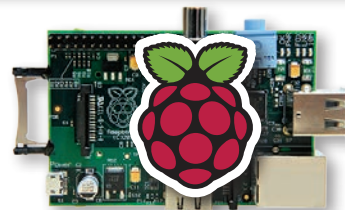




LINUX FORMAT

Главное в мире Linux

Август 2013 № 8 (173)



Родитель Pi

« Во мне недостаточно альтруизма, чтобы делиться своей IP »
Эбен Аптон — прагматик

Сбежим от Google и от Facebook » Twitter » Flickr и прочих!



Вырвите свои личные данные у компаний, которые норовят украсть ваши письма, фотки и друзей и вторгнуться в вашу жизнь

Поиск в Google

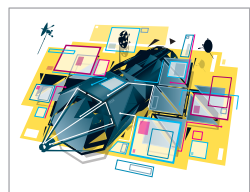
Пойду-ка я отсюда

Kdenlive Чем дышит кинематограф
Blender Лихо моделируем трехмерку



ПЛЮС!
Летопись оконных менеджеров
с. 56

Debian 7.0
» Разбираемся, почему 8 из 10 сисадминов* используют этот дистрибутив с. 16



Mir
» Серверы графики выходят на орбиту с заменой Wayland от Ubuntu с. 60

Финты Libre Office
» Внесем в документы стиль с. 68

Также в номере...

Внутри EFF
Супергерои, которые защищают нас и наши данные с. 42



KDE 5.0
Чего ждать от свежей версии взломщика парадигм с. 46



Сисадминам
Навеки покончим споры о подкачке с. 62



Raspberry Pi Raspbian
» Установим версию Debian, заточенную под Pi

Администрирование
Рулим объемами
» Велим пользователям не шалить на диске

Обработка данных
MariaDB
» Масштабируем базу данных скорости и безопасности ради

ПОДПИСНЫЕ ИНДЕКСЫ В КАТАЛОГАХ
Агентство «Роспечать» — 36343
«Почта России» — 11932, «Пресса России» — 90959

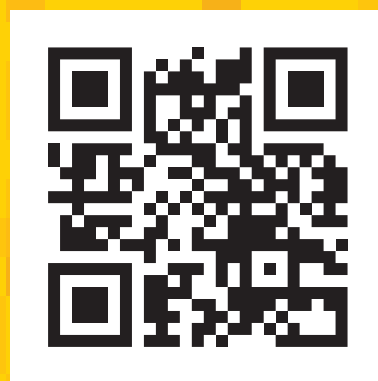


АИШ.ИТЭ

ОФИЦИАЛЬНОЕ ИЗДАНИЕ

theRunet

Бизнес. Медиа. Будущее.



23-24-25 ОКТЯБРЯ 2013 ГОДА

РЕКЛАМА

18+

Добро пожаловать в робототехнику!

ScratchDuino

Электронный комплекс на основе свободного аппаратного обеспечения для школ и вузов

ScratchDuino.Лаборатория

Плата расширения с датчиками ввода-вывода — как установленными на плате, так и подключаемыми. Предназначена для программирования взаимодействия компьютера с внешними устройствами.

Полная интеграция («из коробки») со средой программирования Scratch, предназначенной для детей.

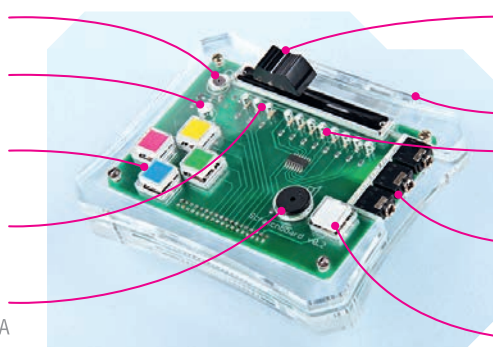
ДАТЧИК ЗВУКА*

ДАТЧИК ОСВЕЩЕННОСТИ*

КРЕСТОВИНА КНОПОК

ЦВЕТНЫЕ СВЕТОДИОДЫ

ДИНАМИК ДЛЯ ВЫВОДА ЗВУКА



ПЕРЕМЕННЫЙ РЕЗИСТОР (ПОЛЗУНОК)*

КАРТРИДЖ ARDUINO

СВЕТОДИОДЫ, УПРАВЛЯЕМЫЕ ЧЕРЕЗ РАЗРЯДНУЮ МАТРИЦУ

РАЗЪЕМЫ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ

КНОПКА*

* РАБОТАЮТ В РЕЖИМЕ ЭМУЛЯЦИИ RISCBOARD, НЕ ТРЕБУЯ ДОПОЛНИТЕЛЬНОГО ПРОГРАММИРОВАНИЯ.

ЦЕНА **5000 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

ScratchDuino.Робоплатформа

Внешний робот-исполнитель, управляемый из среды программирования Scratch. Не требует навыков программирования на языках высокого уровня и может применяться для обучения, начиная с младших классов.

ScratchDuino.Робоплатформа

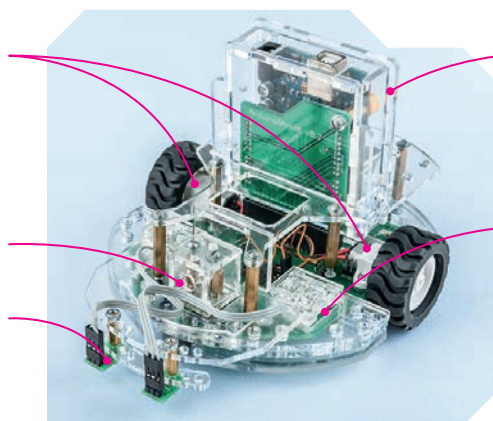
обеспечивает возможность:

- подключения внешних датчиков без механических соединений и пайки;
- расширения через последовательную шину RoboBus;
- установки деталей Lego Technics.

МОТОР-РЕДУКТОРЫ

ФОТОДАТЧИК

ДАТЧИК ЛИНИИ



КАРТРИДЖ ARDUINO

МАГНИТНЫЙ ЦИФРО-АНАЛОГОВЫЙ РАЗЪЕМ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ БЕЗ МЕХАНИЧЕСКИХ СОЕДИНЕНИЙ И ПАЙКИ*

* ВСЕГО НА РОБОПЛАТФОРМЕ 5 РАЗЪЕМОВ. ВОЗМОЖНО ПОДКЛЮЧЕНИЕ ДАТЧИКОВ КАСАНИЙ, ИК-ДАТЧИКОВ РАССТОЯНИЙ/ПРЕПЯТСТВИЙ И ДР.

ЦЕНА **10 000 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

Назначение ScratchDuino

- Изучение взаимодействия компьютера с внешней средой
- Изучение изменения параметров внешней среды
- Изучение процессов передачи информации и принципов ее построения
- Изучение внешних устройств управления
- Моделирование устройств

Среда разработки

Lazarus (язык Pascal) ■ Scratch ■ Arduino IDE

ScratchDuino адаптирован для учебных заведений. Поставляется с комплектами учебно-методических материалов.

Продукт разработан при финансовой поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере

WWW.SCRATCHDUINO.RU

(812) 309-0686 (Санкт-Петербург) ■ (499) 271-4954 (Москва)

23-25 октября, 2013 г. Москва. Центр Digital October

Приглашаем

Аудитория:
до 800 делегатов
из 250+ компаний
и 20+ стран

Приз 1000 Евро
за лучший
исследовательский
доклад

Бесплатное
участие для
докладчиков

- Программистов и инженеров качества
- Системных аналитиков и архитекторов
- Лидеров команд и менеджеров проектов
- HR-специалистов и руководителей производства
- Исследователей, студентов, аспирантов
- Продуктовые компании, компании, реализующие ПО «в облаке»
- Центры разработки и филиалы транснациональных компаний
- Аутсорсинговые компании и ИТ-департаменты государственных учреждений

Подробнее: www.secr.ru, contact@secr.ru, +7 812 336 93 44



Спонсоры



Партнеры



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

В данном номере мы рассмотрели способ заменить Google собственным сервисом. Мы спросили команду, как бы они этот сервис назвали.



Крис Торнетт

Нет ничего хуже, чем названия релизов Android. Кроме разве что релизов Ubuntu.



Эндрю Грегори

Коли я пообещал для этого номера учебник про ТВ-контроль за домашними животными, пусть будет Poole.



Эфраин Эрнандес-Мендоса

Мой — Энтские жены. Он может найти все и принесет мир в Лес Фангорна.



Бен Эверард

Я бы построил паромеханический движок и назвал его Найдите Разницу.



Маянк Шарма

Я так подсел на Portal, что назову свой Mortal. Он будет заниматься поиском поисков.



Джонатан Робертс

Я бы назвал его Virtual Ethics и использовал для поисков ответов на моральные дилеммы.



Майк Сондерс

Я бы создал поисковый движок для славного города Мюнхена и его классной поддержки Linux.



Валентин Синицын

Voogle. С жутковатым дизайном, цепями, стеновыми и кнопкой «Мне не повезет».



Ник Вейч

Уж не знаю, как бы я назвал сам движок, но девиз его будет «Делай зло».



Сюзан Линтон

У меня уже и так есть успешный портал — это чудесный tuxmachines.org. Зашли бы на огонек...



Шашанк Шарма

Я бакалавр права, и мой движок бесплатно давал бы юридические консультации проекта открытого кода.



Нейл Ботвик

Хотел было сказать «DuckDuckGo», но полез в Google и увидел, что это имя уже занято. Подумаю.



Два сисадмина

» Оба — американцы. На момент «зенита славы» обоим не было и 30, хотя по возрасту первый годится второму в отцы. И фамилии обоих начинаются на букву «С».

Первого зовут Клиффорд Столл [Clifford Stoll]. В далеком 1986 году он, будучи админим лабораторией имени Лоуренса в университете Беркли, обнаружил взлом десятков серверов оборонных подрядчиков и вышел на след немецкого хакера Маркуса Гесса [Markus Hess], работавшего на КГБ. Причем все началось с расследования недостачи оплаты использованного машинного времени, размером в 75 центов.

Эти события стали первым доказанным случаем кибершпионажа и подробно описаны самим Клиффом в книге «Яйцо кукушки» [The Cuckoo's Egg, 1990]. Вышедший в 1997 году русский перевод давно стал библиографической редкостью.

Самое интересное в этой истории то, что все расследование Клифф вел самостоятельно, так и не убедив представителей многочисленных спецслужб в важности происходящего. Полковники просто не хотели слушать «этого гражданского», вчерашнего студента, толкующего им про угрозы из какой-то непонятной Сети.

Второй сисадмин — Эдвард Сноуден [Edward Snowden] — в этом году сообщил миру о широкомасштабных операциях американских спецслужб по перехвату интернет-трафика.

Значит, полковники все-таки услышали Столла. Что делает тему номера особенно актуальной.

Кирилл Степанов

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

Общие вопросы: info@linuxformat.ru

Проблемы с дисками: disks@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

Сайт: www.linuxformat.ru, группа «ВКонтакте»: vk.com/linuxform

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

» **Телефон редакции:** (812) 309-0686. Дополнительная информация на с. 112

Содержание

Давайте мы вам все нагуглим...

Обзоры

Debian 7 16

12 месяцев после заморозки функций... добьется ли успеха этот оплот стабильности?



› Раз уж Линус Торвалдс вернулся на Gnome 3, почему бы не попробовать Debian?

Mint 15 18

Чувствуем, нас скоро начнут попрекать, что журнал превращается в *Mint Format*...



› Наш раздел Обзоры заделался дистрибутивом, и король его — Mint!

Mageia 3 19

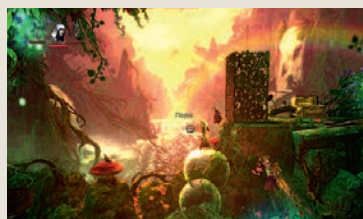
Как новый альбом Daft Punk, данный французский товар возрождает старый жанр.

Gizmosphere 20

Крошечная микроконтроллерная плата, но не на базе ARM. Это x86 до мозга костей.

Trine 2 21

В этой игре три ваши ипостаси — рыцарь, маг и вор, и все в одном персонаже. Каким лицом вернуться к врагу?



› Портреты персонажей расставлены на полочках. Выбирайте по обстановке.

Сбежим от Google

Защитите свою конфиденциальность с помощью нашего гида по миграции...

Поиск в Google

Мне повезет

с. 30

Сравнение: Производные Ubuntu

с. 24

Люди говорят



«Было бы печально и даже фатально для нас продавать в 2016-м все тот же Raspberry Pi»»

Эбен Аптон — о том, что делать после победы с. 38

На вашем бесплатном DVD



Debian 7

» Сверхстабильный дистрибутив

Mageia + Sabayon

» Простота или инновации — решать вам

ПЛЮС: HotPicks и коды учебников **с. 106**



Ищите в этом номере



Что такое EFF 42

Внутри Electronic Frontier Foundation.

KDE нового поколения 46

Расслабьтесь: KDE 5 не будет!

Феномен Ubuntu 50

Откуда же все это взялось?

Mir 60

Замена *Wayland* от Canonical.



Пропустили номер?

Узнайте на с. 108, как получить его прямо сейчас!



Учебники

Офисные приложения	
Будем стильными	68
Как придать своей документации элегантный вид.	
Raspberry Pi	
Мастер загрузки	70
Записываем образы диска и ищем неисправности.	
Администрирование	
Квоты пользователей	74
Приглажайте за фильмотекой своих родных.	
Базы данных	
MariaDB	76
Как сделать свою базу данных больше, быстрее и мощнее.	
Обработка в GPU	
CUDA	80
Перекинем математические расчеты на графический процессор.	
Программирование	
Erlang	84
Практика, практика и еще раз практика — вот путь джедая.	
Обработка видео	
Kdenlive	88
Если вы не стали Стивеном Спилбергом, недостаток финансирования тут ни при чем.	
3D-моделирование	
Blender	92
Заморим низкополигонального червячка. А от него и до человека недалеко.	

Постоянные рубрики

Новости	6	Интервью LXF	38	Hotpicks	100
«Пингвин Софтвр» занялся мониторингом, на свободный стандарт переходит Австралия, телефоны с Firefox OS двинулись в Европу, Ubuntu попал в школы Америки, а Canonical — за <i>Mir</i> .		Триумфатор <i>Эбен Аптон</i> не намерен сдавать позиций и останавливаться на достигнутом.		Отведajte горяченького: лучшие в мире новинки свободного ПО. <i>Ascii Design, Converseen, Cosmoasmash, Crayonizer, Ncdu, RText, QMentat, TMSU, Tower Toppler, Typer, ZedLog</i> — что бы значили эти странные слова?	
Электронное будущее! ...	13	Что за штука	60	Диск Linux Format	106
Алтайский край решительно намерен задавать тренды развития.		<i>Mir</i> во всем мире — таковы глобальные планы Canonical. А потому что конвергенция.		Содержимое двустороннего DVD этого месяца.	
Новости Android	22	Рубрика сисадмина	62	Пропустили номер? ...	108
Невзирая на вирусы, бытовая техника устремляется в Интернет, а Google наделяет смартфон Motorola смекалкой.		Доктор играет с огнем и запускает <i>Ntar</i> , не выяснив, нужен ли вам большой раздел подкачки.		Коллекция неполна? Если на вашей полке с журналами образовалась ниша размером с <i>LXF</i> , вы знаете, куда обратиться.	
Сравнение	24	Ответы	96	Через месяц	112
Подумываете сменить Ubuntu? В нашем Сравнении рассмотрены лучшие альтернативы.		ВАШИ ПРОБЛЕМЫ РЕШЕНЫ! <i>Нейл Ботвик</i> — про образы диска, LVM, <i>/proc</i> , форматирование SSD, виртуальные машины и проблемы зависимости и прочее.		Какую замечательную информацию доставит <i>LXF174</i> ?	



ГЛАВНОЕ Российское СПО » Австралия, вперед! » Начало продаж Firefox OS
» Ноутбуки для школьников » Canonical сжигает мосты

НАШИ ЛЮДИ В СПО

«ПингВин Софтвер» как инноватор



» Рубрику готовил
АРТЕМ ЗОРИН

Первый российский участник Linux Foundation предложил новые решения.

За последние четыре года российский рынок СПО значительно вырос, а его участники стали предлагать новые, интересные и эффективные решения для бизнеса. Успешный бизнес невозможен без полного контроля функционирования ИТ-инфраструктуры предприятия. Как построить для этого эффективную систему, минимизировав затраты и не проигрывая в качестве? Использование решений на основе платформ с открытым исходным кодом — набирающий популярность вариант обеспечения комплексного мониторинга за разумные деньги.

Построению подобной системы был посвящен вебинар, проведенный 4 июля 2013 года компанией «ПингВин Софтвер».

Созданная в 2009 году, «ПингВин Софтвер» является постоянным и первым российским членом Linux Foundation.

«Успешный бизнес невозможен без контроля ИТ-структуры.»

Эксперт «ПингВин Софтвер» Михаил Логинов в ходе вебинара отметил: ИТ-инфраструктура большинства компаний построена по общепринятым стандартам, поэтому для большинства задач выгодно использовать типовое решение. Разработанная компанией «ПингВин Софтвер» универсальная система мониторинга ИТ-инфраструктуры «PWS ИТ-мониторинг», построенная методом доработки и интеграции двух известных платформ с открытым кодом — Zabbix и Pentaho, позволяет контролировать типовые объекты

в составе ИТ-инфраструктуры организации: производительность серверов, web-приложений, баз данных, работоспособность сетевого оборудования и многое другое.

Предлагаемое решение работает в гетерогенных средах, не ограничено типами оборудования, обладает большим потенциалом масштабирования. Слушателей вебинара также заинтересовала экономическая сторона, поскольку решение «ПингВин Софтвер», благодаря отсутствию лицензионных отчислений за право пользования и обновления системы, позволяет существенно сократить расходы на реализацию задачи обеспечения мониторинга и оптимизировать работу ИТ-служб.

Не были обойдены вниманием также и вопросы внедрения и работы с системой: требования к «железу», инструментам генерации шаблонов, стабильности мониторинга серверов и т.д. Специалисты сошлись во мнении, что функциональность идентична предоставляемой проприетарными решениями, тогда

как стоимость владения «PWS ИТ-мониторинг» обойдется предприятиям значительно дешевле, что, несомненно, важно для бизнеса.

Так что же предлагает «PWS Мониторинг»? Среди возможностей и преимуществ универсальной системы мониторинга ИТ-инфраструктуры «PWS Мониторинг», предлагаемой компанией «ПингВин Софтвер», можно отметить следующие:

» практически неограниченный спектр оборудования и приложений, которые



» Вот уже четыре года «ПингВин Софтвер» трудится на отечественном рынке СПО. И вполне успешно.

поддаются контролю, а также широчайший диапазон контролируемых параметров;

» гибкий механизм настройки правил контроля, позволяющий реализовать механизм интеллектуального мониторинга. При настройке правил имеется возможность задать очередности и подчиненности проверок — например, если система отследит сбой некоторого головного устройства, она не будет проверять работоспособность подчиненных и отправлять оповещения об их отказе, а уведомит ответственное лицо о сбое головного устройства;

» реализуемая Pentaho аналитическая часть позволяет проводить интеллектуальный анализ, выстраивать прогнозы на основании данных предыстории и в целом подготавливать отчетность, необходимую для принятия управленческих решений;

» ни один из компонентов предлагаемого решения не подразумевает отчисления лицензионных платежей.

На вебинаре были также рассмотрены практические аспекты работы с универсальной системой мониторинга ИТ-инфраструктуры «PWS Мониторинг».

Более подробную информацию о ходе вебинара см. на сайте компании www.pingwinsoft.ru.

“Unix прост. Но надо быть гением,
чтобы понять его простоту.” (с) Д.Ритчи

Учим на гениев!



Академия Информационных Систем

Ждем технических специалистов!



По всем интересующим вопросам обращайтесь:
Академия Информационных Систем
Тел: +7(495) 231-30 49
E-mail: edu@infosystems.ru
www.infosystems.ru

За рубежом переходят на СПО

Перевод жителей Мюнхена с Windows XP на Linux и австралийский демарш.

Успешно осуществив миграцию госучреждений на Linux, Городской совет Мюнхена не остановился на достигнутом и выступил с инициативой по предоставлению всем жителям города инструментов для перехода с Windows XP на Linux. В соответствии с планом, прошедшим стадию предварительного одобрения, до весны следующего года планируется распространить около 2000 дисков с дистрибутивом Lubuntu, редакцией Ubuntu Linux на основе

аппаратуре. В качестве одной из главных целей миграции называется сокращение электронных отходов — если на системах, не удовлетворяющих спецификациям Windows 7 и Windows 8, будет установлена менее требовательная к ресурсам операционная система, они прослужат еще немало лет, и удастся избежать утилизации вполне работоспособных машин.

