа-файлы из одного формата в другой с помощью *Mencoder* в командной строке, вы, небось, рвали на себе волосы при виде жуткого количества непознаваемых опций. А вот небольшой графический инструмент *Arista*, при помощи Gnome и библио-



» Mandriva изобилует приложениями для работы в Интернете, в офисе и для просмотра мультимедийных файлов.

Очень ВАЖНО!

» 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.



» Центр управления Mandriva запускается кнопкой с изображением монитора и красного гаечного ключа.

теки *Gstreamer*, делает перекодировку аудио и видео совершенно пустячным делом. Лучше всего то, что он снабжен уже готовыми настройками форматов для разных устройств: если, к примеру, вы хотите конвертировать видео для iPod Touch или Sony PSP, это потребует лишь пары щелчков мышью.

Тем временем *Gnumeric* подступает к версии 2.0. Промежуточная версия 1.9.8 предлагает улучшенный импорт и экспорт формата OpenDocument (ODF), а также отполированный интерфейс. Будучи предельно простой, она не предназначена для ежедневного использования, но вы сможете сделать релиз 2.0 супер-ровным, если протестируете ее и отправите отчет об ошибках на <http://projects.gnome.org/gnumeric>.

Интернет-приложения

Chromium и Get-Flash-Videos

Google поднял шумиху в web-сообществе, сообщив о выходе *Chrome*, своего собственного браузера на базе *WebKit* (который, в свою очередь, основан на *KHTML* от KDE). Официально *Chrome* пока существует только для Windows, но исходный код доступен в проекте *Chromium*, и мы можем испытать раннюю версию в Linux.

В разделе **Интернет** нашего DVD вы найдете его Deb-пакеты для Debian и Ubuntu и RPM'ы для Fedora, Mandriva и OpenSUSE со товарищи. Информация по установке пакетов есть в разделе **Справка/Новичку в Linux**. Помните, что это не официальная версия *Chrome* для Linux, так что весьма вероятны сбои и ошибки; но, почитав наш материал *Что за штука...* на стр. 42, вы, наверное, захотите узнать, каким видится будущее web-браузеров. Запустив приложение, вы увидите предупреждение разработчиков, что это пре-альфа версия; на данном эта-

пе нечего даже отправлять сообщения об ошибках!

В разделе **Интернет** также представлена удобная утилита Perl под названием *Get-Flash-Videos*. Название все объясняет: она скачивает из сети Flash-видео. Можете использовать ее, чтобы набрать клипов с YouTube, Metacafe, Vimeo и других сайтов и сохранить их на своем компьютере для дальнейшего просмотра: это очень удобно, если у вас не слишком стабильное соединение с Интернетом.

Документация

Подшивка LXF

В выпуске **LXF115** мы опубликовали электронную подшивку Linux Format за первую половину 2008 года. С тех пор прошло целых пять месяцев, и мы рады предложить вашему вниманию недостающие шесть номеров: **LXF107–112**. Данные выпуски в формате PDF можно найти в разделе **Журнал**. Читая их, вы узнаете, как запустить Linux на самых различных устройствах, разберетесь, почему Ричард Столлмен [Richard Stallman] создал движение свободного ПО, научитесь обрабатывать домашние видеозаписи и выйдете к истокам рубрики *сисадмина*. Мы расшифруем для вас сообщения об ошибках, покажем, как производить статистическую обработку данных без дорогостоящих коммерческих пакетов, поможем получить диплом Linux-гуру или с минимумом неудобств сменить дистрибутив. Ну и конечно же, в каждом выпуске вас ждут традиционные обзоры, сравнения, интервью и истории успеха. Пользуясь случаем, мы также приглашаем всех желающих принять участие в «викификации» указанных номеров журнала, то есть переносу их содержимого онлайн на wiki.linuxformat.ru. Это достаточно большая работа, но ее значительно легче выполнить совместными усилиями!

Не забывайте, что все материалы, опубликованные на wiki.linuxformat.ru, подпадают под действие Open Publication License v1.0, то есть переводя материалы в формат вики, вы помогаете не только нам, но и всему российскому Linux-сообществу. Чтобы приступить, просто зарегистрируйтесь по указанному адресу и создайте страницы для материалов, с которыми будете работать.

► Устройтесь поудобнее с чашечкой чая или кофе и подшивкой LXF за вторую половину 2008 года.