Инициаторы акции намерены предложить всем желающим альтернативное решение на базе Linux, которое позволит сохранить в работе текущее оборудование и не потребует вложения дополнительных средств на его модернизацию и покупку лицензии на более новую версию Windows. Lubuntu будет поставляться в форме LiveCD, что позволит каждому заинтересовавшемуся горожанину предварительно протестировать применимость новой системы для своих задач, прежде чем принимать решение по миграции с Windows XP на Linux. Кроме низких системных требований Lubuntu (256 МБ ОЗУ), одним из критериев выбора данного дистрибутива стал привычный для пользователей Windows XP классический интерфейс.

Компакт-диски будут выдаваться в рамках специально организованных мероприятий, которые планируется провести в начале следующего года. Кроме раздачи дисков, на данных мероприятиях будут объясняться положительные аспекты использования Linux и демонстрироваться приемы миграции. На выпуск дисков и печать сопутствующих руководств планируется потратить около €4000. В настоящее время рассматриваются способы снижения затрат за счет сотрудничества с местными предприятиями, например, проведения промо-акций.

Все это происходит на фоне постепенного отказа большинства развитых стран от использования проприетарных форматов документов. Так, правительство Австралии отменило решение, принятое в 2011 году, требующее от государственных учреждений использование в своем документообороте проприетарного формата Microsoft Open XML.

Технический специалист австралийского правительства Джон Шеридан [John Sheridan] написал в своем блоге, что принятие формата ODF в качестве стандарта согласуется с политикой Совета Европы (хотя — где Европа и где Австралия?), которая требует от стран-участников Евросоюза использование общих открытых стандартов документооборота.

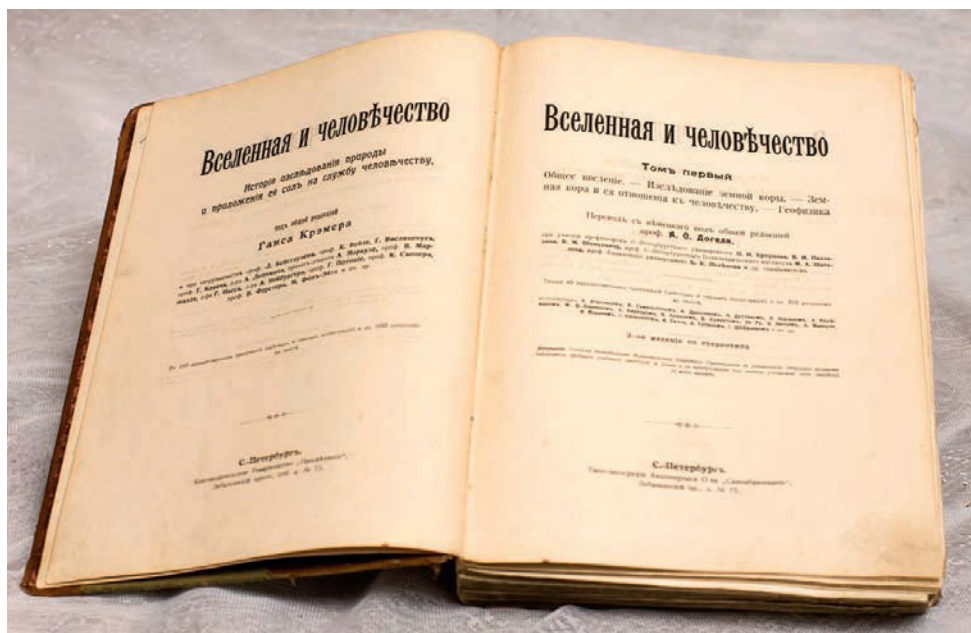
Шеридан говорит: несмотря на то, что пока большинство государственных органов Австралии используют *Microsoft Office*, принятие ODF как стандарта документов по умолчанию не отразится на производительности госорганов, так как все современные версии офисных пакетов от Microsoft обеспечивают встроенную поддержку ODF (начиная с *MS Office 2007 SP2*). Это хорошая новость для австралийцев. Ведь в будущем планируется полный отказ от проприетарных офисных пакетов и переход на *LibreOffice*, что позволит сэкономить значительные средства налогоплательщиков, которые сейчас тратятся на лицензионные отчисления в пользу Microsoft.

По словам Джона Шеридана, историки будущего должны иметь право читать документы в истинно открытом формате, каковым является ODF, поскольку у проприетарных форматов документов зачастую отсутствует обратная совместимость, что затрудняет их открытие и чтение.

«Политика Совета Европы требует общих открытых стандартов.»

окружения LXDE, ориентированной на использование в устаревших системах.

В настоящее время в Германии в эксплуатации находится приблизительно 20 млн компьютеров, на которых установлена ОС Windows XP, выпущенная 11 лет назад. Время выпуска обновлений для Windows XP истекает в апреле следующего года; при этом пользователи данной системы столкнутся с проблемой перехода на более новую версию Windows, которая не будет работать на устаревшей



► Благодаря открытым форматам документов у наших потомков не возникнет проблем с изучением истории. Прямо как по бумажным книгам.

softline®



Services Software Cloud

ИТ-архитектура вашего бизнеса



ПУСТИТЕ В ЕВРОПУ

Firefox OS ищет рынки

Анонсировано начало продаж в Польше, Германии, Греции и Венгрии.

Спустя два года после старта, проект Firefox OS начал приносить первые плоды. 15 июля 2013 года компания Deutsche Telekom объявила о начале продаж смартфонов на базе Firefox OS в Польше, Германии, Греции и Венгрии. Это модель Alcatel One Touch Fire, с 3,5-дюймовым сенсорным экраном (HVGA, 480×320), 1-ГГц CPU Qualcomm Snapdragon MSM7225A, 512 МБ ОЗУ, 512 МБ Flash и камерой на 3,2-Мпикс. Сотовый оператор Movistar (Telefonica) в Испании официально предлагает аппарат ZTE Open с процессором Snapdragon на 1 ГГц, 256 МБ оперативной памяти, 4 ГБ флэш-памяти, 3,5-дюймовым дисплеем, HSDPA, 3,2-Мпикс камерой, Bluetooth, A-GPS и Wi-Fi без контракта за €69, причем €30 поступает на счет абонента. Доступен и новый выпуск симулятора Firefox OS Simulator 4.0, изменения в котором позволяют предоставлять средства тестирования распространения платных при-

ложений через каталог-магазин Firefox Marketplace.

Deutsche Telekom — один из участников-основателей группы компаний Ubuntu Carrier Advisory Group (CAG) по содействию разработке и продвижению мобильной платформы Ubuntu Touch. Но это не мешает продвижению иных платформ.

Firefox OS — первая открытая ОС на ядре Linux, на которой нативно запускаются приложения HTML5. Все программы (звонки, SMS, почта) на телефоне — это web-приложения. В апреле этого года ограниченную партию экспериментальных телефонов для web-разработчиков раскупили за пару часов, так что у Firefox OS неплохие перспективы. ОС позиционируется как более дружественная альтернатива Android, не требующая лицензионных отчислений, которые, даже будучи всего по \$5–10 за устройство, выливаются в существенный процент от стоимости смартфона начального уровня.

➤ Недорогие телефоны всегда имели успех в развивающихся странах. Теперь им предстоит покорить развитые.



Продавать HTML5-смартфоны пожелали уже 18 операторов сотовой связи по всему миру, включая два российских. Mozilla рассчитывает на рынки Центральной и Восточной Европы, Азии и Латинской Америки, где очень популярны недорогие Android-устройства. Крупнейший производитель бытовой электроники, компания Foxconn, в рамках стратегического альянса с Mozilla планирует нанять 2–3 тыс. сотрудников на разработку Firefox OS и создание облачных сервисов для мобильных устройств на базе этой ОС.

UBUNTU В ДЕЛЕ

1700 ноутбуков для школьников

В школе Пенсильвании бесплатно раздают компьютеры с Ubuntu.

Чарли Рейзингер [Charlie Reisinger], руководитель ИТ-департамента средней школы округа Пенн-Мэнор [Penn Manor] в Ланкастере, штат Пенсильвания (США) сообщил, что его учебное заведение использует в своей работе около 1800 компьютеров с Ubuntu на борту. А недавно запущен проект «Один ноутбук в одни руки», и учащимся раздадут 1700 компьютеров с СПО. Администрация образовательного учреждения считает, что это позволит школьникам глубже изучать образовательные предметы и стать лучше подготовленными к жизни. Проект сначала получил одобрение у комиссии, состоящей из учителей, администраторов и членов школьного совета. 1 апреля 2013 года, после дискуссий и презентаций, началась пилотная стадия. Первым выдали ноутбуки студентам Open Campus, отделения по изучению СПО; все остальные их получают до января 2014 года.

По времени проект совпал с плановой заменой компьютеров по гранту

правительства США. В 2007–2008 гг. средняя школа округа Пенн-Мэнор получила \$471 000 на приобретение компьютеров, срок жизни которых скоро заканчивается.

За эти годы мир СПО изменился к лучшему, и цены на ноутбуки снизились, что позволило проекту «Один ноутбук в одни руки» оптимально потратить \$578 000, закупив 1800 компьютеров для школы и 1700 ноутбуков для учащихся.

Опираясь на прошлые успешные установки Ubuntu, было решено устанавливать на ноутбуки школьников свой вариант Ubuntu под названием Ubermix,



➤ Чарли Рейзингер (первый справа) и его коллеги радуются успеху проекта.

разработанный специально для образования Джимом Клейном [Jim Klein], преподавателем средней школы Согус Юнион (Калифорния). Ubermix включает более 60 образовательных приложений. ИТ-отдел в Пенн-Мэнор увеличил число программ и улучшил интеграцию с существующими сервисами учреждения. Ubermix уникален тем, что документы и файлы хранятся отдельно от ОС, на особом разделе диска. Если что-то пойдет не так, учащийся легко восстановит систему с помощью штатных средств ОС, не потеряв свои данные.

У проекта «Один ноутбук в одни руки» эффективная техподдержка: эта задача решается командой студентов-волонтеров, чьи знания могут помочь их сверстникам. Многие из волонтеров уже работали под руководством специалистов из ИТ-отдела школы в рамках летней стажировки, и получили глубокие знания и навыки работы в реальном проекте, чтобы в будущем применять их в своей профессиональной деятельности.

15–16 ОКТЯБРЯ САНКТ-ПЕТЕРБУРГ



IX ПРАКТИЧЕСКАЯ
КОНФЕРЕНЦИЯ
ПО HR-БРЕНДИНГУ

РАЗБИВАЕМ СТЕРЕОТИПЫ

Осенняя конференция HeadHunter традиционно имеет практический, интерактивный, острый формат. В этот раз мы хотим обсудить ряд стереотипов, которые мешают нам всем принимать оптимальные решения, быстро реагировать на изменения рынка труда и развивать конкурентные HR-преимущества наших компаний.

HR-брендинговые ошибки и заблуждения

МИФЫ ВОВЛЕЧЕННОСТИ ПЕРСОНАЛА

Важна ли отраслевая HR-специфика?

Кто «прогнется под изменчивый мир»
поколение Y или X?

КРОСС-ОТРАСЛЕВЫЕ КЕЙСЫ.

КНУТ, ПРЯНИК, КРІ ИЛИ МОТИВАЦИЯ 3.0?

Геймификация бизнес-процессов –
спасение или очередная мода?

Должны ли мы заниматься
счастьем сотрудников?

И многое другое...

Стоимость участия по кодовому слову
«Раннее бронирование»
действует только **до 1 августа**.
Обращайтесь к своему персональному
менеджеру НН или по адресу:
hrconf@hh.ru.

Спешите забронировать места,
звоните своему персональному
менеджеру НН!

 +7 812 607 77 29



ДИСПЛЕЙНАЯ РЕВОЛЮЦИЯ

Mir дому Canonical

Ubuntu 13.10 будет работать не на X.org, а на *Mir*, и это окончательно.

Оливер Рис [Oliver Ries], директор по развитию продуктов Unity и *Mir* в компании Canonical, объявил в начале июля о переходе Ubuntu на использование по умолчанию дисплейного сервера *Mir* со следующего релиза, Ubuntu 13.10. Такая стремительность объясняется намерением поставлять *Mir* в версии Ubuntu 14.04 LTS (с долгосрочной поддержкой), что требует дополнительной стабилизации технологии как минимум в одном промежуточном выпуске. Иначе внедрение *Mir* в настольный вариант дистрибутива пришлось бы отложить как минимум на год или решиться на выпуск сырого LTS-релиза, что неприемлемо для реализации планов по продвижению редакции Ubuntu Touch, ориентированной на применение в планшетах и смартфонах. Canonical намечает переход на единую кодовую базу, формирующую окружения для различных классов устройств (от смартфонов до ПК), экранных разрешений и систем ввода (клавиатура/мышь, сенсорный экран, пульт ДУ).

Разработка *Mir* продвигается интенсивно, и Canonical постоянно расширяет штат вовлеченных сотрудников. По словам Риса, *Mir* уже соответствует критериям качества, предъявляемым к компонентам в составе релиза. Доступен для тестирования прототип оболочки Unity 8, работающей поверх *Mir*, и реализована возможность запуска всех популярных рабочих столов.

В LTS-релиз *Mir* войдет по умолчанию, но рабочий стол по умолчанию (Unity 7) по-прежнему будет использовать X-сервер. *Mir* будет взаимодействовать с ним через прослойку *XMir*; по сути, *XMir* выступает заменой драйверов ввода/вывода.

Кроме того, в 13.10 предусмотрен специальный режим отката — *fallback*. Он

будет применяться в случае отсутствия поддержки используемой графической карты в *Mir* или при установке проприетарных драйверов NVIDIA и AMD. Поддержка *Mir* в проприетарных драйверах еще не реализована, но компании Nvidia и AMD намерены ее обеспечить для выпуска Ubuntu 14.04. Начиная с Ubuntu 14.10, дистрибутив полностью перейдет на графический стек на базе *Mir* и Unity 8, без дополнительной прослойки и *fallback*-режима.

Работу над официальными редакциями Ubuntu от сообщества (Kubuntu, Xubuntu, Lubuntu, Ubuntu GNOME и т.п.), планируется организовать на основе единого с Ubuntu графического стека на *Mir* и *XMir*. Но лидер проекта Kubuntu заявил, что предпочитает *Wayland*. Участники основанного на оболочке LXDE Lubuntu отказались от перехода ввиду необходимости обеспечить поддержку маломощных систем.

Оптимизация *Mir/XMir* еще не завершена. Согласно тестам Phoronix, связка Unity+*XMir*+*Mir* пока работает медленнее Unity+X-сервер. **LXF**



► Одним из аргументов перехода Canonical на *Mir* является то, что *Mir* идеально подходит для мобильных устройств, работающих на Ubuntu Touch.

Новости короткой строкой

► 19 июня 2013 года проекту FreeBSD исполнилось 20 лет. За это время он вырос в быструю, стабильную и надежную ОС для систем i386. Источник: www.freebsd.org

► *The Wall Street Journal* сообщил о тупике переговоров по покупке компанией Microsoft мобильного бизнеса Nokia: Microsoft сочла цену завышенной. Источник: <http://online.wsj.com>

► Стало известно: Microsoft тайно сообщала спецслужбам данные о еще не исправленных 0-day уязвимостях в Windows и Skype до выхода официальных обновлений, давая шансы атак на компьютеры и организацию слежки. Источник: <http://blogs.computerworlduk.com>

► В информационно-развлекательных системах для автобусов городов Барселоны и Омска используются Linux-дистрибутивы Slax и Fedora. Источник: www.omsklug.com

► Компания Valve начала тестирование Linux-версии клиента для DOTA 2, многопользовательской командной игры — гибрида стратегии реального времени и ролевки, доступной для установки через клиент Steam для Linux. Источник: <http://ru.wikipedia.org>

► В ночь на 2 июля в возрасте 88 лет умер во сне Дуглас Энгельбарт [Douglas Engelbart], изобретатель компьютерной мыши. 9 июля 2013 года погиб в аварии Сет Видал [Seth Vidal], один из ведущих разработчиков Fedora, автор *Yum* и репозитория *Fedora Extras*. R. I. P. Источники: www.ietf.org и www.durham.io

► С 14 июня 2013 года потеряна связь со шхуной «Нина», с Эви Немет [Evi Nemeth], автором «Руководства системного администратора Unix», на борту. Согласно Sail-World.com, судно затонуло. Источник: www.theregister.co.uk

► Ричард Столлмен [Richard Stallman] удостоен в 2013 включения в Зал славы Интернета [Internet Hall of Fame], среди 32 других выдающихся личностей. Источник: www.fsf.org

► По решению окружного суда штата Юта (США) официально возобновлено разбирательство «SCO против IBM». Источник: www.groklaw.net

► Неделя Российского Интернета-2013 (Russian Internet Week, RIW-2013) пройдет в Москве с 17–19 октября. Источник: <http://2013.russianinternetweek.ru>

Революционеры XXI века

В течении вот уже без малого 10 лет Canonical была и остается революционером. Часто неоднозначные, с первого взгляда — непродуманные, решения компании Марка Шаттлворта привносят дух инноваций и будоражат умы людей, по-новому заставляя взглянуть на Linux, и влияют на миллионы пользователей Ubuntu. Для многих Ubuntu является синонимом Linux.

Но бывают решения, способные и погубить. Не относится ли к таковым *Mir*, дисплейный сервер нового

поколения? Он призван заменить давно уже устаревший X.org и идет вразрез с планами всего сообщества СПО, которое сейчас активно развивает другую альтернативу «иксам» — *Wayland*. Однако чутье Шаттлворта и его талант организатора не раз приводили Canonical к победе. Пока что Марк доволен прогрессом проекта: окружение Unity 7 поверх *XMir* уже работает стабильно и гладко, и развитие идет быстрее, чем у аналогов от конкурентов, благодаря собственному API.

Алтай как точка роста

В 69-й раз состоялся комплекс мероприятий Международного фестиваля «Электронное будущее-2013!»

Фестиваль прошел 26–28 июня 2013 года в г. Барнауле Алтайского края. Основным событием для края стала Всероссийская конференция «Информационные технологии в управлении: прогнозируемые риски и возможности».

Организаторами мероприятий выступили Российское агентство развития информационного общества «РАРИО» и Ассоциация обучающихся центров.

Мероприятия прошли при поддержке Аналитического центра при Правительстве РФ, Общероссийского совета некоммерческих организаций, ассоциации «Сибирское соглашение» и Ассоциации руководителей служб информационной безопасности. Генеральными партнерами мероприятий выступили удостоверяющий центр «БТП», учебный центр «Ключевые решения», Алтайский государственный университет, Институт безопасности управления рисками, Алтайский диагностический центр и Многофункциональный центр Алтайского края.

69-й Сибирский окружной этап фестиваля был посвящен актуальным вопросам развития информационного общества и включил следующие мероприятия:

- » Всероссийская конференция «Информационные технологии в управлении: прогнозируемые риски и возможности»
- » Пресс-конференция и тематическая площадка, посвященная актуальным вопросам информационной безопасности
- » Круглый стол с участием некоммерческих организаций Сибирского федерального округа, посвященный вопросам совершенствования государственной поддержки социально значимых проектов
- » Круглый стол, посвященный вопросам развития инфраструктуры информационного общества в Алтайском крае (в рамках расширенного заседания Оргкомитета «МФЭБ-2013!»)
- » Круглый стол «Электронные сервисы для государства»

» Тематическая отраслевая площадка «Будущее свободного программного обеспечения»

» Тематическая отраслевая площадка «Электронное образование»

» Встречи с бизнес-сообществом Алтайского края, обсуждение перспективных проектов в области информатизации

» Семинар для молодых предпринимателей Алтайского края «Все о сайтостроении за пару часов»

В качестве ведущего пленарного заседания конференции выступил руководитель Российского агентства развития информационного общества «РАРИО»,

председатель президиума Общероссийского совета некоммерческих организаций Александр Айгистов. Модератором первой панельной дискуссии по направлению «Информационная безопасность» стал председатель правления Ассоциации руководителей служб информационной безопасности Виктор Минин. Модератором второй панельной дискуссии по направлению «Будущее свободного программного обеспечения» выступил профессор, д.т. н., директор Автономной некоммерческой организации содействия развитию индустрии программного обеспечения «Национальная программная платформа» Леонид Ухлинов. Модератором третьей панельной дискуссии по направлению «Электронные сервисы для государства» выступила советник губернатора Томской области, член координационного совета по информатизации межрегиональной ассоциации «Сибирское соглашение» Наталья Маслова. Модератором четвертой панельной дискуссии по направлению «Электронное образование» выступил профессор, д.пед.н., заведующий кафедрой педагогики высшей школы и информационных образовательных технологий Алтайского государственного университета Геннадий Лаврентьев.

Также в конференции приняли участие: главный советник департамента обеспечения безопасности в области информации и информационных технологий аппарата Совета Безопасности РФ Светлана Коновченко, заместитель директора Департамента государственного регулирования экономики Минэкономразвития РФ Павел Малков, советник руководителя Аналитического центра при правительстве РФ Юрий Вороненков, начальник департамента по связям с институтами гражданского общества администрации Алтайского края Валерий Труевцев, президент Ассоциации обучающихся центров Татьяна Басаргина, исполнительный директор Общественного совета информационного развития «Росинформразвитие» Анатолий Ганин, руководитель Сибирского центра социальных технологий Нина Шабалина, руководитель пресс-службы Регионального сетевого информационного центра RU-CENTER Виктория Бунчук и др.

Участниками мероприятий стали более 300 представителей органов государственной власти, бизнес-сообщества, научно-образовательной сферы и общественности Сибирского федерального округа.

Участниками мероприятий стали более 300 представителей органов государственной власти, бизнес-сообщества, научно-образовательной сферы и общественности Сибирского федерального округа.

Делегаты Конференции — представители субъектов РФ Сибирского федерального округа — высказались за необходимость формирования четкой государственной политики, направленной

«Участниками фестиваля стали более 300 представителей.»



» Перелетный символ Фестиваля — Золотой голубь вручается Администрации Алтайского края как принимающей стороне.

на привлечение граждан к активной жизни в информационном обществе. По мнению экспертов — участников мероприятия, отсутствуют эффективные мотивационные инструменты популяризации государственных электронных услуг и сервисов среди населения. Без массового информирования граждан о широких возможностях информационного общества, в том числе и механизмов электронной демократии, невозможно сформировать среду доверия к этим инновациям, к проводимому в стране курсу на модернизацию.

В рамках конференции был представлен проект создания на территории Алтайского края технопарка в сфере свободного программного обеспечения, который, как предполагается, станет единственным технопарком в России, специализирующимся на разработке отечественных продуктов с использованием СПО. Андрей Басаргин, генеральный директор ЗАО «Удостоверяющий центр «БТП», показал присутствующим, какой большой объем работы уже был проделан его командой по разработке данного инновационного мега-проекта, что нашло положительный отклик среди делегатов и участников конференции.

Фундаментом этого проекта стало торжественное присвоение статуса регионального Центра компетенций по разработке, внедрению и сопровождению программных решений в составе технологической платформы «Национальная программная платформа» на базе Алтайского диагностического центра, в состав которого вошли ведущие университеты, инновационные предприятия и общественные организации Алтайского края.

В ходе конференции была обозначена серьезная обеспокоенность ситуацией, связанной с недостаточно качественной подготовкой IT-специалистов, в частности, в сфере информационной безопасности. Представители Института национальной безопасности и управления рисками заявили, что на сегодняшний день в России процесс подготовки и переподготовки кадров в сфере информационных технологий заточен не под потребности предприятий, а под конкретные инфраструктурные и преподавательские особенности российских ВУЗов. По мнению участников мероприятия, необходимы выработка единых стандартов обучения специалистов в IT-сфере, а также четкое выполнение этих стандартов учебными заведениями.

На мероприятии также было отмечено, что предприятия, преимущественно малого и среднего бизнеса, существенно недооценивают риски, возникающие за счет активного использования информационных технологий в их работе. Так, например, зачастую, системный администратор организации обладает исключительным доступом ко всей информации, включая даже секретную, и в любой момент может ее использовать в своих целях. В связи с этим, по мнению делегатов Конференции, необходимо массовое обучение руководителей российских предприятий основам информационной безопасности и безопасному управлению информационными потоками.

Участники мероприятия также констатировали значительное повышение степени участия государства в управлении Интернетом. Так, например, по мнению экспертов, за последний год количество законопроектов в сфере регулирования российского сегмента сети Интернет возросло в два раза, что однозначно свидетельствует о повышении влияния Всемирной паутины на жизнь нашей страны. Но, тем не менее, при принятии государственных решений в части регулирования Интернета участники мероприятия настоятельно рекомендуют советоваться с профессиональным сообществом.

В рамках Сибирского окружного этапа Международного фестиваля «Электронное будущее-2013!» также состоялась встреча



► Пленарное заседание Алтайского этапа Международного фестиваля «Электронное будущее-2013!».

председателя президиума Общероссийского совета некоммерческих организаций Александра Айгистова и членов официальной делегации «МФЭБ-2013!» с руководителями некоммерческих организаций Сибирского федерального округа. Ключевой темой мероприятия стало совершенствование государственной поддержки социально значимых проектов. В рамках встречи руководители НКО представили на обсуждение ряд проектов, которые заинтересовали экспертов.

По итогам Всероссийской конференции «Информационные технологии в управлении: прогнозируемые риски и возможности» будет подготовлена резолюция. С докладами и текстом итогового документа можно будет ознакомиться на сайте организаторов: <http://rf-seminar.ru/>.

В завершение мероприятий для московской делегации было организовано посещение курортных и значимых мест в Алтайском крае. Участники мероприятий посетили Бирюзовую Катунь, археологические раскопки в районе Талдынских пещер, познакомились с удивительной коллекцией экзотических цветущих растений, лиственных и хвойных кустарников уникального в Сибирском регионе дендрария «Биолит», поднялись по канатно-кресельной дороге на гору Церковку, посетили ряд лечебных и культурных мест города-курорта Белокуриха и памятных мест, связанных с Василием Шукшиным и Михаилом Евдокимовым. **LXF**

«Необходима выработка единых стандартов обучения в IT-сфере.»



► Руководитель «РАРИО» Александр Айгистов говорит об актуальных проблемах развития информационного общества.



АЛЕКСЕЙ ФЕДОРЧУК
Тэг <сарказм>
по умолчанию,
смайлики по вкусу.

Куда идет Kubuntu?

С давних лет привыкли мы, что Ubuntu и ее отпрыски — близнецы-братья между собой (да и со своим предком): один инсталлятор, один базис, одни репозитории. Разве что десктопы разные — но они и в одном дистрибутиве бывают разные. Но так ли будет предь?

Началось с того, что Canonical сняла с довольствия Джонатана Ридделла [Jonathan Riddell] — чуть ли не единственного оплачиваемого разработчика одного из отпрысков, Kubuntu. Тот, впрочем, недолго сидел не при делах, будучи взят на содержание фирмой Blue Systems, известной финансированием ряда проектов, так или иначе связанных с KDE. И тогда возник вопрос: сохранит ли Kubuntu свое название? Компонент ее имени после литеры «K» — торговая марка, разрешенная к использованию на определенных условиях.

Как показала история, имя Kubuntu сохранил. Но и, с выходом релиза 13.04, приобрел своеобразие: свой инсталлятор, еще проще прародительского, и Muon Discover — аналог Центра приложений головного проекта и остальных его спутников.

Главное же — в другом: Ubuntu, начиная с грядущего релиза 13.10, вводит дисплейный сервер *Mir*, призванный заменить *X Window System*, которую давно уже пытаются списать в тираж, а Kubuntu хранит верность последней как минимум на два ближайших релиза. А затем, вслед за всем прогрессивным человечеством, планирует переход на *Wayland*.

Не начало ли это распада былого единства? Поживем — увидим.

alv@posix.ru

Сегодня мы рассматриваем:

- Debian 7.0** 16
Свободное ПО прекрасно тем, что мы стоим на плечах гигантов. И самые широкие плечи — у Debian. Он стабилен, мощен и всеобъемлющ, и снабжает платформой разработки ...надцать дистрибутивов. Ну, а каков он в роли настольной системы?
- Mint 15 Olivia** 18
Улучшились драйверы, интерфейсы, управление проприетарными кодеками... Mint все хорошеет. Однако не торпедует ли его успех нехватка убойных функций?
- Mageia 3** 19
Наш любимый французский дистрибутив навел косметику и включил новые панели настройки. А главное, обзавелся клиентом *Steam*, так что Ubuntu можно не устанавливать.
- Gizmosphere-комплект** 20
Крошечные машинки на базе ARM стали повсеместны, но некоторые считают ограниченный набор команд ARM уж слишком ограниченным. Данную проблему пытаются решить в Gizmosphere, для чего и утрамбовали в комплекте разработчика плату с микрочипом x86, GPIO, клавиатуру и мини-дисплей.
- Trine 2** 21
В этой игре вам придется не только скакать с платформы на платформу, но и перевоплощаться, сменяя, в зависимости от ситуации, свой облик: когда на прямолинейно-рыцарственный, а когда и на воровской.



» Если Debian хорош для Международной космической станции, то и для нас сойдет.



» Те, кого уже тошнит от *buntu, могут взять Mageia — он уважает пользователей, и ему к лицу берет.

Сравнение: Производные Ubuntu

с. 24

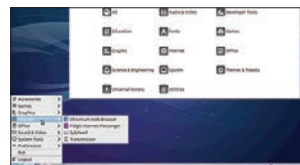
Bodhi Linux



Kubuntu



Lubuntu



Ubuntu



Zorin



При простой установке, огромных репозиториях и несравненной поддержке оборудования Ubuntu созрел давать потомство. Вот пять его лучших отпрысков, каждый со своей философией.

Debian 7.0



Дождались: перед нами новая версия самого уважаемого в мире дистрибутива во всем блеске. **Майк Сондерс** выясняет, держит ли она марку.

Вкратце

» Сообщество разработало и выполнило поддержку мегадистрибутива с гигантским набором пакетов и портированием на множество платформ. См. также: все дистрибутивы на базе Debian, такие как Ubuntu и Linux Mint Debian Edition.

Ключевое ПО

- » Ядро Linux 3.2
- » X.org 7.7
- » EGLIBC 2.13
- » GCC 4.7
- » KDE 4.8.4
- » Gnome 3.4
- » Xfce 4.8
- » LibreOffice 3.5.4
- » Gimp 2.8.2
- » Iceweasel 10
- » Apache 2.2.22
- » Samba 3.6.6
- » MySQL 5.5.30

Дebian ближе всего к тому, что считается стандартом дистрибутива Linux. Он служит основой для множества других систем, используется повсюду — от домашних компьютеров до серверов «большого железа [big iron]» — и поддерживается некоммерческой организацией (Software in the Public Interest — «ПО в интересах общества»).

С момента своего первого официального явления в 1996-м Debian добился огромного уважения среди сообществ Linux и свободного ПО, и не только за создание высококачественного и хорошо продуманного дистрибутива, но и за верность своим принципам и избегание всего проприетарного.

Медленно, но верно

Параллельно, Debian приобрел репутацию такого ленивца; его гигантский размер (нынешняя версия насчитывает 37 500 пакетов), длительные перерывы между релизами и частенько устаревшие версии приложений придают ему довольно потрепанный вид по сравнению с остроумными дистрибутивами. На серверах, где админы предпочитают как раз самый

консервативный и наименее изменчивый вариант, это не столь проблематично, но обычным пользователям, падким на новинки, такое не по душе.

Новым пришельцам на Debian может показаться непросто даже найти дистрибутив на www.debian.org. Если вы не замечаете крошечную кнопку «Скачать Debian»

как можно больше с налета, скачивайте серию из восьми установочных CD-образов или трех ISO DVD. Альтернатива — купить реальные комплекты CD/DVD, оказав финансовую поддержку проекту Debian; см. на www.debian.org/CD/vendors.

С 6.0 до 7.0 установщик особо не изменился, разве что избавились от мультязычной космической

«Можно установить Debian при помощи распознавателя речи.»

в правом верхнем углу, которая радушно отправляет к сетевой установке ISO-образа, то придется плутать по страницам Получение Debian, разбираясь с кучей вариантов загрузки. Разумеется, Debian и не ориентирован на новичков, но более внятного вида страница с проводкой по экранному снимкам не помешала бы.

Мы решили остановиться на сетевой версии в 485 МБ, скомбинированной для i386 и x86-64, которая, как следует из названия, скачивает большинство пакетов при установке. Если вы желаете захватить

темы. Добавлена поддержка UEFI (а не Secure Boot), а людям с нарушениями зрения отныне можно уста-

новить Debian при помощи распознавателя речи. Текстовый установщик позволит запустить Debian при 64 МБ ОЗУ, а для графического необходимо 128 МБ. Последующие шаги уточняют, как именно вы хотите установить Debian: в качестве графического рабочего стола, web-сервера, почтового сервера и т. д.

Переходим к графике

Выбрав графический настольный вариант, вы получаете Gnome 3.4 с LibreOffice 3.5 и Iceweasel 10, укомплектованный Adblock



» Любите ли вы Gnome или ненавидите, но в Debian 7.0 это рабочий стол по умолчанию, хотя любой другой, как и менеджер окон, тоже в вашем распоряжении.

Plus (фантазия на тему Firefox без маркировки Mozilla). На нашей машине процесс загрузки занял 43 секунды, от Grub до рабочего стола, задействовав 157 МБ ОЗУ на момент начала работы. Для сравнения, Fedora 18 с Gnome 3.6 требуется 53 секунды, и использует она 283 МБ ОЗУ, а Ubuntu 13.04 (в версии Unity) — 36 секунд и 348 МБ.

С точки зрения пользователя, главное нововведение в Debian 7.0 — добавление медиа-кодеков по умолчанию. Благодаря Libav, ответвлению Ffmpeg, поддерживается широкий спектр аудио- и видеоформатов; Rhythmbox и ему подобные могут «из коробки» воспроизводить MP3. Ранее заставить кодеки работать было непросто, и такая перемена необычайно радует.

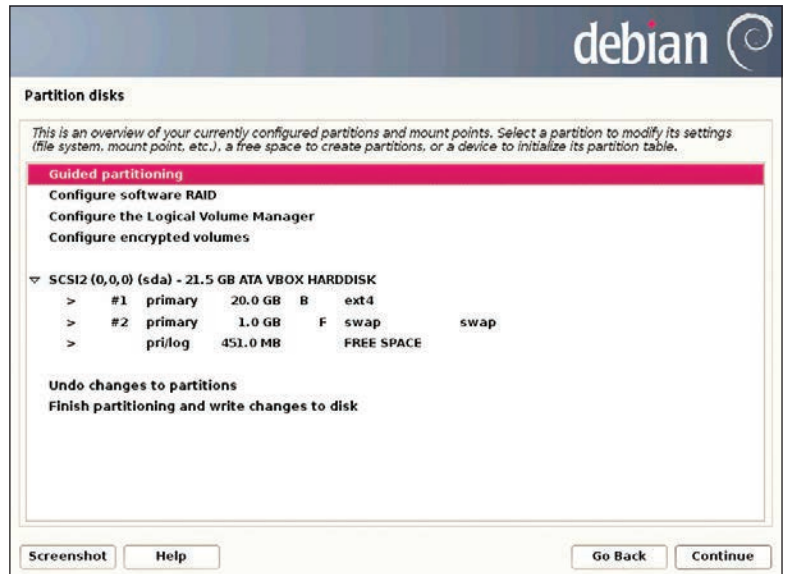
Под капотом, добавлена поддержка многоархитектурных установок, что позволяет легко запускать 32-битные программы в 64-битных окружениях x86, с автоматическим соблюдением всех зависимостей. Во многих пакетах активирована усиленная защита GCC; стали доступны официальные пакеты OpenStack для создания собственных облаков.

Так уж стабилен?

Но вот мы подходим к дискуссионной проблеме: стабильность. Безусловно, Debian — стабильнейший дистрибутив, если «стабильностью» считать «отсутствие крупных изменений». За период жизни одного релиза команда Debian выпускает только обновления безопасности и наиболее важные исправления, и все это тщательнейшим образом протестировано. С Debian можно не бояться, что новая порция заплат порушит систему.

Для сервера это здорово, но как насчет настольных систем? Может, он и стабилен, но это не гарантирует отсутствия ошибок. Так, разработка Debian 7.0 была заморожена 30 июня 2012 г. — значит, тогда и произошло последнее глобальное обновление пакетов. А следовательно, Debian 7.0 представляет слепок мира Linux и свободного ПО, каким он был... примерно год назад. Во многие основные при-

» Установщик Debian предполагает наличие среднего уровня навыков работы с компьютером, но зашедших в тупик выручит кнопка Help.



ложения за это время были внесены существенные обновления, и, зачастую, море исправлений. Естественно, за критическими ошибками с угрозой потери данных команда Debian следит, но более мелкие проблемы порой остаются.

Возьмем, скажем, Xfce: этот рабочий стол набирает сейчас невиданную популярность (особенно в контексте громкого негодования по поводу дизайна Gnome 3), а в Debian 7.0 он всего только в версии 4.8, хотя 4.10 вышла еще 28 апреля 2012 г. — даже до заморозки кода Debian, и в ней не только уйма классных новых функций, но и полным-полно исправлений. Так что пользователи Debian 7.0, не желающие вникать в тонкости с тестированием и ненадежными ветками, получают обветшавший рабочий стол.

Великоват?

Одна из проблем, бесспорно, и необычный размер Debian: 37 500 пакетов, с экзотическими портами вроде kFreeBSD (это среда Debian на базе ядра FreeBSD) и необычайно усложненной политикой управления. Может, это и не сделалось бы панацеей, но, вероятно, понижение статуса некоторых менее популярных платформ

до второго уровня помогло бы разработчикам сократить период замораживания.

В разных формах, мы используем Debian годами и очень любим его. Несомненно, это исключительно надежный дистрибутив, в который вложено много труда и души. И Debian 7.0 держит ожидаемую планку: это первоклассная серверная ОС.

В настольном варианте он также достаточно хорош, и добавление медиа-кодеков — важный прорыв. Но его устарелость ощущается сразу же после установки, и те, кто не следит за тестируемостью/нестабильностью и по каким-либо причинам не использует обратные порты, остаются с приложениями годичной давности и в долгом ожидании.

Сокращение фазы тестирования (например, до 6 месяцев вместо почти года) и более частые релизы помогли бы решить проблему, и это было бы реально, будь Debian менее основательным, более целеустремленным проектом. Хотя об этом судить его разработчикам — в любом случае, мы всегда благодарны им за их упорный труд. **LXF**

Срок жизни Debian 7.0

Как долго вы сможете получать обновления безопасности и основные исправления, если установите Debian 7.0 сегодня? На этот вопрос нет ясного ответа — по крайней мере, пока. Debian 7.0 будет поддерживаться еще год после выхода следующего релиза, и со времен Debian 3.1 в 2005 проект выпускал основные релизы в промежутке

от 22 до 27 месяцев. Так что текущая оценка срока поддержки Debian 7.0 — до середины 2016.

Сравним с ситуацией у конкурентов: Ubuntu 12.04, релиз с долгосрочной поддержкой, будет обновляться до апреля 2017. Red Hat Enterprise Linux 6 (как и его свободная сборка CentOS), между прочим, до 2020. Что звучит прекрасно, если

не подумать вот о чем: это срок поддержки для всей 6-й серии Red Hat.

Так что срок поддержки Debian весьма неплох, хотя и короче, чем долгосрочная поддержка Ubuntu. Но здесь вступает в силу качество обновлений, а мы знаем кое-кого, кто в прошлом обжегся на некоторых небезупречных новшествах от Canonical...

LINUX FORMAT Вердикт

Debian

Разработчик: Проект Debian
Сайт: www.debian.org
Цена: Бесплатно по лицензиям FOSS

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

» Еще один качественный релиз от Debian, лучшая версия для настольных систем. Но не тянет ли проект назад его же самолюбие?

Рейтинг 8/10

Mint 15 Olivia

Продолжит ли Mint царить на рабочем столе? Укрепив на флагштоке версию 15, Бен Эверард смотрит, кто салютует этому юзеролюбивому дистрибутиву.