И напоследок... Игровой квинтет

Давайте прикинем: кто любит стратегии в реальном времени (RTS)? Мы! Командовать армиями, планировать маневры и идти на штурм — огромное удовольствие, и *Globulation 2* — превосходный образец данного жанра. Вместо того, чтобы слать отдельные боевые единицы прямо в необъятную военную машину, вы управляете их группами, и этот подход намного лучше, поскольку позволяет не погрязнуть в управленческих мелочах. Вместо этого вы концентрируетесь на общей стратегии, и можете играть как против компьютера, так и против других игроков через интернет. Прочтите *Install.txt* в разделе **Игры/Globulation2** нашего диска — это руководство по запуску игры.

Последний раз мы включили *Hedgewars* в раздел **Игры** в выпуске **LXF113/114** — версию 0.9.7. С тех пор в эту замечательную игрушку в стиле *Worms* был вложен серьезный труд. Версия 0.9.11 предоставляет новые карты и локации (например, пещеры); новые темы и музыкальные треки; «вспомогательное» оружие, которое падает ящиками, чтобы увеличить вашу силу, способности, а также множество других хитростей и полезностей. Это от-

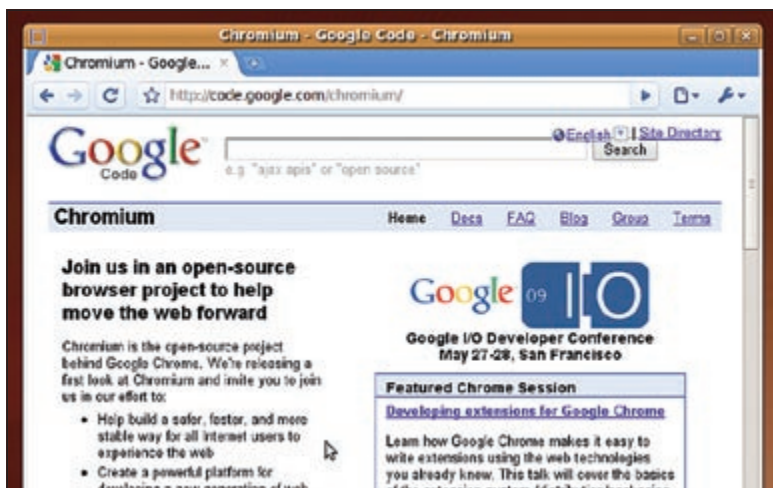


► Создайте себе армию в *Globulation 2*.

лично сделанная и отлаженная игра; она доказывает, что Open Source может успешно конкурировать и по части внешнего вида игр, и в плане увлекательности процесса.

И, наконец, парочка головоломок. *Hexalate* предлагает совмещать цвета на дисках, которые можно двигать и поворачивать. На каждом диске — шесть цветных полос, идущих от центра, и ваша задача — двигать и вращать диски таким образом, чтобы все цвета совпали за минимальное количество сделанных ходов.

A *Minestein*, в свою очередь, являет собой клон классической *Minesweeper*, но включает головоломки, решаемые исключительно логическим путем: никаких вам угадываний. В нем есть система подсказок — это если вы безуспешно скребли себе затылок... **LXF**



► Сможет ли Google покорить мир благодаря *Chrome*? Попробуйте последнюю версию *Chromium* — но пока она годится только для тестирования!



VI Конференция разработчиков свободных программ на ПРОТВЕ

27-29 июля 2009
г. Обнинск
Калужская область

Организатор: Компания ALT Linux
Тел./факс: +7 (495) 662-38-83
www.altlinux.ru, conference@lists.altlinux.org

Приглашаются
слушатели и докладчики!

Темы конференции

- Проекты разработки свободного программного обеспечения
- Научные проекты в различных областях знаний, включающие разработку продуктов под свободной лицензией
- Культурные, философские и правовые особенности свободного лицензирования
- Свободные программы для органов государственной власти



**СИСТЕМНЫЙ
администратор**

Клонируем Windows
с помощью Symantec Ghost

Насколько неуязвима 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



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 любые другие. 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 смогли 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!