Вкратце

» Один из самых популярных дистрибутивов Linux. См. также: Ubuntu и Mageia



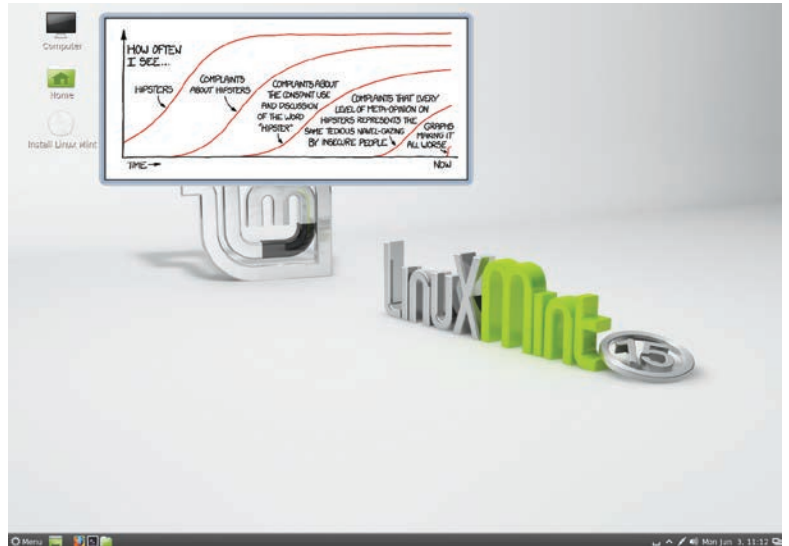
Грэм сказал...

Причина популярности Mint вполне очевидна: он предлагает богатую среду и не лезет в то, как вы в ней работаете. Чего Mint 15 не изменил — это самая убойная функция.

З апустив версию live Linux Mint 15 (кодовое имя — Olivia), трудно понять, что же изменилось по сравнению с версией 14. Нам пришлось заглядывать в примечания к релизу, чтобы вообще распознать какие-то изменения. Помимо обычного лоска и обновленных версий, в Olivia есть три новых функции: новый менеджер дисплея (MDM), исходники программ (mintsources) и программа поиска драйверов (mintdrivers). Все это приятные дополнения, но ни одно из них не назвать революционным.

Более крупные изменения произошли в средах рабочего стола: Cinnamon и Mate. Cinnamon — который дошел до версии 1.8 — предлагает так называемые десклеты [desklet]. Это приложения, привязанные к рабочему столу, подобны плазموидам в KDE или виджетам в Android. Пока что их немного. На момент написания было три, и они были предустановлены (часы, фоторамка и программа запуска), и три можно было дополнительно скачать через приложение-десклет (программу просмотра XKCD для умников с переизбытком свободного времени, да еще двое часов). Однако, принимая во внимание популярность Mint, мы полагаем, что к ним вскоре присоединятся и другие.

Но большая часть улучшений Mate осталась за сценой. Впрочем, то, что они невидимы для пользователя, еще не повод считать, что они незначительны. В кодовой базе навели порядок и упростили ее, чтобы ею было удобнее пользоваться в будущем.



» Десклеты либо позволяют настраивать рабочий стол так, как вам удобно, либо становятся первым шагом на пути к супер-сложной чудовищности (см. KDE).

Отсутствие изменений в Mint радует. Он достиг состояния стабильности, и не меняется во имя этого. Пользовате-

система, то нет никакой необходимости обновляться каждый раз до нового релиза. Однако, начиная с этого релиза, все-таки придется обновляться каждый раз, чтобы обеспечить обновления системы безопасности.

Это изменение делает намного привлекательнее версии с долгосрочной поддержкой — LTS. Mint 13 будет поддерживаться до апреля 2017, а очередная LTS-версия не выйдет до мая 2014. И если вы поклонник Mint, пора решать, стоят ли новые функции более быстрого цикла изменений. **LXF**

«Mint достиг стабильности, и не меняется во имя этого.»

лям совершенно незачем ломать нарабатанные навыки просто потому, что так захотелось кому-то из разработчиков. Фактически, самое крупное изменение в системе пришло не от команды Mint, а из Ubuntu, на котором Mint основан. Текущий релиз Ubuntu, Raring Ringtail, это первая версия Ubuntu, поддержка которой предоставляется на срок всего лишь девять месяцев. Учитывая, что Mint 15 вышел через месяц после своего родителя, даже если вы только что его установили, у вас остается только восемь месяцев поддержки (которая закончится в январе 2014 года). Предыдущая версия поддерживалась на три месяца дольше (до апреля 2014).

Боязнь перемен

Это решение было принято не командой Mint, оно было навязано им Canonical, которые сократили период поддержки с 18 месяцев до девяти для версий не-LTS (Long Term Support). Команда Mint давно заявила, что если вас устраивает ваша

Свойства новскидку



Cinnamon 1.8

В этой версии впервые появляется функция десклетов, и вы сможете читать комиксы или проверять время прямо с рабочего стола.



Mate 1.6

С точки зрения графики, не сильно отличается от Mate 1.4, но разработчики уверяют, что зато за сценой все приведено в порядок.

LINUX FORMAT Вердикт

Mint 15 Olivia

Разработчик: Команда Linux Mint

Сайт: www.linuxmint.com

Цена: Бесплатно под разными лицензиями

Функциональность 9/10

Производительность 9/10

Простота использования 9/10

Документация 8/10

» Linux Mint 15 — это настольный Linux в своем лучшем виде, но короткий срок поддержки — серьезный недостаток.

Рейтинг 9/10

Mageia 3

Грэм Моррисон нашел дистрибутив расцветшим на пепле старой платформы.

Вкратце

» Простой в использовании дистрибутив, устанавливаемый с live CD или полнофункционального DVD.



Бен сказал...

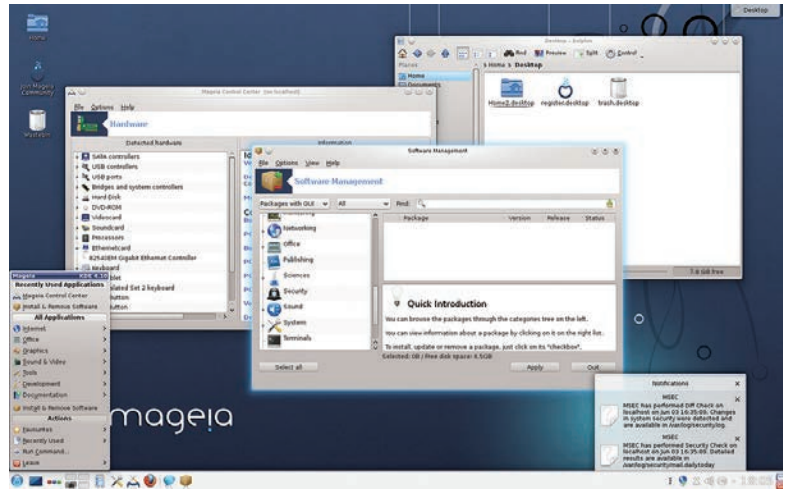
В Mageia один из лучших вариантов установки KDE, которые мне доводилось видеть; инструмент настройки также великолепен и прост в работе. Хотя продвинутым пользователям его функции могут показаться несколько примитивными.

Мageia — как горячее домашнее рагу холодным зимним вечером: этот дистрибутив согревает, наполняет комфортом и напоминает о более невинном возрасте. В мире вечных перемен, экспериментальных рабочих окружений и неразберихи среди разработчиков, Mageia возвращает вас в лучшие времена, когда дистрибутивы поддерживали status quo, изо всех сил стараясь соответствовать желаниям пользователей. Разве что история Mageia больше походит на *Игру престолов* [Game of Thrones], чем на *Все создания, большие и малые* [All Creatures, Big and Small].

Это детище расколовшегося сообщества Mandrake и того времени, когда они взяли дело в свои руки, дабы создать дистрибутив, каким по их представлению должен был стать Mandrake. И они сделали великое дело. Без всякой иронии и предвзятости — он обращен к пользователю; уже установщик спрашивает у вас, какой рабочий стол вам нравится. Краткий обзор, предлагаемый при установке, до сих пор не имеет аналога, а теперь есть и новая функция, позволяющая заодно устанавливать проприетарное и закрытое ПО. Это подкупает, как и все то, что снискало Mandrake добрую славу среди новичков, когда Linux было так трудно использовать; и здорово, что Mageia удается сохранить прежнее обаяние.

Развивая чары

Поскольку исходный Mandrake был дистрибутивом KDE, мы последовали традиции и выбрали при установке рабочий стол KDE 4.10. Немного иначе теперь



» Простота управления пакетами — до сих пор одна из главных причин использовать Mageia, наряду с фантастической широтой возможностей.

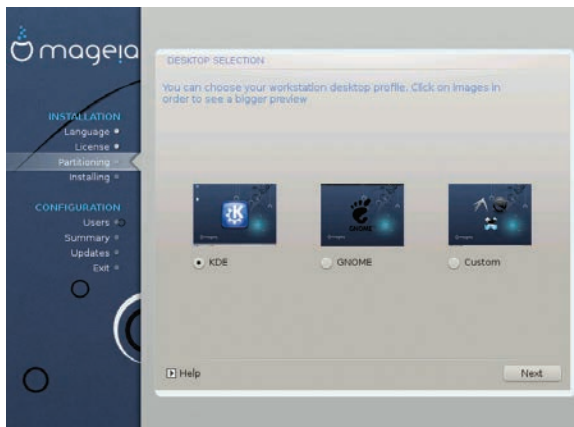
выглядит логотип Mageia и прочее оформление, и на наш взгляд, выглядит прекрасно. Стандартная синяя тема хорошо сочетается с KDE, и весь рабочий стол, по ощущениям, превосходный. Приложения запускаются через меню типа «Старт», вместо громоздкого меню KDE, убраны лишние комнаты, но стиль и эффекты сохранены. Никуда не делось все то же странное синее подсвечивание окон, но его пока еще ни один дистрибутив отменить не решился.

Сильной стороной Mageia, унаследованной от Mandrake, всегда был знаменитый *Control Center*, ныне занимающий почетное место в панели запуска KDE. Благодаря доработке, он теперь прекрасно сочетается с родными панелями KDE, и как прежде предоставляет доступ к настройке многих программ и элементов оборудования. Но вот над удобством еще стоит поработать. Здесь явно используется много старого кода, и поведение многих панелей непоследовательно: одни при нажатии Выход возвращают на главную страницу, а другие при этом все отключают. Также нет никакой «путеводной нити», чтобы понять, где именно в иерархии настроек вы находитесь. Подобный недостаток лоска, возможно, обусловлен иными приоритетами.

Оффлайн

Приоритеты, в основном, заключаются в обновлении пакетов и добавлении новых приложений, и это еще одна причина устанавливать дистрибутив с полноценного

DVD, поскольку сделать это вы можете, когда захотите — идеальный вариант при отсутствии широкополосного доступа. Здесь и цифровая аудио станция *Ardour*, и когорта сопровождающих ее музыкальных приложений. Облегченный рабочий стол RazorQT тоже входит в состав DVD, составляя прекрасную альтернативу KDE, на основе Qt, а пользователей Gnome порадует добавление прекрасного почтового клиента с функцией обсуждений, *Gearsy*. Официальный клиент *Steam* осчастливит игроманов, как и прекрасная поддержка драйверов Nvidia, что превращает Mageia в достойную альтернативу Ubuntu в плане игр на Linux. Короче говоря, команда Mageia потрудились на славу. Им удалось уловить суть Mandrake и подарить ему новую жизнь. **LXF**



» Использовать Mandrake-производный дистрибутив без KDE идеологически неверно.

LINUX FORMAT **Вердикт**

Mageia 3

Разработчик: L'équipe de Mageia
 Сайт: www.mageia.org
 Лицензия: Разные свободные лицензии

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

» Скажем с дурацким французским акцентом: Ooh-la-la-la, c'est Magnifique!

Рейтинг 8/10

Gizmosphere-комплект

Процессоры ARM слетелись на мини-компьютеры, но не дадут ли x86 отпор?

Вкратце

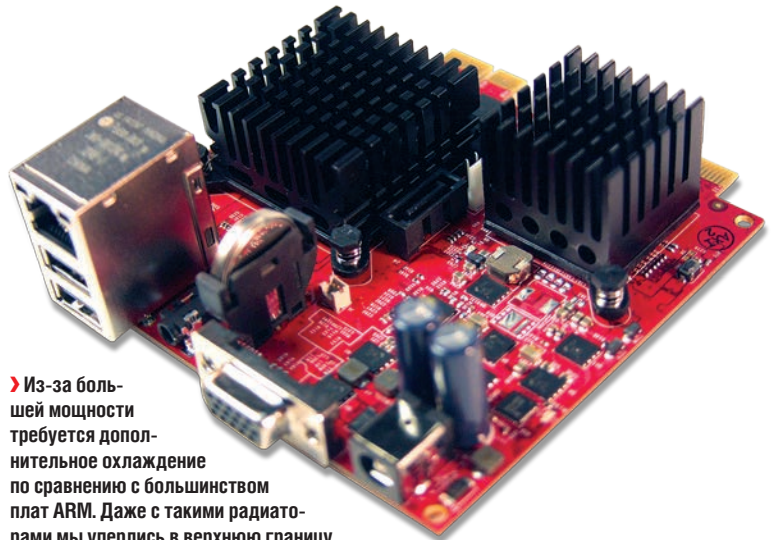
» Мини-компьютер на x86. См. также платы ARM, типа Raspberry Pi или Beagleboard.

С тех пор как Raspberry Pi в прошлом году штурмом взяла мир, архитектура ARM стала модным выбором для небольших плат, и чуть ли не еженедельно выходит новая такая плата. Однако x86 еще жива. Она все еще применяется на настольных компьютерах и серверах, и ее тоже можно масштабировать.

Основная разница между двумя архитектурами — набор команд, которые понимает процессор: x86 — чип CISC (Complex Instruction Set Chip — с комплексным набором команд), а ARM — чип RISC (Reduced Instruction Set Chip — с сокращенным набором команд). То есть в программах для x86 набор команд шире; но, как следствие, сами микросхемы сложнее. Впрочем, количество команд играет роль только на самом низком уровне. Одну и ту же программу можно скомпилировать для x86 и ARM, и на уровне пользователя все будет одинаково, хотя на разных системах, разумеется, она будет выполняться по-разному.

Gizmosphere — маленькая плата на основе серии G гибридных центральных процессоров (APU) архитектуры x86 от AMD. APU, если вам интересно, означает Accelerated Processing Unit [Ускоренный процессор]. В зависимости от способа его рассмотрения, это либо центральный процессор с графическим процессором сбоку, либо однокристалльная система с оперативной памятью на отдельной микросхеме. По сути, это попытка уменьшить габариты системы x86.

Даже при такой миниатюризации она почти вдвое больше Raspberry Pi и гораздо дороже, но зато вы получаете производительность и возможность запуска программ, скомпилированных для x86 (например, проприетарных, которые не получится перекомпилировать).



» Из-за большей мощности требуется дополнительное охлаждение по сравнению с большинством плат ARM. Даже с такими радиаторами мы уперлись в верхнюю границу температурного диапазона.

Производительность см. в таблице результатов наших тестов.

По результатам видно, что она примерно втрое быстрее Raspberry Pi и почти сравнялась по производительности с Celeron M 1,5 ГГц или вышла на одну десятую скорости компьютера Specialist Intel Core i5. Другими словами, она способна конкурировать с самыми быстрыми схемами ARM. Мы ожидали прекрасных показателей в нескольких областях, например, в 64-битной обработке. За время наших тестов плата прилично нагрелась, поэтому в зависимости от вашего плана ее монтажа охлаждение будет иметь значение.

Нишевая привлекательность

Это делает ее в какой-то степени нишевым продуктом — когда вам нужен полноценный x86, но не нужны чрезмерная вычислительная мощность и компьютер обычных размеров. У нее есть еще несколько компонентов, обычно отсутствующих на маленьких платах ARM: например, разъем SATA, который пригодится, если вы захотите построить NAS или PVR.

Но Gizmosphere — не рядовая плата x86: это и среда встроенной разработки с GPIO, клавиатурой и маленьким дисплеем. Их можно программировать с помощью Sage SmartProbe (в комплекте), которая подключается к другому компьютеру по USB, а к Gizmosphere через JTAG. Набор содержит проприетарное ПО, но только в пробных версиях, ограниченных 20 часами использования. Чтобы продлить этот

срок на год некоммерческого использования, нужно приплатить \$299 — неслабый довесок для платы ценой всего \$199, но гораздо меньше стоимости полноценной коммерческой лицензии: \$3000.

Если вы намерены пойти таким путем, Gizmosphere предоставляет хорошую площадку для старта в мир встроенных систем x86. Sage — мощная программа с набором примеров, хотя и не имеет такого же уровня поддержки сообщества, как, например, Raspberry Pi, поэтому найти в Сети ответы на свои вопросы сложнее.

Если вам понадобится освоить разработку для встроенных систем x86, это хороший выбор. Однако, если это всего лишь хобби и вам нужно только больше процессорной мощности и GPIO, честно говоря, Raspberry Pi лучше. **LXF**

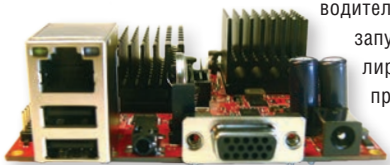


Таблица результатов

Тест	Gizmo Explorer	Raspberry Pi
Blowfish	26,98	68,45
Cryptohash*	45,94	13,33
Fibonacci	7,96	18,13
N-Queens	31,38	58,95
FFT	14,52	102,55
Raytracing	65,24	89,31

*Примечание: для этого теста чем больше, тем лучше. Для остальных — наоборот.

LINUX FORMAT Вердикт

Gizmosphere-комплект

Разработчик: Gizmosphere
Сайт: www.gizmosphere.org
Цена: \$199

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

» Одна из лучших миниатюрных плат x86, хотя выглядит огромной по сравнению с конкурентами ARM.

Рейтинг **7/10**

Trine 2

Евгений Балдин, один в трех лицах, перепрыгивает с платформы на платформу — естественно, ради спасения Вселенной.

Вкратце

» Старый добрый платформер с изумительными картинками и загадками.

Компания Frozenbyte, из страны, породившей Линуса-Бенедикта Торвальдса, уже отметилась на игровом поле GNU/Linux, весьма зачетно выступив на Humble Bundle с фантастической «бродилкой-стрелялкой» *Shadowgrounds* и красивой головоломкой-платформером *Trine* — она же, в русской транскрипции, *Триединство*. Неспешно-основательные финны не стали останавливаться на достигнутом и выпустили *Trine 2*... года полтора назад. С тех пор уже вышло дополнение «Гоблинская угроза». И только 31 мая 2012 г. стала доступна Linux-версия, хотя информация о планах на нее прошла аж в 2010-м. Но все хорошо, что хорошо кончается. Купить *Trine 2* можно на сайте разработчиков, а также через сервисы цифрового распространения игр: на Steam — с DRM, но со скидками; на Desura — без DRM, но подороже.

Версия игры для данного обзора куплена на распродаже Steam, работающего достаточно стабильно; тут все стандартно. Как минимум, на Ubuntu 12.04. При запуске *Trine 2* на двухмониторной конфигурации в режиме TwinView (видеокарта Nvidia) хромает режим полного экрана: остается либо выбрать оконный, либо отключить один монитор в настройках Nvidia X Server, благо это можно сделать «на лету». При запуске можно выбрать русскую локализацию; озвучка вполне себе на высоте. Но хроническое горе — кириллические шрифты: межбуквенные расстояния ужасны. Разобрать можно, но глаза стремятся куда-то под лоб. Текст переведен только в основном выпуске, а в расширении — (пока?) нет.



» Наш комплимент графике в *Trine*: она, пожалуй, способна потягаться с диснеевской.

В *Триединстве-2* зыскательный любитель обнаружит упрощения, делающие игровой процесс ненавязчивым и легким. Нет теперь маны. Всегда можно перераспределить очки опыта, и можно не бояться ошибок в развитии персонажей или экспериментов с разными комбо. Почти везде расставили контрольные точки, избавляя от горечи фатальных промахов. Двигаясь по сюжету, игрок не ощущает особых «напрягов». По умолчанию игра явно не для желающих застрять надолго; для истинных энтузиастов имеется режим *hardcore*.

Удачливому вору (точнее, предпринимательнице) Зое, простому, как молоток, рыцарю Понтию и ныне остепеневшемуся магу Амадею внезапно явился разумный супер-артефакт Трайн (по-нашему, Триединство) и принудил эту не слишком дружную тройцу спасти очередное волшебное королевство от очередной же напасти, магически слив трех упомянутых индивидов в одного, способного в любой момент сменить личину. Возможна кооперация — тогда героям удается помогать друг другу. Игрока, как всегда, мотивируют жадной достичь, невзирая на препятствия, главного босса и спасти Вселенную (королевство). Добавочные стимулы — шансы поглядеть на красоты, облазить все закоулки уровня, открыть все сундуки и выпить все бутылки с опытом — делают *Trine* интереснее проходных прыгалок. Накопленный опыт идет на освоение новых приемов — чем выше их уровень, тем они полезнее, но очков опыта вечно не хватает, и придется что-то выбирать.

У каждого героя свои уникальные способности. Вор Зоя — основная единица — умеет крюком с веревкой цепляться за структуры, подтягиваться и прыгать, кстати и на балки, зацепленные крюком. Воину Понтию в свалке с врагами долго не жить, он спец по ближнему бою. Из его умений важен молот: в пути вас ждет масса секретов за каменными стенками. Маг Амадей в бою почти бесполезен, зато умеет создавать и левитировать предметы; перебираясь через препятствие, станьте им, да пусть создаст поактивнее. В начале полезны замораживающие стрелы, а в конце — замедление времени.

Я помню времена, когда под *nix играли разве что в *xtank* или *xkobo*. Но все заметно изменилось, и новые игры привлекают не только умы, но и глаза. **LXF**

Свойства навскидку



Логические загадки

Решаются как грубой силой, так и ловкостью или привлечением волшебства. Где не помогут два ящика один на другом — спасут четыре.



Персонажи

Каждый герой владеет своими, присущими только ему, уникальными способностями. Героя в любой момент можно сменить.

LINUX FORMAT Вердикт

Trine 2

Разработчик: Frozenbyte

Сайт: www.frozenbyte.com

Цена: \$ 14,99 на сайте разработчиков, 468,51 руб. на Desura, 349 руб. на Steam

Сюжет	6/10
Графика	9/10
Увлекательность	9/10
Оправданность цены	9/10

» Сюжет в таких играх не главное, а все остальное вполне себе на месте.

Рейтинг 8/10



КОНФЕРЕНЦИЯ ИННОВАЦИЙ

В России замечен рост и потенциал

Крупнейший поставщик чипов для телефонов и планшетов обозначил свои приоритеты.

3 июля 2013 года компания Qualcomm провела в Москве конференцию Qualcomm Innovation, рассказав о достижениях и планах компании. Посетителям показали две новые технологии: Vitality (создание приложений с дополненной реальностью) и AllJoyn (объединение разнородных устройств в единую сеть).

Президент европейского отделения Энрике Сальватори [Enrique Salvatori] посулил к 2015 году в России 100 млн устройств с поддержкой стандарта 3G: «доля чипсетов компании на нашем рынке очень велика, но главное — наш рынок продолжает расти и развиваться, что делает Россию одной из приоритетных стран для Qualcomm. Компания родилась мобильной, поэтому она эволюционирует вместе с мобильным рынком».

Министр связи Николай Никифоров рассказал, что в РФ более 60 млн устройств уже подключено к Сети, но из-за больших расстояний доступ не одинаков: на Дальнем Востоке и в Сибири положение хуже, чем в Центральном регионе, и государство обязано решить эту проблему, ведь развитие телекоммуникаций ведет к росту ВВП. Планируется продвигать бесконтактные платежи, используя офисы Почты России как базу. Еще одна важная цель — обеспечить широкополосным интернетом школы. Также нельзя не отметить развитие связи 4G в России — постепенно она появляется все в большем количестве городов.

Президент всей компании Qualcomm Пол Джейкобс [Paul Jacobs] указал, что в телекоммуникации вкладываются все большие средства, и сейчас во всей индустрии налицо 3 основных тренда: переосмысление персональных компьютеров (т.н. эра пост-ПК), 1000-кратный рост передачи данных и создание «шестого чувства» на цифровой основе. Говоря об увеличении количества передачи данных, он сообщил, что с 2010 по 2011 год произошел двукратный рост: 100 млн устройств с поддержкой 3G/4G соединения — это самый большой показатель по Европе. В скором времени ожидается

Президент Qualcomm Пол Джейкобс отметил приоритетность Российского рынка мобильных устройств.



еще более резкий рывок. Для снижения нагрузки на сети компания представила ряд концептов, например, протокол LTE-D, работающий по схеме “personal to personal”. Соединение происходит напрямую между устройствами, без Интернета. К 2020 году к сети подключат 25 млрд устройств. Джейкобс уверен,

«В скором времени “умной” станет вся бытовая техника.»

что в скором времени «умными» станут не только телевизоры, но и автомобили, холодильники, чайники, духовые шкафы и прочая бытовая техника.

Для этого Qualcomm разработала платформу AllJoyn — т.н. «Интернет вещей». С ее помощью можно связать даже устройства, не имеющие своей ОС: пультом управления станет смартфон. Она также имеет потенциал в образовательной сфере — через приложения на ее базе преподаватель сможет отправлять на устройства детей учебный контент через Wi-Fi, Bluetooth или Ethernet, однако разработчик волен добавлять в свое приложение поддержку других технологий связи, например, NFC. Система открыта, она может работать на iOS и Android, а ее SDK доступен на официальном сайте.

Для создания механизмов взаимодействия с окружающей средой Qualcomm предлагает разработчикам платформу Vitality — на ее базе независимые программисты могут создавать свои решения, использующие дополненную реальность. Более 4000 приложений уже встроили данную платформу: это учебники, игры, журналы, туристические гиды.

Qualcomm является одним из ведущих игроков на рынке чипсетов (все флагманские смартфоны сейчас имеют хотя бы одну версию на процессоре Snapdragon), поэтому ее желание взаимодействовать с разработчиками (создание открытых платформ и референсных устройств) должно помочь потребителям — больше программ и приложений будут оптимизироваться под решения Qualcomm.

Выступивший на конференции Евгений Касперский рассказал о ситуации с вирусами на мобильных платформах. По данным его компании, за последние два года зафиксировано больше вирусов, чем за предыдущие пять, а самая беззащитная ОС — Android: на нее приходится около 97% всех существующих угроз, в том числе взлом мобильного банкинга, кража персональной информации, рассылка спама, фишинг, «поддельные» программы и блокировки системы. Касперский отметил, что основная задача его компании — это оптимизация алгоритмов защиты под мобильные платформы.

ЗАГАДОЧНЫЙ СУПЕРТЕЛЕФОН

Новый телефон Google

Создатели приоткрыли завесу тайны над Motorola X Phone.

Восхищение запланированным к выпуску Motorola X Phone, кажется, уже достигло предела, но Motorola и Google продолжают подогревать интерес публики к этому устройству различными способами. Утверждается, что это устройство революционное. Интерес к нему наблюдается не только в США, где оно появится в продаже в первую очередь, но и в России, где Motorola официально не существует уже три года.

В начале года появлялись слухи о том, что Motorola X Phone будет полностью настраиваемым. Буквально от начала и до конца. Внутри и снаружи. Сейчас, покупая смартфон, максимум, что мы можем — это выбрать из пары-тройки цветов да решить, нужно ли нам 16 гигабайт встроенной памяти или побольше. В случае с Motorola X Phone это будет совсем иначе, утверждают создатели. Сообщалось, что смартфон выйдет не менее чем в 20 цветовых вариациях и с использованием различных материалов корпуса: металл, пластик, дерево, карбон.

Реклама Motorola X Phone сообщает, что этот гаджет полностью будет разработан и продуман, а также собран и выпущен в США. При этом, несмотря на страну сборки и необычные на сегодняшний день аспекты кастомизации, в Motorola планируют выпустить агрегат, доступный массовому пользователю, продаваемый не по завышенной цене.

Казалось бы, что смартфон, который позволяет все то, что описано выше, должен стоить недешево. Но Google и Motorola уверяют, что он будет по карману большинству любителей Android. Самые первые слухи сообщали, что его стоимость составит \$199 без контракта оператора. Да, это не совсем про нашу страну, у нас такой практики нет, да и цены даже на серые поставки будут наверняка выше. Но все же, хотя бы в США цена привлекательна. Вот что сказал об этом Деннис Вудсайд [Dennis Woodside] — президент Motorola: «Одна из областей, которая, как мы думаем, открыта для Motorola — создание недорогих, но качественных смартфонов. Сейчас обычный телефон стоит \$30, смартфон стоит \$650. Этой пропасти не должно существовать».

Сейчас Motorola полностью принадлежит Google, о чем они напоминают теперь постоянно в обновленном логотипе компании. Выпуская свои смартфоны,

➤ Самый ожидаемый телефон 2013 года удалось «случайно» сфотографировать на одной из закрытых презентаций для операторов.



Google может устанавливать настолько низкие цены, насколько захочет, пополняя при этом армию любителей Android и увеличивая доходы за счет поиска, рекламы и предоставления услуг.

Ожидается, что Motorola X Phone получит четырехъядерный процессор MSM8960 Pro, работающий на частоте 1,7 ГГц, видеоускоритель Adreno 320,

синхронизировать информацию, искать ее в Сети; умеет работать с различными приложениями, развлекать игроми и мультимедиа-контентом; обладает датчиками, помогающими ориентироваться в пространстве.

Motorola X Phone будет куда интереснее, утверждают разработчики. По их словам, он научится понимать, что вы собираетесь делать, еще до того, как вы начнете действовать. Motorola X Phone будет анализировать данные со всех сенсоров и согласно внутренним алгоритмам предугадывать действия своего владельца. При этом за каждую часть поступающей информации и ее анализ будет отвечать отдельное ядро, что позволит постоянно держать все необходимые сенсоры активными и при этом экономить заряд аккумулятора. К примеру, если взять его в руку так, будто нужно сделать фото, автоматически запустится приложение камеры.

Все это намекает на глубокую интеграцию с Google Now!

Ни один из существующих смартфонов не обладает такой догадливостью и интуицией. Motorola X Phone должен стать инновационным устройством. **LXF**

«Motorola X Phone будет предугадывать действия своего владельца.»

2 ГБ оперативной памяти, 16 ГБ встроенной, HD-экран 720p, десятимегапиксельную основную камеру и фронтальную на 2 мегапикселя.

Смартфон будет продаваться через Google Play, ритейлеров и операторов. Охват аудитории будет максимальный. В Россию поставок не будет, однако «серые» продавцы уже принимают предзаказы в Интернете.

Насколько богат функциональностью и умен современный смартфон? Он умеет

Сравнение

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

Вариации Ubuntu

Любите Ubuntu, но вам ненавистен Unity? Маяк Шарма сравнивает пять дистрибутивов на базе Ubuntu, которые идут с разными рабочими столами.



Про наш тест...

Сравнивать такие дистрибутивы — хитрая задача. Поскольку все они на базе Ubuntu, по обычным критериям сравнения (например, установке) они одинаковы. Суть нашего Сравнения в том, чтобы помочь вам выбрать дистрибутив, с которым вам будет комфортнее, чем с Unity. Еще один важный параметр сравнения — системные требования. Unity требует ускорения графики, исключая использование стандартного Ubuntu на старых компьютерах.

Мы также учли инструменты индивидуальной настройки, которые поставляются с рядом дистрибутивов, имеющих собственный менеджер пакетов. Некоторые заменили инструменты по умолчанию в своей среде рабочего стола, ради большего удобства.

Ни одно сравнение дистрибутивов не будет полным без оценки наличных приложений и опций настройки, и это особенно важно, поскольку оба этих фактора зависят от рабочего стола.

Наша подборка

- » Bodhi Linux
- » Kubuntu
- » Lubuntu
- » Ubuntu GNOME
- » Zorin

Вы — пользователь настольного Linux и любите Ubuntu, но Unity побаиваетесь? Вам повезло. И у Canonical, и у сторонних разработчиков налицо масса вариаций Ubuntu, сохраняющих базовую инфраструктуру и суть Ubuntu, но не рабочий стол по умолчанию Unity. Canonical производит официальные разновидности с момента выхода второго релиза, но на них стали обращать куда больше внимания после перехода Ubuntu на Unity. Самая старая, и одна из самых популярных — Kubuntu, предлагающая рабочий стол KDE; если кому нужны все прелести Ubuntu на маломощном компьютере — для тех есть легковесный

Lubuntu; а начиная с релиза 13.04 Raring Ringtail, пользователи также смогут использовать рабочий стол GNOME благодаря новой разновидности Ubuntu GNOME.

Помимо этих официальных разновидностей, многие независимые разработчики выбирают Ubuntu в качестве основы для своих собственных дистрибутивов. Большинство таковых — просто релиз Ubuntu, обогащенный парой-тройкой

приложений, но некоторые разработчики вкладывают дополнительные усилия. Bodhi Linux — полувозобновляемый дистрибутив, основанный на стабильных релизах Ubuntu Long Term Support (LTS). Он устанавливает элегантную минималистскую систему, легко настраиваемую и под новую, и под старую машину. Также рассматривается Zorin OS, разработанная для пользователей, переходящих с Windows.

«И у Canonical, и у сторонних разработчиков налицо масса вариаций Ubuntu не с Unity.»

По умолчанию

Что припасено в коробке?

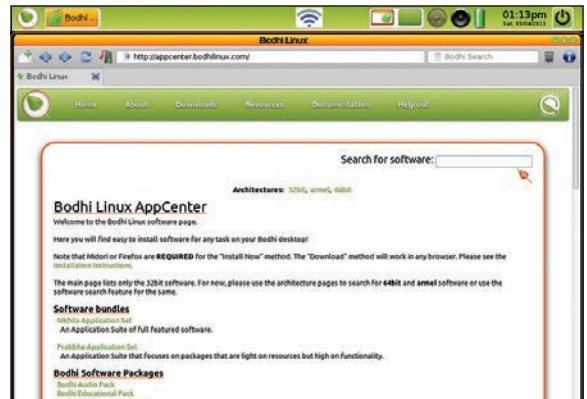
Для большинства дистрибутивов в нашем Сравнении выбор программ по умолчанию зависит от рабочего стола. Поэтому в Ubuntu Gnome вы найдете приложения Gnome, такие, как *Evolution*, *Shotwell*, *Rhythmbox* и *Totem*, а в Kubuntu — приложения KDE, такие, как *Kmail*, *Amarok* и *Dragon Player*.

Хотя оба эти дистрибутива намерены хранить верность своему рабочему столу, есть и исключения. Самое примечательное — *LibreOffice*, пакет по умолчанию в Ubuntu Gnome, Kubuntu и Zorin. В легковесном Lubuntu вместо него *AbiWord* и *Gnumeric*, а в Bodhi — лишь базовый текстовый редактор. По сути, единственное готовое приложение в Bodhi — web-браузер *Midori*, через который можно получить доступ к онлайн-магазину приложений Bodhi и скачать новые приложения. Интересно, что Google Docs считает *Midori* устаревшей версией *Google Chrome*. Zorin идет с настоящим проприетарным *Google Chrome*, а Lubuntu включил его брата-близнеца с открытым кодом, *Chromium*. Вместо браузера Gnome, *Epiphany* (который теперь перекрестили в *Web*), Ubuntu

Gnome предлагает *Firefox*. Kubuntu идет с собственным браузером, *Reconq*, хотя в нем есть ссылка на установку *Firefox*.

По контрасту с Bodhi, другой легковесный дистрибутив, Lubuntu, прямо-таки набит приложениями. В нем несколько приложений *GTK* — программа просмотра документов *Evince*, *Archive Manager* и редактор изображений *mtPaint*; а также многофункциональные приложения, поставляемые с рабочим столом *LXDE*, например, текстовый редактор *Leafpad* и *PCManFM*. В Lubuntu еще имеются клиент электронной почты *Sylpheed* и браузер *Chromium*, плюс *Audacious* и *Gnome MPlayer*, и вы сможете воспроизвести большинство популярных форматов: MP3, AVI и MP4 и т.д. Ubuntu Gnome также позволяет воспроизвести MP3 с помощью *Rhythmbox*. В Kubuntu плагинов нет, но его приложения мультимедиа умеют скачивать кодеки. Лучшее в производных Ubuntu то, что при установке дистрибутива можно добыть проприетарные плагины и кодеки для воспроизведения закрытых форматов.

Zorin — единственный дистрибутив в нашем Сравнении, который позволяет



» Bodhi предоставляет лишь минимальный рабочий стол, но наполнить его приложениями не будет очень уж тяжким трудом.

просматривать содержание в проприетарных форматах с рабочего стола в режиме live. При размере 1,5 Гб, Zorin — самый увесистый из всей компании. Помимо вышеупомянутых приложений, он включает *GIMP*, *Shotwell*, *Google Chrome*, *Gwibber*, *Thunderbird*, *Empathy*, *Totem*, *Rhythmbox*, *VLC* и видеоредактор *OpenShot*, а также *Wine*, для установки приложений Windows.

Во всех случаях, кроме Zorin, вскоре после установки вам придется нанести визит в приложение по управлению пакетами. В Bodhi это первое, что вам придется сделать; а Lubuntu, Ubuntu Gnome и Kubuntu с их выбором приложений по умолчанию дают вам неплохие возможности.

Вердикт

- Zorin ★★★★★
- Lubuntu ★★★★★
- Kubuntu ★★★★★
- Ubuntu Gnome ★★★★★
- Bodhi Linux ★★★★★

» Сразу после установки пользователь больше всего получит в Zorin.

Заявленная цель

Хорошо ли они выполняют обещанное?

Выбранные нами дистрибутивы предоставляют GUI, отличные от Unity, но каждый из них делает достойную работу для своей целевой аудитории.

Цель Ubuntu Gnome — обеспечить относительно чистый рабочий стол Gnome.

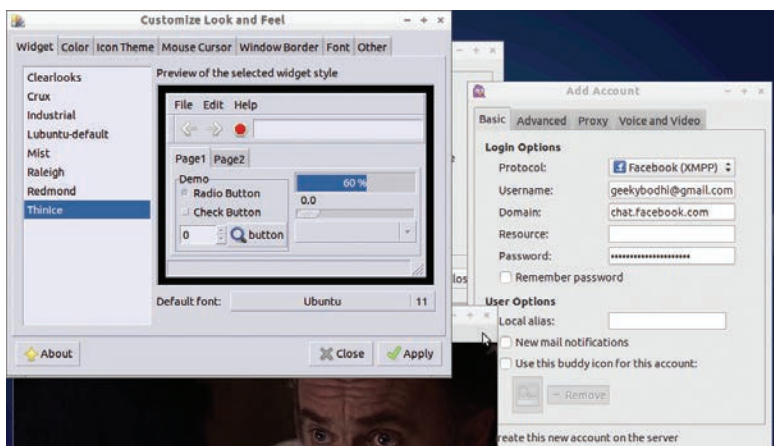
Поскольку основной дистрибутив Ubuntu по-прежнему применяет библиотеки из Gnome 3.6, Ubuntu Gnome 13.04 идет с более старым релизом Gnome, и самую свежую версию Gnome придется устанавливать вручную из Person Package Archive

(PPA). Пользователи также лишатся новых приложений Gnome, *Voxes* и *Web*, так как те зависят от библиотек из последнего релиза. Kubuntu делает для KDE то, что Ubuntu Gnome делает для Gnome Shell. Kubuntu отлично интегрирует рабочий стол KDE, но все же не включает всех приложений, созданных проектом KDE — в частности, *Calligra Office Suite*.

Разработчики Lubuntu хотели создать нетребовательный к ресурсам дистрибутив, и *LXDE* — именно то, что надо. Lubuntu также содержит программы вроде *MPlayer*, который намного удобнее своих сородичей.

Bodhi Linux тоже легковесный дистрибутив, но он поощряет пользователей к индивидуальной настройке системы с помощью своей системы управления пакетами.

И, наконец, Zorin OS разработан для пользователей Windows, и отлично справляется со своей задачей благодаря своему интерфейсу по умолчанию и настраиваемому приложению *Look Changer*.



» Это многозадачность Lubuntu на старом ноутбуке Celeron!

Вердикт

- Bodhi Linux ★★★★★
- Lubuntu ★★★★★
- Zorin ★★★★★
- Kubuntu ★★★★★
- Ubuntu Gnome ★★★★★

» Ubuntu Gnome всегда будет зависеть от своего родительского дистрибутива.

Настольная деятельность

В каком дистрибутиве самая удобная работа, если без Unity?

До появления Unity Ubuntu был самым популярным дистрибутивом Linux для пользователей настольных систем — и на то были веские причины: лучшая в своем классе программа установки *Ubiquity*, а также удобная система управления пакетами, позволяющая обновлять систему одним щелчком.

Все дистрибутивы в нашем Сравнении восходят к одному и тому же предку. И хотя некоторым из них для нормального функционирования требуется больше ресурсов, от Ubuntu и друг от друга они отличаются только внешним видом и способом работы с ними. Это также главная причина, по которой вы можете предпочесть производное

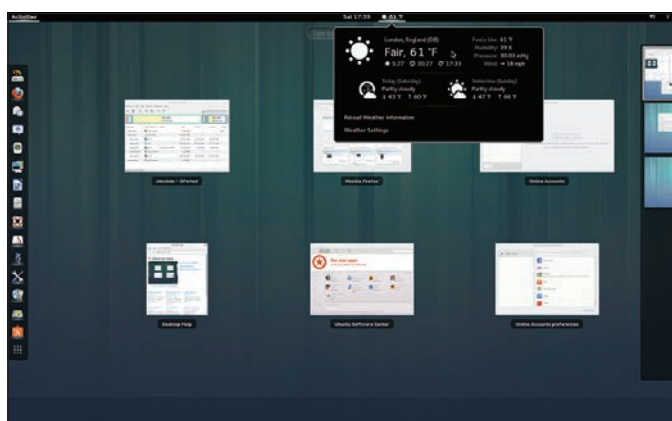
от Ubuntu самому прародителю. Вопросы вот в чем: подходят ли эти производные для новых пользователей Linux, пришедших с других ОС, таких, как Windows и Mac OS? А если вы — уже пользователь Linux, у которого за плечами Ubuntu, какой именно дистрибутив-производное вернет вас в эру до Unity?