Превосходный дистрибутив для новичка в мире Linux

- » **Удобный инсталлятор**
Можно установить систему рядом с другой ОС
- » **KDE 4**
Самый современный рабочий стол в своем лучшем исполнении
- » **Простая настройка**
Центр управления Mandriva предоставляет доступ ко всем параметрам системы из одной точки



Июль 2009
LXF DVD 120

LINUX
FORMAT

Июль 2009
LXF DVD 120

LINUX
FORMAT

Эксплуативная LXF-редакция включает Xfce, KDE, дополнительные приложения и средства разработки

А также:

- » **Подписка Linux Format**
Номера за вторую половину 2008 года в формате PDF
- » **Chromium**
Попробуйте Linux-версию браузера от Google
- » **Пять отличных игр**

ubuntu
9.04



Сторона 2

HOT PICKS

- Ardenryst** – скролл-шутер
- DiscWrapper** – дизайнер обложек для компакт-дисков
- Graysn** – это графический интерфейс для GnuS
- Gudyadeque** – аудиоприоритизатор
- Klavaro** – клавиатурный тренажер
- KnowledgeRoot** – база знаний
- Neverball** – аркадная игра
- Psi** – свободный label-клиент на Qt
- Rawstudio** – RAW-конвертер
- Unit** – интерфейс для Nmap

ИГРЫ

ИГРЫ
Globalization2 — стратегия реального времени
Hedgewars — клон Worms
Hexalate — логическая игра
LordsAWar — пошаговый воергейм
Minestein — версия «Санера»

BRUTE – книга по администрированию GNU/Linux

RUTE — книга по администрированию GNU/Linux

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь по адресу disk@linuxformat.ru

Тираж издан в 000 «Фортмедиа», Россия, Санкт-Петербург, 196006, Цветочная ул., 7, тел. (812) 388-82-90. Лицензия МПТР России ВАР № 77-225.

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 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):

Plexstor= 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 вы 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 кладку *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.

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.

А что, если у вас нет устройства, с помощью которого можно было 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.

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 Линукс-центр 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:

LXF117

Апрель 2009

- » Чиним Linux: типовые проблемы и пути их решения.
- » GnuPG: на страже ваших данных.
- » Lemote Mini PC: китайский народный компьютер.
- » LyX: процессор документов, объединяющий в себе мощь LaTeX и удобство OpenOffice.org.

LXFDVD: Knoppix 6, Foresight Linux 2.1 и набор исследователя открытых ОС (15 шт.)

Печатная версия:

http://www.linuxcenter.ru/shop/books-and-magazines/Linux-Format/lxf_117/

Электронная версия в формате PDF:

http://www.linuxcenter.ru/shop/electr/magazine/elxf_117/



LXF118

Май 2009

- » Ищем идеальный дистрибутив Linux.
- » Cuneiform и Tesseract: две ведущие открытые OCR-системы.
- » Спутниковое ТВ: как настроить его в Linux и смотреть любимые передачи в Kaffeine, VLC и MPlayer.

LXFDVD: PCLinuxOS 2009.1, CentOS 5.3, Clonezilla 1.2 и SystemRescueCd 1.1

Печатная версия:

http://www.linuxcenter.ru/shop/books-and-magazines/Linux-Format/lxf_118/

Электронная версия в формате PDF:

http://www.linuxcenter.ru/shop/electr/magazine/elxf_118/



LXF119

Июнь 2009

- » Будем продуктивнее: новый взгляд на знаковые приложения.
- » Slack в будущее: в чем изюминка одного из старейших дистрибутивов Linux?
- » Moblin: мобильная Linux-платформа от Intel с точки зрения пользователя и программиста.
- » Java FX: новая технология от Sun Microsystems обещает легкий путь в мир богатых интернет-приложений.

LXFDVD: SimplyMEPIS 8.0, Slackware 12/Zenwalk 6.0, FreeBSD 7.2, OpenBSD/BSDanywhere 4.5 и NetBSD 5.0

Печатная версия:

http://www.linuxcenter.ru/shop/books-and-magazines/Linux-Format/lxf_119/

Электронная версия в формате PDF:

http://www.linuxcenter.ru/shop/electr/magazine/elxf_119/



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!

LINUX FORMAT

Главное в мире Linux

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 Авакумов, Александр Бикмеев, Юлия Дронова, Светлана Кривошеина, Александр Казанцев, Алексей Опарин, Валентин Развозжаев, Татьяна Цыганова

Редактор диска

Александр Кузьменков

Верстка, доредакционная подготовка

Сергей Рогожников

Креативный директор

Станислав Медведев

Технический директор

Денис Филиппов

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатели

Виктор Федосеев, Павел Фролов

Отпечатано в типографии ОАО «СПбГК»

198216, Санкт-Петербург, Ленинский пр., 139

Заказ 2142

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Пол Хадсон [Paul Hudson] paul.hudson@futurenet.co.uk

Редактор обзоров Грэм Моррисон [Graham Morrison]

graham.morrison@futurenet.co.uk

Редактор диска Майк Сондерс [Mike Saunders] mike.saunders@futurenet.co.uk

Художественный редактор Эфраин Эрнандес-Мендоса

[Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.co.uk

Литературный редактор Эндрю Грегори [Andrew Gregory]

agregory@futurenet.co.uk

Подготовка материалов

Нейл Ботвик [Neil Bothwick], Макс Брукс [Max Brooks], Крис Браун [Chris Brown], Энди Ченнел [Andy Channell], Мэттью Кляпхем [Matthew Clapham], Алекс Кокс [Alex Cox], Майкл Дж. Хэмел [Michael J. Hammett], Энди Хадсон [Andy Hudson], Джульетта Кемп [Juliet Kemp], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Евгений Балдин, Виталий Бахтинов, Андрей Боровский, Николай Игotti, Николай Колдыряев, Евгений Крестников, Григорий Рудницкий, Алексей Федорчук, Антон Черноусов

Иллюстрации: Крис Хидли [Cris Hedley], Крис Винн [Chris Winn], iStock Photo

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Tel +44 01225 442244 Email: linuxformat@futurenet.co.uk

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, являющиеся 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.

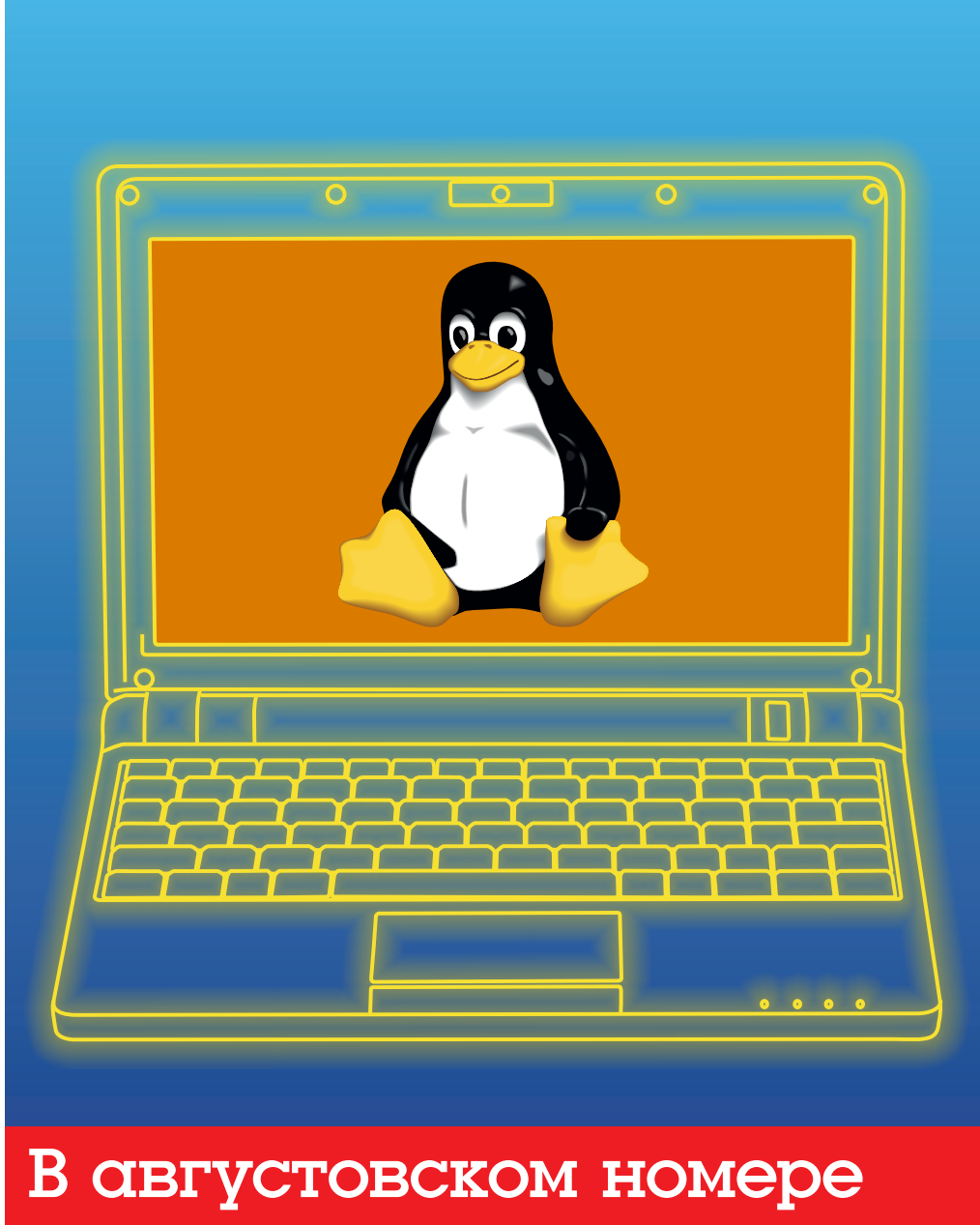
Все присланные материалы могут быть помещены на CD или DVD-диски, поставляемые вместе с журналом, если не было оговорено иное.