Ubuntu Gnome ★★☆☆☆☆

Если вы испытали горькое разочарование от Unity, то среда рабочего стола Ubuntu Gnome вряд ли сделает вас значительно счастливее. Gnome 3 и Unity во многом похожи, хотя и различаются по своей реализации.

Весьма сомнительно, что тем, у кого вызывает отвращение вертикальная программа загрузки Unity, придется по душе практически такая же программа загрузки в Gnome. Да вдобавок в Unity-то программа запуска всегда находится в пределах видимости, а в Gnome, чтобы ее увидеть, придется вызывать Activities View.

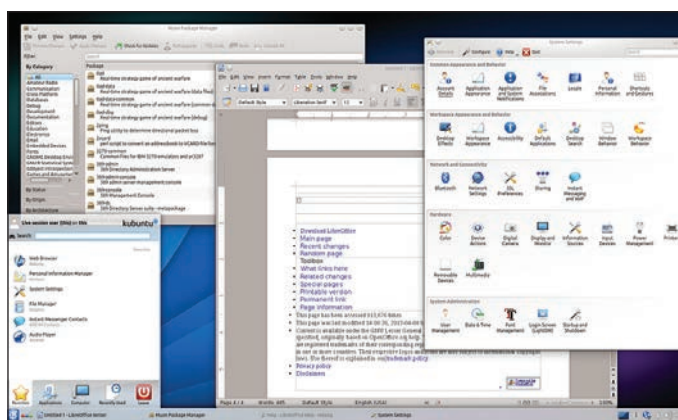
Среди прочих недостатков Gnome — минимальная фурнитура окон и неспособность создавать значки рабочего стола, что превращает его в дистрибутив, способный удовлетворить существующих пользователей Gnome только чисто теоретически. Да и тем он, скорее всего, не особенно понравится, поскольку поставляется с устаревшим Gnome 3.6.



Kubuntu ★★☆☆☆☆

На первый взгляд, KDE — вылитый Windows: панель а-ля системный лоток внизу экрана, меню запуска в углу, значки быстрого запуска и область уведомления со значками системного лотка. Однако представлять KDE новичкам просто как «внешне похожий на Windows» означает оказать ему медвежью услугу. Его истинная мощь кроется в изобилии опций настройки и функций типа Activities; но пользователи, к сожалению, сочтут их весьма запутанными, а в худшем случае вообще проигнорируют. Activities — это настраиваемые рабочие области или виртуальные рабочие столы, которые можно установить и переключаться между ними в зависимости от рода работ: например, чтобы рабочий стол был готов к web-разработке и все приложения были открыты.

Kubuntu — самая старая разновидность Ubuntu, с активным сообществом, и этот дистрибутив подходит пользователям, обжегшимся на Unity; однако здесь нет ряда лучших функций Ubuntu, например, Software Center.



Системные требования

Какое оборудование вам понадобится?

Из-за различия системных требований разных рабочих столов некоторые участники Сравнения не будут работать на всех видах оборудования.

Дистрибутив с самыми высокими запросами в нашем Сравнении — Ubuntu Gnome, по причине настоятельного требования ускоренной графики.

Зато KDE, а значит, и Kubuntu, можно запустить на компьютере без отдельной графической карты. Эффекты при настройках по умолчанию остаются только

самые примитивные, но рабочий стол сохранится полнофункциональным.

Если вам нужен дистрибутив для старой машины, ни Ubuntu Gnome, ни Kubuntu не сравнятся с Lubuntu или Bodhi Linux.

Для продуктивного использования основанного на LXDE Lubuntu вам потребуется как минимум 512 МБ ОЗУ. Проект также предоставляет специальные установочные ISO для маломощных компьютеров с ОЗУ менее 700 МБ.

Bodhi Linux еще более снижает эту планку. Его можно установить на систему

всего со 128 МБ ОЗУ, и он займет на жестком диске скромные 2,5 Гб. Прелесть Bodhi в том, что его легко настроить и под более современную машину: достаточно будет скачать через его менеджер пакетов полнофункциональные приложения вместо легковесных.

Zorin OS предоставляет вам лучшее из обоих миров. Обычная версия основана на Gnome и требует такого же количества ресурсов, что и Kubuntu, но есть также основанная на LXDE редакция Lite для более старых компьютеров.

Вердикт

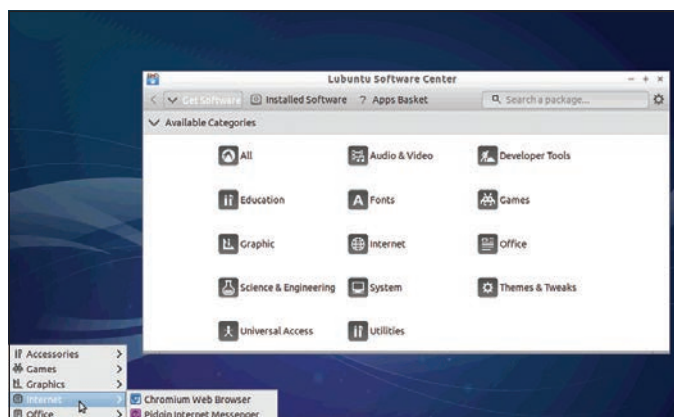
- Bodhi Linux ★★★★★
- Lubuntu ★★★★★
- Zorin ★★★★★
- Kubuntu ★★★★★
- Ubuntu Gnome ★★★★★

» Bodhi Linux может превратить большинство машин в ПК общего назначения.

Lubuntu ★★★★★

Подобно KDE, рабочий стол Lubuntu, LXDE, внешне напоминает Windows, с панелью внизу. Однако меню имеет сходство не с чем иным, как с Windows 98. Если хорошенько подумать, то подобная отсталость не очень-то способна соблазнить пользователей Windows 7/XP, особенно с учетом того, что в LXDE отсутствуют другие функции Windows — например, «живой» просмотр миниатюр.

Вдобавок в Lubuntu не предусмотрено Ubuntu One, да и интеграция с PCManFM и Sylpheed пока что остается в списке планов разработчиков. Но Lubuntu и предназначен для другой пользовательской аудитории: для тех, кто хочет получить Ubuntu-подобный рабочий стол на старой и маломощной машине. И с этим Lubuntu справляется идеально. А тот факт, что он предоставляет симпатичную среду рабочего стола, является дополнительным бонусом.



Bodhi Linux ★★★★★

Также минималистский, дистрибутив Bodhi Linux основан на элегантном Enlightenment Window Manager и предлагает свой собственный менеджер файлов, несколько гаджетов и композиционные эффекты. Bodhi поможет вам не только вернуть к жизни списанную за малосильность машину, но и сделать ее элегантной.

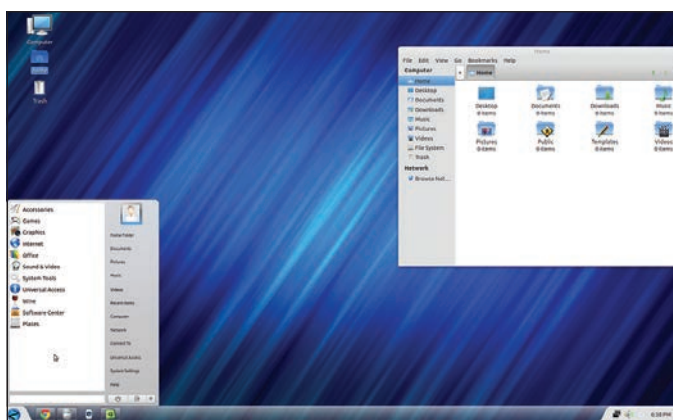
Дистрибутив также предлагает несколько стилей раскладки рабочего стола: Desktop — это традиционная раскладка с меню, панелью задач и системным лотком внизу; Laptop/Netbook размещает лоток, меню и панель вверх, вместе с гаджетами, спустив программу запуска приложений вниз; а Fancy размещает программу запуска приложений в левой части экрана.

Есть также стиль для небольших устройств и устройств с сенсорным экраном, использующий меню приложений, которое напоминает Unity Dash или Gnome 3 Activities.

Zorin OS ★★★★★

Это, несомненно, лучший дистрибутив, который предлагает возможности рабочего стола как для линуксоидов, так и для тех, кто пришел с других операционных систем. И дело не только в том, что рабочий стол Zorin по умолчанию напоминает Windows 7, а для усиления этого сходства его программа запуска приложений имитирует меню Start из Windows 7. Существующие пользователи Linux могут воспользоваться приложением Zorin Look Changer, чтобы рабочий стол приобрел внешний вид и повадки Gnome 2. Если вы не пожалеете €10, можете скачать Ultimate edition, где есть дополнительные стили Mac OS X и Windows 2000.

Zorin также включает все прелести Ubuntu, типа Ubuntu One, который здесь отлично интегрирован. Он также внедряет хорошие методы работы, регулярно напоминая пользователям о необходимости настраивать приложение резервного копирования.



Документация и поддержка

Куда податься за помощью?

Одна из причин популярности Ubuntu — инфраструктура поддержки и активное сообщество пользователей. Ну, а как обстоят в этом плане дела у его разновидностей?

У Ubuntu Gnome растущее сообщество с собственным каналом IRC и списком рассылки; форумов у него пока нет, но подсказки пользователям регулярно размещаются на официальной странице сообщества в Google+.

У Kubuntu, старейшей разновидности Ubuntu, весьма активное сообщество: оно

ведет форумы по KDE, списки рассылки и канал IRC. Также дистрибутив предлагает информацию по получению помощи на нескольких языках, а не только по-английски, и всестороннее руководство в формате ebook на ubuntuguide.org.

Lubuntu размещает документацию (по приложениям и настройке компонентов) и поддержку в инфраструктуре Ubuntu. Ссылка на форумы отобразит все посты с пометкой «lubuntu» на ubuntuforums.org. Можно общаться с разработчиками Lubuntu через списки рассылки и в IRC.

Руководство по установке Zorin OS пребывает в зачаточном состоянии, хотя у них есть форумы с разделом, посвященным руководствам и справочникам. Заплатив €5, вы получите возможность три раза обратиться за технической поддержкой.

Bodhi Linux имеет лучше всего организованную поддержку и документацию. Здесь есть руководство для начинающих и подробные руководства по Enlightenment и индивидуальной настройке Bodhi. На его форумах даже есть советы касательно устройств ARM.

Вердикт

- Bodhi Linux ★★★★★
- Kubuntu ★★★★★
- Lubuntu ★★★★★
- Ubuntu Gnome ★★★★★
- Zorin ★★★★★

» Вам нечего опасаться насчет поддержки для любого дистрибутива Сравнения.

Инструменты настройки

В каждом ли дистрибутиве инструменты правильно заточены?

Как мы уже сказали, от несметного множества других дистрибутивов на базе Ubuntu наших участников отличает старание разработчиков заставить дистрибутив соответствовать его пользовательской базе.

Взять, например, Zorin. Он разработан специально для пользователей Windows, которым нужен безболезненный переход к Linux, почему и включает приложение *Look Changer*, позволяющее одним щелчком изменять интерфейс: в доступной для скачивания свободной редакции ОС приложение предлагает выбрать между

Windows 7, XP или Gnome 2. Не пожалев €10 на Ultimate edition, вы получите также поддержку для Mac OS X, Unity и Windows 2000. Менеджер web-браузеров Zorin облегчает установку разных браузеров, и дистрибутив включил также переделанный Ubuntu Software Center.

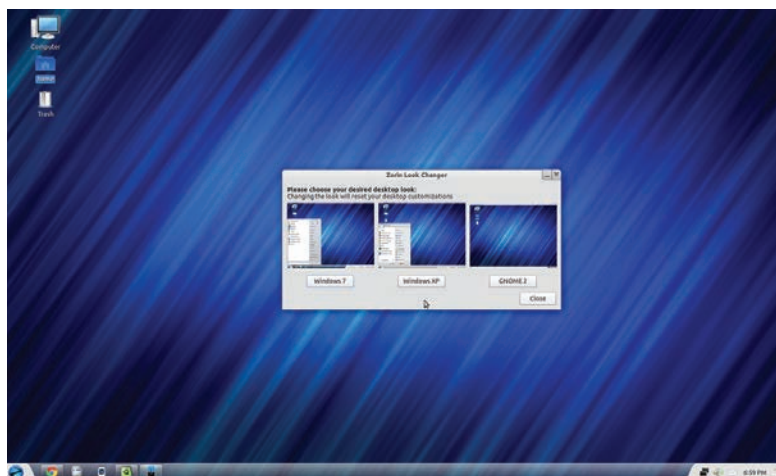
У Bodhi Linux во главе угла настраиваемые инструменты. Он предлагает eScess System Tool для базовых задач по администрированию — таких, как управление пользователями и временем. На форуме Bodhi Linux есть ветка, посвященная настраиваемым инструментам, написанным

на Elementary и Python, которые находятся в официальных репозиториях. Еще один важный момент — онлайн-вый App Center, который облегчает установку приложений, группируя их в пакеты и создавая пакеты из похожих приложений. Например, желая установить набор образовательных приложений, можно выбрать Educational Pack, который включает *TuxPaint*, *TuxTyping*, *GCompris*, и т.п.

Разработчики Lubuntu решили проблему управления пакетами, создав облегченную версию Ubuntu Software Center под названием — угадайте! — Lubuntu Software Center. Он размещает программы по разным категориям, вы отмечаете программы, которые хотите установить, добавляете их в корзину программ и устанавливаете все разом. Есть также режим Эксперт для установки отдельных библиотек.

Kubuntu также включает собственный менеджер пакетов, *Moum*, и программу запуска, *HomeRun*, которая работает в полноэкранном режиме и похожа на Dash в Unity и Activities в Gnome.

Только в Ubuntu Gnome нет настраиваемых инструментов: он включил Ubuntu Software Center, но оставил за бортом некоторые инструменты Gnome, например, браузер и приложение виртуализации *Voxes*. Вящую путаницу вносит тот факт, что в дистрибутиве имеются два приложения User Accounts, из Gnome и из Ubuntu.



➤ Zorin использует *Look Changer*, чтобы имитировать рабочие столы других ОС.

Вердикт

Bodhi Linux ★★★★★

Zorin ★★★★★

Lubuntu ★★★★★

Kubuntu ★★★★★

Ubuntu Gnome ★★★★★

» Легковесный Bodhi Linux предлагает массу инструментов, как мы и ожидали.

Опции настройки

Легко ли настраивать эти дистрибутивы?

Большинство дистрибутивов перенесли задачу настройки рабочего стола в GUI. И это большой недостаток Ubuntu Gnome: для среднего пользователя Linux его опции настройки по умолчанию сгодятся, но пользователям продвинутого уровня придется искать дополнительные инструменты, чтобы построить свой рабочий стол.

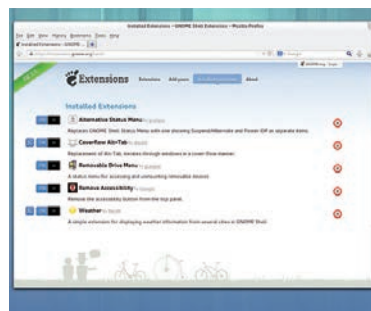
На другом конце спектра располагается KDE со своим необъятным набором опций настройки, который может показаться просто угнетающим. В Lubuntu тоже полно опций настройки. Здесь есть менеджер настройки *Openbox* и программа индивидуальной настройки от проекта *LXDE*. Есть также разные приложения для модификации клавиатуры, мыши, монитора

и управления энергопотреблением. В противоположность ему, Kubuntu не предлагает собственного набора инструментов настройки типа *Yast* от OpenSUSE или *Control Center* в Mageia.

Bodhi Linux позволяет выбрать тему и обои для изменения вида рабочего стола. Можно также загрузить разные гаджеты рабочего стола, например, батарейку и часы. Здесь есть панель настройки, пригодная, чтобы поменять обои и тему рабочего стола и приложений, а также настроить несколько рабочих столов, индивидуально настроить меню, программы запуска, менеджер файлов, управление энергопотреблением и т.д.

Zorin — единственный здесь дистрибутив, который предлагает инструмент

Ubuntu Tweak и использует приложение брандмауэра *Ufw*, настраиваемое через интерфейс *gufw*.



➤ Вы можете с легкостью настроить Gnome 3 благодаря его обширной библиотеке расширений.

Вердикт

Bodhi Linux ★★★★★

Kubuntu ★★★★★

Lubuntu ★★★★★

Zorin ★★★★★

Ubuntu Gnome ★★★★★

» Bodhi Linux и Lubuntu предлагают должное количество опций настройки.

Дистрибутивы на базе Ubuntu

Вердикт

Все дистрибутивы в нашем Сравнении используют разные рабочие столы, да и имеют весьма различные целевые аудитории. Тогда было бы логично для пользователей KDE использовать Kubuntu; а если кому нужен дистрибутив для старой машины, пылящейся в кладовке, им стоит выбрать Lubuntu.

Увы, эта логика неприменима к Ubuntu Gnome. Нынешний релиз дистрибутива (13.04) страдает из-за того, что его родитель по-прежнему использует устаревший релиз Gnome 3.6, и он вынужден делать то же самое. И хотя вряд ли это помешает любителям Gnome и Ubuntu вручную скачать самый свежий релиз рабочего стола Gnome из PPA, мы не можем порекомендовать готовый дистрибутив неопытным пользователям Linux.

Готовый релиз Bodhi Linux также заставит вас прогуляться в менеджер пакетов, однако онлайн-магазин приложений данного дистрибутива и его система

управления пакетами делает этот процесс очень простым. Плюс к тому, Bodhi предоставляет на выбор несколько раскладок рабочего стола, а благодаря минимальным требованиям дистрибутива к оборудованию вы сможете работать с ним практически на любом компьютере.

Портрет убийца

Победителем этого сравнения, причем с большим отрывом, становится Zorin OS (названный в честь Макса Зорина [Max Zorin], злодея из фильма о Джеймсе Бонде). Zorin успешно применил ядро Ubuntu, создав готовый к использованию настольный Linux, который также подойдет пользователям, решившим отказаться от других операционных систем.

Внешний вид по умолчанию Zorin имитирует Windows 7; фанаты Gnome 2 могут изменить рабочий стол, чтобы он выглядел и работал, как их любимец, и сделать это одним нажатием кнопки.



Дистрибутив предлагает приложения для среднестатистического пользователя настольного ПК и свободную редакцию Core edition, а также производит несколько премиум-редакций (стоимость от €7) для специальных целей. Например, Business edition снабжена инструментами для бухучета, ведения бухгалтерии, анализа активов и т.д. Среди других премиум-редакций — Gaming и Multimedia, а Ultimate edition содержит — как и следовало ожидать — все инструменты, включенные в другие редакции.

» Разработчики трудятся над Zorin 7, который, по их словам, получит новый дизайн и будет эстетически привлекательнее.

«Zorin OS применил ядро Ubuntu, создав готовый настольный Linux.»

I Zorin OS ★★★★★
 Сайт: www.zorin-os.com Лицензия: GPL и др.
 » Безусловно, самая полная и симпатичная разновидность Ubuntu.

IV Lubuntu ★★★★★
 Сайт: www.lubuntu.net Лицензия: GPL и др.
 » Легковесный дистрибутив, идеальный для реанимации устаревших машин.

II Kubuntu ★★★★★
 Сайт: www.kubuntu.org Лицензия: GPL и др.
 » Пожалуй, лучший для KDE-поклонников, но при наличии опыта в Linux.

V Ubuntu Gnome ★★★★★
 Сайт: www.ubuntu gnome.org Лицензия: GPL и др.
 » В своем нынешнем виде мало что может предложить даже гномофилам.

III Bodhi Linux ★★★★★
 Сайт: www.bodhilinux.com Лицензия: GPL и др.
 » Хороший минималистский дистрибутив, предлагающий разные раскладки.

Обратная связь
 Вы согласны с нашим выбором? Возможно, вы бы взяли совсем другие дистрибутивы? Присылайте нам свои мнения на lxfl.letters@futurenet.co.uk.