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 [Linux 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>



В августовском номере

Тестирование нетбуков

Мы заперли Майка Сондерса в комнате, впахнув туда все нетбуки, которые только смогли найти – пусть он один правит всеми.

Проверка боем

Наше пошаговое руководство поможет найти и закрыть наиболее распространенные уязвимости Linux. А насколько защищена ваша система?

Ulteo: сетевой рабочий стол

Спустя три года после увольнения из Mandriva, Гаэль Дюваль довел свой новый Linux-проект до состояния готовности к использованию: мы посмотрим на него вблизи.

Содержание последующих выпусков может меняться без уведомления.



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

GNU/Linuxcenter приглашает за покупками!

В магазинах ГНУ/Линуксцентра всегда большой выбор товаров, интересных сторонникам свободного программного обеспечения: дистрибутивы GNU/Linux и BSD, устройства со встроенным GNU/Linux, журналы и литература по свободному программному обеспечению, а также большое количество атрибутики: футболки, кружки, кепки, значки, игрушки и другие нужные вещи.

Москва

Краснознаменская ул., 17 (метро «Авиамоторная»),
в помещении АТС МЭИ
Тел. 8 (499) 271-4954

Санкт-Петербург

Лиговский пр., 50, корп. 15 (метро «Площадь Восстания»)
Тел. 8 (812) 309-0686

Интернет: www.linuxcenter.ru

**Ждем вас с понедельника по пятницу с 9:30 до 20:00
в фирменных магазинах ГНУ/Линуксцентра!**

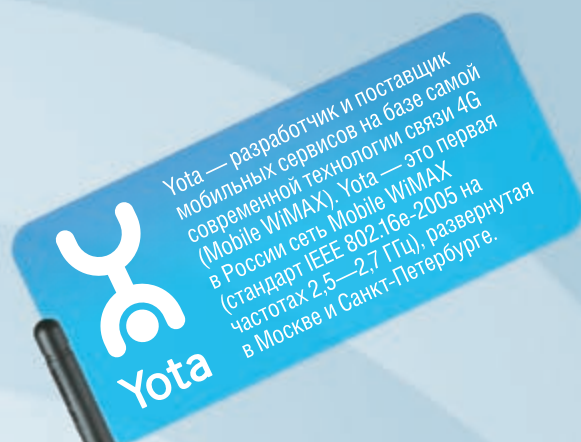
ИНТЕРНЕТ-РЕШЕНИЕ
ОТ WIMAXSTORE

Mobile WiMax

ИНТЕРНЕТ
НОВЕЙШЕГО
ПОКОЛЕНИЯ 4G

Интернет WiMax — это:

- Высокоскоростной доступ — до 10 Мбит/с в любой точке зоны покрытия
- Связь в тех местах, которые раньше были вне досягаемости широкополосного доступа
- Легкое и быстрое подключение
- Не требуется прокладки проводов и, соответственно, дополнительных инвестиций
- Triple play: возможность интеграции разнородного трафика (данные, голос, видео)
- Круглосуточная техническая поддержка без выходных



1 + РОУТЕР ASUS WL 500GP
МОДЕМ SAMSUNG SWC-U200
6940 руб.



2 + РОУТЕР D-LINK DIR-320
МОДЕМ SAMSUNG SWC-U200
5490 руб.

 **wimaxstore**

САНКТ-ПЕТЕРБУРГ
+7 812 309-06-86

МОСКВА
+7 499 271-49-54

WWW.WIMAXSTORE.RU

Технические специалисты WiMax Store разработают для вас заказные
программно-аппаратные решения любой сложности на базе WiMax. Звоните сегодня!

Электронная копия журнала Linux Format. Полное распространение осуществляется по закону РФ. Заказ LC126034. Подписан: Исламов Дамир Ревинирович, email: damir@trefle.ru
Цены указаны с учетом НДС.