Рассмотрите также...

Ubuntu более популярен как основа специализированных, а не общецелевых дистрибутивов. Edubuntu предлагает образовательные приложения; Mythbuntu предназначен для создания НTPС с помощью *MythTV*; Ubuntu Studio — для создания и редактирования мультимедиа; BackBox — для тестирования возможности взлома; Bio Linux и Poseidon Linux — для научного сообщества.

Нужна настольная система общего назначения? Тогда самой популярной разновидностью Ubuntu будет Linux Mint: если верить DistroWatch, он превзошел самого Ubuntu, благодаря привычному рабочему столу и инструментам настройки.

Пуристам свободного ПО стоит обратить внимание на проект Trisquel Linux, который прилагает немалые усилия, чтобы создать на базе Ubuntu

дистрибутив, соответствующий суровым требованиям Free Software Foundation (FSF).

Одна из не попавших сюда официальных разновидностей — Xubuntu, на базе *Xfce*. Он позиционируется как легковесный, но заодно тянет лапы к старым машинам. А еще есть PinguinOS с настраиваемым рабочим столом Gnome; в нем будут как дома пользователи Mac OS X. **LXF**

Сбежим от Google

и от Facebook » Twitter » Flickr и прочих!



Вырвите свои личные данные из-под гнета компаний, похищающих вашу почту, фотографии и друзей, и сами рулите своей жизнью.

Поиск в Google

Пойду-ка я отсюда

Мы все активнее употребляем облачные сервисы. По мере роста скорости и надежности интернет-соединения все привлекательнее становится мысль о том, чтобы иметь под рукой все наши данные на всех наших устройствах. Однако у облачных сервисов есть и недостатки, в особенности у бесплатных, которым приходится как-то зарабатывать на жизнь.

Есть ряд важных, хотя и жутковатых вопросов, которые стоит задать себе, прежде чем решиться доверить свои данные посторонним, и мы перечислили их ниже. И, как вы увидите из нашей статьи, ответы на них частично вовсе не те, какие вам хотелось бы услышать:

» **Приватность** — ваши данные просто хранятся или используются для рекламных и маркетинговых целей?

» **Надежность** — есть ли уверенность в том, что выбранный вами сервис всегда будет доступен? Какие у вас гарантии безопасности ваших данных, и можете ли вы что-то сделать для улучшения существующей ситуации?

» **Безопасность** — зашифрованы ли ваши данные? Кто имеет доступ к ключам шифрования? Можно ли изменить или похитить ваши данные?

» **Долговечность** — может ли провайдер облачного сервиса в силу неких причин приостановить или закрыть вашу учетную запись-аккаунт, возможно, уничтожив все ваши данные?

» **Производительность** — достаточно ли скорости вашего интернет-соединения, чтобы использовать нужные вам сервисы без задержек?

«Какие у вас гарантии безопасности ваших данных?»

» **Копирайт** — кто владелец загруженного вами контента? Можно ли опубликовать или продать ваши фотографии без вашего согласия?

Хотя в крупной компании политика безопасности и резервного копирования явно лучше организована и реализована, чем ваша, все же остаются

вопросы насчет того, что они могут делать с вашими данными. Быть может, вы решите, что разрешение на разбор вашей электронной почты для рассылки более адекватной рекламы, которую вы можете и проигнорировать — это нормальная издержка бесплатного сервиса. С другой стороны, вам, возможно, нужна конфиденциальность своей электронной почты — что еще важнее, если вы используете ее в своем бизнесе: тогда вы отвечаете не только за свои данные, но и за данные организаций, с которыми работаете. Также вас, видимо, обеспокоит сохранение конфиденциальной информации внутри вашей частной сети, не только в целях защиты коммерческой тайны, но и для сохранения конфиденциальности данных, касающихся партнеров организации.

Один закон для нас, другой для них

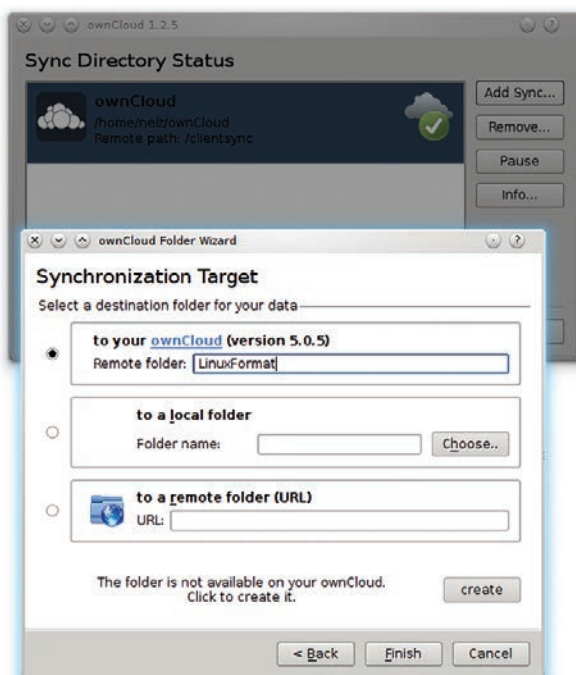
Еще одна серьезная проблема заключается в том, что облачный сервис может находиться не в вашей стране и, следовательно, подчиняться чужим законам. Даже если вас вполне устраивает его

местоположение, ничто не может помешать провайдеру сервиса все переместить в дата-центр третьей страны по неким экономическим или политическим причинам — или просто чтобы отвертеться от уплаты налогов. Google немало претерпел из-за своего подхода к политике конфиденциальности (не говоря уже о его явном убеждении, что уплата налогов в Великобритании — дело сугубо добровольное), однако здесь он не одинок, и даже не самый худший. Просто Google — провайдер высшего ранга. К его чести скажем, что он в разумных пределах открыт насчет употребления им ваших данных ради получения прибыли: обзаводясь учетной записью на Gmail, большинство из нас в курсе, что Google будет читать наши сообщения электронной почты.

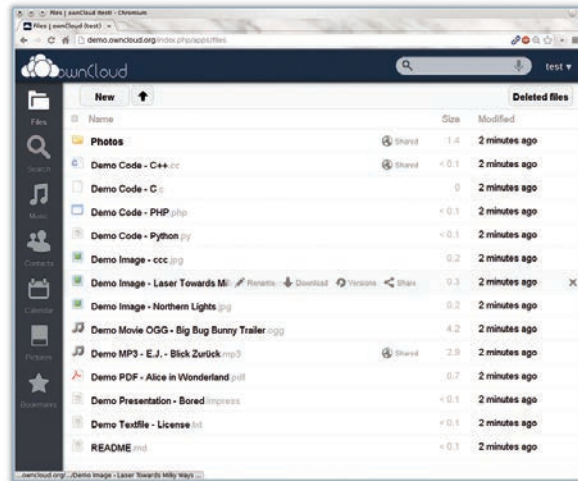
И невзирая на это, идея облачного сервиса выглядит удобной и привлекательной; как же нам разрешить эту двойственность? Ответ — использование своего личного облачного сервиса. В таком случае вы получаете прямой доступ к своим данным с любой платформы и из любого места, причем сами контролируете свои данные. Нет, мы не предлагаем вам создать собственный центр обработки данных, реализовав личные *Gmail*, *Dropbox* и *Facebook*, но вам и не требуется масштаб этих организаций для дома или малого бизнеса.

Что вам нужно?

Обычно облако базируется на Сети, а Linux обладает определенной репутацией в сфере предоставления web-услуг. Все, что вам нужно — это компьютер Linux, на котором работает стандартный LAMP (Linux, Apache, MySQL, PHP), и вы сможете за считанные минуты установить большинство программ со следующих страниц. Остается ответить всего на два вопроса: что и где. В зависимости от объема намеченных вами работ, оборудованием может послужить небольшая машина-сервер, ваш настольный компьютер, постоянно включенный во все требуемое время, или нечто действительно небольшое и эффективное, вроде Raspberry Pi или какого-либо «компьютера в розетке [plug computer]». Все перечисленные устройства можно собрать в локальную сеть, и вы получите отличную скорость при использовании внутри LAN. К ним можно также получить доступ и вне пределов LAN — здесь встанет вопрос о скорости вашего интернет-соединения. Большинство



» На сайте предлагается два способа установки *ownCloud*; простейший из них — скачать и распаковать tar-архив, если только в вашем дистрибутиве нет такого пакета.



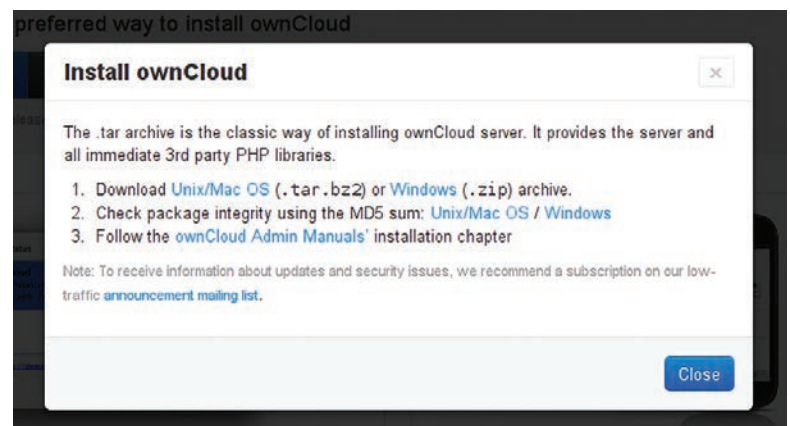
» Демо на <http://demo.owncloud.org> позволяет побаловаться с *ownCloud* до того, как установить его.

домашних интернет-соединений являются асимметричными, то есть предоставляют большую скорость на скачивание и меньшую — для выгрузки. Доступ к данным из внешнего мира считается выгрузкой, и он будет медленнее — в общем не запретительно, но все же об этом стоит помнить, особенно если вы собираетесь поддерживать несколько пользователей. Если же вы предоставляете доступ из мира в целом, то ваша жизнь намного упростится, если ваш интернет-провайдер предоставит вам статический IP-адрес; в противном случае придется обратиться к сервису динамических DNS.

Как альтернатива, сгодится VPS, Virtual Private Server. По сути это сервер под Linux, работающий на виртуальной машине в центре обработки данных, и вы получите все преимущества высокой скорости, предоставляемой подобными средами, а система остается вашей собственной, и вы можете контролировать применяемые вами программы, а также все пароли и ключи шифрования, и провайдеру вовсе незачем быть в состоянии читать ваши данные. Можно также использовать для этого web-хост, поскольку все работает на LAMP, но это сократит опции защиты ваших данных.

Какие бы программы вы ни выбрали, реальные или виртуальные, вам нужна полная установка LAMP. Если вы выделяете для этой цели весь компьютер, то проще всего будет установить на нем сервер-ориентированный дистрибутив, например, Ubuntu Server, или нечто вроде Debian, выбрав при установке опцию сервера. На уже готовой системе должно быть вполне достаточно *Apache*, *MySQL* и *PHP*. Если вы установите обсуждаемые здесь программы через менеджер пакетов своего дистрибутива, он сам позаботится о них.

»



» Клиент синхронизации рабочего стола позволяет задать несколько задач синхронизации, выполняемых по очереди — вы не ограничены одной директорией.

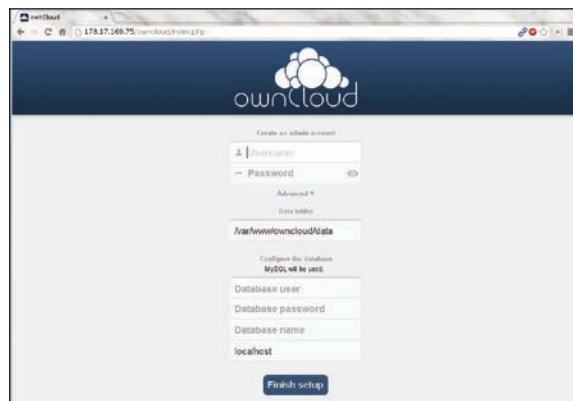
ownCloud

Обезопасьте свои данные — настройте и запустите личный облачный сервер.

Причин, почему вам может (или не может) понадобиться собственный облачный сервис, предостаточно; но как вам его создать? «Облако» — довольно широкое понятие, которое включает практически любой сервис, позволяющий хранить данные «где-то», получая доступ к ним через Интернет. Есть сервисы электронной почты, например, *Gmail*, сервисы хранения данных типа *Dropbox*, фотогалереи, как, скажем, *Flickr*... и далее по списку. Не исключено, что вы решите не загонять все в личное облако, поскольку социальный аспект сайтов а-ля *Flickr* делает публичность их сильной стороной — особенно учитывая, что сейчас *Flickr* предоставляет 1 ТБ бесплатно; но нужно это не всегда. Так откуда же начать, какой сервис приватизировать первым? Среди проектов, охватывающих более одной из этих областей — *ownCloud* (<http://owncloud.org>). В основном он сосредоточен на хранении данных, в стиле *Dropbox*, но также предоставляет ряд других функций — часть уже встроена, часть доступна через модули расширения. Если вы хотите побаловаться с *ownCloud*, прежде чем приступать к его установке, попробуйте демо-версию на <http://demo.owncloud.org>.

Это web-приложение, и для начала стоит убедиться, что ваш web-сервер работает. Если на вашем компьютере есть рабочий стол, просто проверьте, работает ли в браузере <http://localhost>. Если вы делаете установку на отдельный сервер, используйте вместо localhost его имя хоста. Самая свежая версия *ownCloud* — 5.0.6. Раздобыть *ownCloud* можно тремя способами. Если он есть в репозиториях вашего дистрибутива, установите его через свой менеджер пакетов. В качестве альтернативы можно скачать самый новый tar-архив с <http://owncloud.org/install/> и распаковать его в web-root. Это директория, заданная как DocumentRoot в конфигурации вашего *Apache*. Обычно это нечто вроде `/var/www/` или `/var/www/htdocs`, хотя некоторые дистрибутивы используют `/srv/www` вместо `/var/www`. Третий способ — web-установка, предназначенная для удаленной работы, например, на VPS. Скачайте

➤ В случае *MySQL* нужно создать базу данных и пользователя. Это делается из командной строки или через *phpMyAdmin*.



➤ Создайте пользователя *admin* и нажмите на кнопку **Advanced**, чтобы настроить полномочия базы данных.

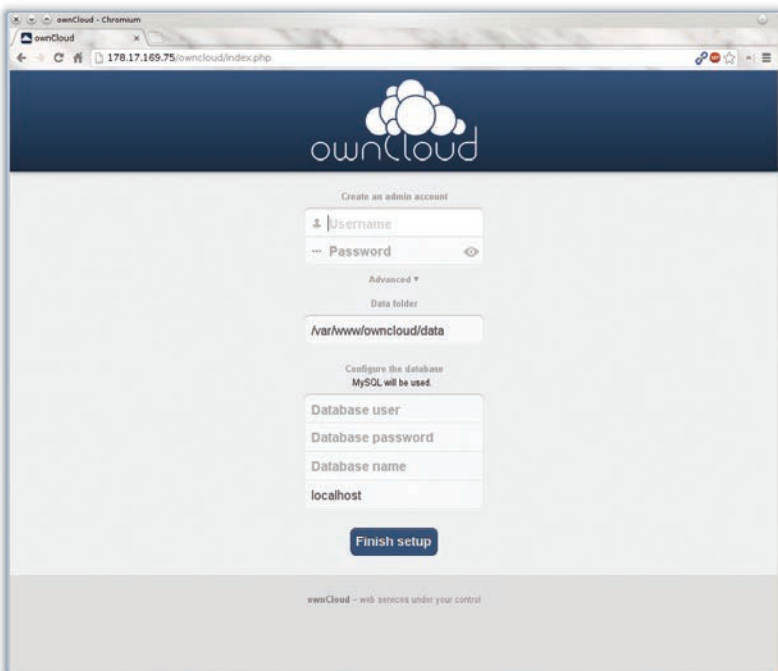
web-программу установки, которая на данный момент находится на <http://bit.ly/12pWiUW>, в свое web-пространство, затем используйте свой браузер, чтобы открыть файл PHP.

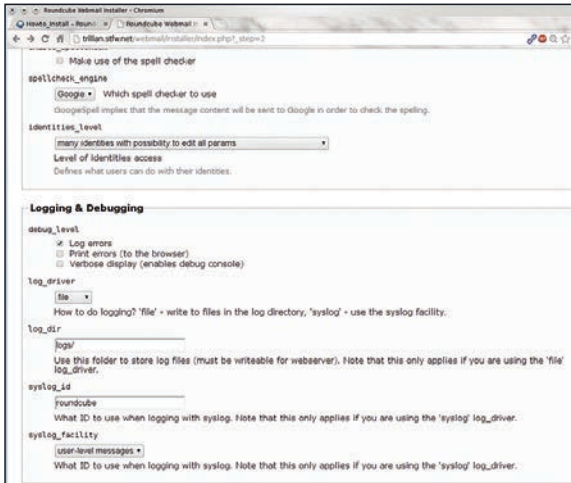
Каким бы способом вы ни установили его, теперь у вас должна быть директория для *ownCloud* в web-root, так что введите в свой браузер <http://hostname/owncloud>, и он загрузит начальную страницу настроек. Главное решение, которое следует здесь принять — выбор движка базы данных. *SQLite* будет самым простым и отлично подойдет для систем с одним пользователем. Однако если хоть что-то у вас использует *MySQL* — а кое-каким пакетам, о которых мы поговорим далее, это нужно — то имеет смысл работать только с одной базой данных и использовать *MySQL* для всего. В любом случае, надо дать базе данных имя и подробную информацию о пользователе. База данных *SQLite* создается автоматически, но пользователям *MySQL* придется создать ее до настройки *ownCloud*, о чем будет рассказано далее в разделе *MySQL*.

Синхронизация

Когда вы введете подробную информацию об администраторе и базе данных, *ownCloud* настроится и будет готов к работе. Но прежде чем приступить к действиям, вы, возможно, захотите подправить еще кое-что. Одно из ограничений *Dropbox* — на размер файлов, там это 300 МБ, а *ownCloud* по умолчанию ставит лимит 512 МБ; но облако-то ваше, и вы вправе изменить эти настройки. Ограничение в 512 МБ идет из настроек по умолчанию в PHP для максимального размера загружаемого файла, и имея скоростную сеть, вы, вероятно, решите изменить его. Щелкните по имени пользователя в правом верхнем углу экрана браузера, чтобы открыть главное меню и выбрать *Settings*. Смените размер загружаемого файла на нужный вам; однако в текущем релизе имеется ошибка (возможно, в PHP), из-за которой возникают проблемы при задании этой величины как 2 ГБ или выше — вы получаете сообщение, что файл слишком велик, хотя это не соответствует действительности. Так что выставляйте лимит максимум в 1,9 ГБ, пока эта ошибка не будет исправлена.

Теперь у вас есть *Dropbox*-подобная область хранения файлов; но как насчет ее синхронизации с вашими устройствами? Для этого нужно установить клиент *ownCloud*. Он имеется для всех популярных ОС, плюс Windows и Mac OS X. Обычно он находится в менеджерах пакетов в виде *owncloud-client*, но некоторые





➤ Возможно, вы решите отключить программу проверки правописания *Roundcube*, иначе Google по-прежнему сможет проверять вашу исходящую почту.

дистрибутивы продолжают называть его *mirall*, это старое имя программы синхронизации. Когда вы его установите и запустите, он разместит значок на вашей панели задач. При первом запуске клиент спросит у вас адрес сервера и директорию, с которой надо наладить синхронизацию. Пока что все работает аналогично *Dropbox*, но здесь есть и другие функции. Вы можете добавить задач по синхронизации, назначив разные директории или даже разные серверы: вы не ограничены синхронизацией единственной директории с единственным сервером. Это не только обеспечивает необходимую гибкость для синхронизации данных из разных папок, но также означает, что можно, например, отдельно синхронизировать рабочие и личные файлы с разными серверами — полная конфиденциальность и множество функций, двойная победа! Есть также клиенты синхронизации для Android и iOS, устанавливаемые обычным способом.

Распределенный доступ к файлам

Как насчет того, чтобы поделиться файлами с кем-нибудь? При наведении мыши на файл в браузере справа появится меню, содержащее опцию Share. Здесь есть поле для адреса электронной почты, куда вы хотите отправить файл, но если в Share отметить поле ссылки, там появится URL, который можно отправить опять же электронной почтой или иным способом перекинуть коллегам. Это ссылка на скачивание, и она не показывает другим содержимое вашего облака, но позволяет им скачать этот файл или просмотреть директорию (и скачать из нее), если вы сделали распределенный доступ к ней. Если у вас несколько пользователей, и вы не хотите, чтобы они делились теми файлами, которыми они делиться не должны, можете объединить пользователей в группы и выбрать опцию на странице администрирования, которая позволит им делиться только с пользователями своей группы. Так вы дадите им возможность сотрудничать, не разглашая коммерческой информации. Пользователи создаются с помощью опции Users в главном выпадающем меню. Кроме распределения их по группам, вы также можете ограничить количество места для хранения, которое им отводится.

Есть еще один способ доступа к вашим файлам — с помощью протокола WebDAV, который поддерживается менеджерами файлов и другими программами. Он позволяет работать со своими файлами в облаке, как если бы они были локальными. Синтаксис,

используемый для этого, варьируется; в *Nautilus* вы будете использовать `dav://your.server/owncloud/files/webdav.php`, KDE предпочитает `webdav://` вместо `dav://`, а *LibreOffice* любит старый добрый `http://`. Да, верно, *LibreOffice* будет загружать файлы прямо из вашего облака. Поскольку WebDAV — протокол двусторонний, он также позволяет сохранять файлы, что сразу же отменяет нужду в отдельном сетевом офисном клиенте; вы можете использовать в своем облаке стандартные инструменты для файлов.

Некоторые модули *ownCloud*, такие, как приложения Calendar, Contacts и Bookmarks, запускаются пустыми, даже если вся эта информация хранится у вас где-то еще. К счастью, импорт из стандартных форматов — ICS, VCF и HTML соответственно — не представляет сложности. Можно также настроить ваш Calendar на синхронизацию с другими устройствами; нажмите на кнопку Settings, чтобы увидеть URL для синхронизации с другими клиентами календаря.

Приложения Music и Photos найдут соответствующие файлы в любой папке вашего облака. Файлы вам не требуется особо тщательно упорядочивать, хотя нет и причин не делать этого. В приложении Music нажмите на кнопку Rescan после загрузки нового файла, чтобы добавить его к списку.

Модули и плагины

Утомившись исследованием имеющихся в *ownCloud* функций, можете добавить еще несколько. В терминологии *ownCloud* они именуется плагины или приложения, и вы можете активировать их, чтобы добавить новые функции. Некоторые из них установлены по умолчанию, некоторые можно скачать с <http://apps.owncloud.com>. Выберите Apps во всплывающем меню, чтобы увидеть, какие плагины установлены. Те,

что выделены жирным шрифтом в начале списка — активированы. Чтобы использовать один из невыделенных, выберите его и нажмите на кнопку Enable. Кнопка More apps пере-

ведет вас на страницу приложений, где можно просматривать или искать приложения. Скачанное приложение будет в виде ZIP-файла, который надо распаковать в директорию приложений вашей системы.

В *ownCloud* есть дополнительные функции, которые еще более его украшают: например, возможность присвоения версий файлам. Если вы переписали файл на более новую версию, старый файл сохраняется, и сохраняется доступ к нему, при наличии достаточного места. Для удаленных файлов есть Корзина [Trash bin], так что любые файлы, удаленные через web-интерфейс, можно восстановить, если вы вдруг передумаете их удалять.

Программа *ownCloud* относительно нова, и релизы выходят густо и часто. В менее зрелом коде велики шансы обнаружения уязвимостей в системе безопасности, так что безопаснее будет постоянно ее обновлять. Страница Admin предупредит вас о наличии версии, более новой, чем ваша. После обновления может обнаружиться, что *ownCloud* застрял в режиме обслуживания. Так сделано для того, чтобы никто не смог получить к нему доступ, пока вы, как администратор, обновляли его. Отредактируйте `config/config.php`, изменив параметр режима обслуживания с true на false. Перезагрузите главную страницу, ответьте на вопросы, которые вам могут задать, и ваше облако снова заработает.



Сбежим от
Google

Webmail

Сохраните конфиденциальность своей почты — настройте свой сервис webmail.

CownCloud можно сделать многое, но один важный облачный сервис здесь не затронут, и это — электронная почта. Из всех облачных сервисов больше всего опасений по поводу конфиденциальности вызывает почта в целом и *Gmail* в частности. Ваши сообщения передаются и хранятся в виде простого текста, и Google открыто признает, что читает вашу электронную почту, чтобы вы получали целевую рекламу. Для большинства это вполне приемлемо, если речь идет о неконфиденциальной почте, но если такое вас не устраивает, создайте собственный сервис сетевой почты — webmail. Это намного лучше: вам даже не придется менять свой электронный адрес или получать статический интернет-адрес, как этого требует собственный сервер SMTP. Программам webmail нет нужды работать на той же системе, что и почтовый сервер, так что можете установить webmail на своем сервере и продолжать использовать свой текущий адрес электронной почты и сервер.

Имеется несколько альтернатив webmail; две самых популярных — *SquirrelMail* и *Roundcube*, обе написаны на PHP и работают на обычном LAMP. Здесь мы рассмотрим *Roundcube*, однако *Squirrel* столь же проста в установке, а функциями не беднее. Ничто не мешает работать и с обеими. В конце концов, это просто клиенты электронной почты, и у немалого количества народу установлен не один такой клиент.

Оба сервиса webmail для общения с почтовым сервером используют IMAP, а значит, сообщения остаются на сервере, и это позволит вам читать любое сообщение и с рабочего стола, и с клиента электронной почты мобильного телефона. У IMAP

есть еще одно преимущество перед старым протоколом POP3: помимо того, что в нем нет необходимости загружать все перед тем, как прочитать, он еще и хранит информацию — например, о том, какие сообщения вы прочитали — на сервере, что позволяет переключаться с одного клиента электронной почты на другой, отслеживая, где вы.

Установка webmail

Установка *Roundcube* проста: или через менеджер пакетов, или напрямую на сервер, скачав tar-архив с www.roundcube.net. Распакуйте архив в корневую директорию своего web-сервера. Получится директория под названием **roundcubemail-0.9.0** (для текущей версии). Переименуйте ее во что-нибудь поудобнее, на-

пример, **roundcube** или даже **webmail**, или сделайте символическую ссылку. *Roundcube* должен писать в файлы своих журналов и временные директории temp, поэтому убедитесь, что ими владеет

пользователь, который запускает web-сервер, обычно apache или www-data. Затем запустите программу установки, введя в браузере <http://address-of-server/roundcube/installer>. Убедитесь, что среди результатов проверки нет Not OK. Отсутствие опциональных модулей — это нормально, вам нужна только база данных (*Roundcube* поддерживает несколько форматов баз данных).

Один из важных параметров, которые он проверяет — опция PHP **date.timezone**; ее надо настроить, чтобы ваши сообщения электронной почты имели правильную отметку времени. Если вы используете VPS вместо локального сервера, его надо настроить на часовую зону физического местоположения сервера. Настройки находятся в **/etc/php5/apache2/php.ini**, и строка должна быть раскомментирована удалением начальной ;, после чего добавьте часовой пояс. Обычно правильный часовой пояс указан в **/etc/timezone**. Изменив часовой пояс или любые другие настройки PHP, перезапустите Apache:

```
sudo apachectl restart
```

Настройка

Перезапустите программу установки и исправьте все дальнейшие ошибки, прежде чем продолжать. Следующая страница задает настройки. Если вы перешли на собственный webmail по соображениям конфиденциальности, вы, вероятно, предпочтете отключить программу проверки правописания, потому что она по умолчанию отправляет текст на проверку не куда-нибудь, а в Google! Вам нужно сообщить ей о вашей базе данных и, если только вы не используете *SQLite*, настроить таблицы самостоятельно, запустив клиент *mysql* от имени root и скомандовав

```
CREATE DATABASE roundcubemail;
GRANT ALL PRIVILEGES ON roundcubemail.* TO username@localhost IDENTIFIED BY 'password'; FLUSH PRIVILEGES;
```

Измените имя пользователя и пароль на свои, но localhost оставьте, поскольку сервер базы данных находится в той же системе, что и web-сервер. Альтернатива — создать базу данных и пользователя с помощью *phpMyAdmin*. Затем добавьте один или более серверов IMAP для соединения. Если вы добавите более одного, у вас будет меню, где можно выбирать нужный

«Webmail лучше: вам не придется менять электронный адрес.»



➤ Программа установки создает два файла, которые вы должны скачать и скопировать в свою директорию config в *Roundcube*.

Администрирование

Пожалуй, самое приятное в публичных сервисах то, что они «просто работают». Не нужно ничего настраивать — достаточно создать учетную запись; незачем переживать насчет создания резервных копий, требований к хранению, сетевой безопасности, DDoS-атак и кучи иных проблем. Если вы действительно решили заняться собой всерьез, все эти проблемы придется учитывать, точно так же, как для традиционных компьютерных технологий и систем

хранения данных. Конечно, риски можно превратить в преимущества.

Использование частного облака для хранения всех ваших файлов имеет единую точку отказа, но также означает, что у вас есть только одна позиция для резервного копирования. Синхронизация одного облачного сервера с чем-то одним вонне проще, чем выполнение той же задачи для пары десятков настольных систем. Функции присвоения файлам версий и отмены удаления

файлов (undelete) обеспечивают «дуракоупорность» при случайном удалении не той папки с компьютера пользователя. В приватной сети часть таких проблем у вашего сервиса просто отсутствует, а если у вас уже есть публичный web-сервер, то эти проблемы уже решались; а значит, все не так страшно, как может показаться на первый взгляд. Однако не советуем вовлекать критически важные данные или сервисы в ваши первые опыты.

при загрузке страницы; если вы оставите его пустым, вам придется вводить имя сервера при каждом обращении к *Roundcube*. Щелкнув по кнопке Create Config, вы увидите страницу со ссылками на скачивание двух файлов, **main.inc.php** и **db.inc.php**; скопируйте их в директорию **config** своего *Roundcube*. Затем щелкните Continue, и начнется проверка вашей конфигурации. Исправьте все, что отмечено флажком (либо вернувшись на предыдущую страницу, либо редактируя файлы напрямую), и перезагрузите тестовую страницу. Здесь же вы должны еще протестировать настройки SMTP и IMAP. Заметьте предупреждение большими красными буквами в конце тестовой страницы: вы должны удалить или отключить программу установки после настройки *Roundcube*, в противном случае ваши настройки и пароли будут видны всем.

Чтение почты

То, какую страницу вы увидите, введя URL вашей webmail в свой браузер, зависит от настроек. Если вы настроили один почтовый сервер, она сразу будет запрашивать имя пользователя и пароль; в ином случае вы увидите либо меню, где надо будет выбрать один из нескольких серверов, или текстовое поле, если вы не добавляли серверов.

Войдя впервые, вы некоторое время прождете, поскольку ваш почтовый ящик будет сканироваться. Скорость загрузки почтового ящика по большей части зависит от скорости соединения между web-сервером и вашим почтовым сервером (если их поместить на одном компьютере, разница в лучшую сторону будет огромной), а также от количества хлама, то есть, извините, почты, которая в нем завалылась.

Меню настроек предлагает опции для изменения как внешнего вида, так и метода работы *Roundcube*, а сайт <http://roundcube.net> содержит wiki с богатой информацией по настройке *Roundcube* в соответствии с вашими нуждами. Здесь же вы найдете немало плагинов, которые можно установить, расширив функциональность и улучшив внешний вид *Roundcube*. Плагины идут в виде архивов, обычно TAR или ZIP; архив нужно распаковать в директорию плагинов *Roundcube* — там вы увидите несколько плагинов, установленных по умолчанию. По умолчанию они отключены; можно включить их, добавив их имена, которые появляются в директории плагинов, в массив `$rcmail_config['plugins']` в **config/main.inc.php**. Например, чтобы добавить имеющийся плагин autologon и плагин от стороннего производителя *SpamAssassin*, настройка будет такой:

```
$rcmail_config['plugins'] = array('autologon', 'sauserprefs');
```

The screenshot shows the 'Database setup' section with fields for db_dsnw, Database type (MySQL), Database server (localhost), Database name (roundcubemail), Database user name (roundcube), and Database password (****). Below it is the 'IMAP Settings' section with fields for default_host (mail.Overthere.com), default_port (143), and usernames_domain.

➤ Нужен по крайней мере один сервер IMAP, с которого вы будете читать почту, и сервер SMTP для отправки почты.

The screenshot shows the 'Roundcube Webmail Installer' main screen. It displays the progress of the installation: 1. Check environment, 2. Create config, 3. Test config. The 'Checking PHP version' section shows 'Version: OK (PHP 5.3.3-7+squeeze15 detected)'. The 'Checking PHP extensions' section lists required and optional extensions, with 'Intl' marked as 'NOT AVAILABLE'. The 'Checking available databases' section is partially visible at the bottom.

➤ *Roundcube* устанавливается собственной сетевой программой установки. Первый шаг — проверка выполнения всех системных требований.

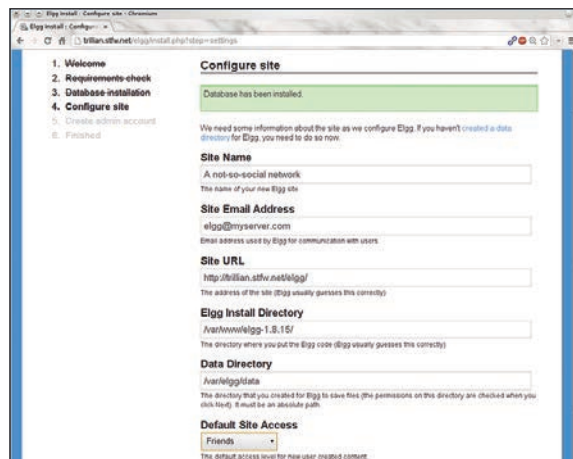
Социальные сети

Объединитесь в защиту приватности — на своей же социальной медиа-платформе.

Настройка собственного webmail — это хорошо и мило, но электронная почта становится пережитком XX века; по крайней мере, в этом нас пытаются убедить Facebook и иже с ним! Из-за социальных сетей возникли проблемы с конфиденциальностью; компании могут использовать Facebook и его подобия, чтобы «привлечь» клиентов, которым это «нравится», но все же не станут применять их для внутренних переговоров. На самом деле, в большинстве организаций весьма жесткая политика по поводу обсуждения любых дел компании в социальных сетях. И все же эти сервисы предоставляют людям бесценную возможность общения — как в обществе в целом, так и внутри организации. Итак, нам нужна социальная платформа, которой мы будем управлять сами — желательно с открытым кодом, чтобы смягчить проблемы по защите конфиденциальности. Есть несколько возможностей — например, *Buddy Press* (<http://buddypress.org>), которая изначально появилась на свет как разновидность *WordPress* и выросла в целую социальную медиа-платформу; но мы здесь рассмотрим *Elgg* (<http://elgg.org>). Популярность и удобство *Elgg* исчерпывающе характеризует список ее пользователей — хорошо известных организаций и образовательных учреждений, выбравших эту платформу.

Знакомая процедура

Процесс установки такой же, как у многих других web-приложений. Скачайте ZIP-файл, распакуйте его в директорию документов вашего web-сервера (желательно сделать символическую ссылку на более удобное имя, чем **elgg-1.8.15**), и настройте базу данных. *Elgg* использует *MySQL*, и стандартные инструкции, которые вы найдете повсеместно, вполне применимы для создания



➤ Настроив *Elgg*, можете начать с задания ей имени и указания адреса электронной почты и места для хранения данных.

базы данных и пользователя. Вам также нужно создать директорию данных для *Elgg* — она не должна находиться в **Document-Root** вашего сервера, то есть в области, откуда он обслуживает страницы, поскольку вам не нужен прямой доступ туда из браузера. Директория должна находиться во владении и с правом записи у пользователя, который запускает web-сервер, обычно это **www-data** или **apache**. Это делается из командной строки, хотя большую часть этой задачи можно выполнить с помощью *phpMyAdmin* и файлового менеджера, работающего с SSH; затем загрузите <http://yourserver/elgg> в свой браузер. Он проведет вас через

Добавление базы данных MySQL

MySQL — это сервер базы данных; база данных нужна всем упомянутым здесь пакетам, чтобы хранить настройки и данные. И хотя *MySQL* — один из самых популярных серверов, альтернатив ему хватает. *SQLite*, как понятно по его названию, является более легкой опцией. Он хранит всю базу данных в одном файле и отлично работает в системах с одним, не особенно требовательным пользователем. Однако когда речь идет о пользователях и загрузке, он не слишком хорошо масштабируется. *MySQL* применяет классическую модель сервер/клиент — один сервер заботится обо всех потребностях базы данных нескольких программ, и если вы используете более одного пакета, о которых мы здесь рассказываем, вероятно, оптимальным выбором для вас будет *MySQL*; правда, вам придется настраивать базы данных и пользователей на сервере для каждого пакета. Когда вы установите *MySQL*, он спросит у вас пароль — это пароль для пользователя **root**, он не должен использоваться отдельными пакетами. Вы можете создать новую базу данных и пользователя для нее с помощью клиента командной строки *MySQL*: команда

```
mysql -u root -p
```

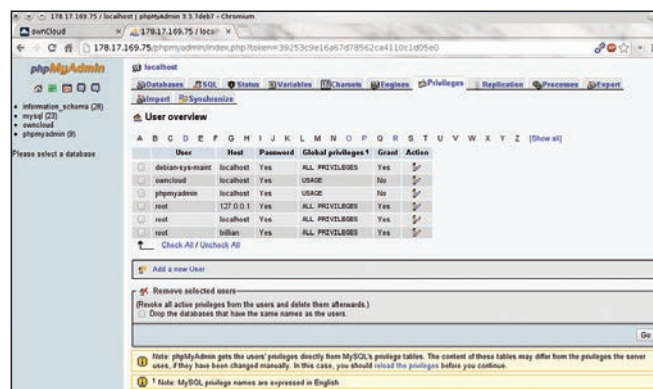
предложит вам ввести пароль, затем вы создадите базу данных и пользователя с помощью

```
CREATE DATABASE dbname;
```

```
GRANT ALL PRIVILEGES ON dbname.* TO username@localhost IDENTIFIED BY 'password';
```

```
FLUSH PRIVILEGES;
```

Первая строка создает базу данных, вторая — пользователя с паролем, разрешая ему полный доступ к этой базе данных. Команда **flush** просто велит *MySQL* немедленно применить изменения. По окончании работы используйте



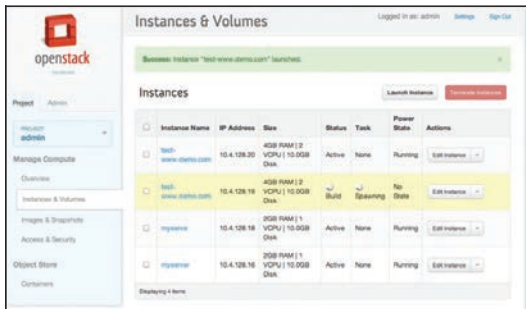
➤ *phpMyAdmin* сводит работу с базами данных *MySQL* до простых действий: навести, щелкнуть и немного попечатать.

↳, чтобы выйти из клиента. Вы можете приказать своей программе использовать эту базу данных и пользователя, и она сама может создать структуру базы данных.

Альтернативой всей этой морочи с командной строкой будет использование *phpMyAdmin*, сетевого инструмента администрирования для *MySQL*. Есть некая ирония в создании пользователя для *phpMyAdmin* командной строкой, однако все, что требуется сделать, подробно изложено в web-интерфейсе.

OpenStack

Хотя здесь мы сконцентрировались на *ownCloud*, есть и альтернативы; одной из них является *OpenStack*. Однако нацелен *OpenStack* в большей степени на широкомасштабное использование. *OwnCloud* устроит и предприятия, да есть у него и коммерческие варианты; но он отлично подходит также и для личного использования, и для небольшой организации. А вот *OpenStack* в большей степени ориентирован на тех, кто желает предложить другим коммерческое использование облачных сервисов. Это отнюдь не означает, что надо сбросить *OpenStack* со счетов, и если ваши требования — нечто большее, чем личный облачный сервис, вам стоит заглянуть на www.openstack.org: возможно, он вас вполне удовлетворит.



» *OpenStack* — еще одна реализация облака с открытым кодом, явно нацеленная на предприятия.

стадии настройки, проверит наличие всех необходимых модулей, настроит учетные данные базы данных и проверит доступ и настройку размещения директории данных. Если на какой-то стадии он выявит проблему, то попытается подсказать вам ее решение.

Решив проблему, нажмите Обновить, чтобы заново провести проверку. Последним шагом будет создание учетной записи администратора. Войдя в систему как администратор, вы сможете радикально изменить внешний вид сайта, а также включить различные плагины и другие опции.

Освойте Elgg

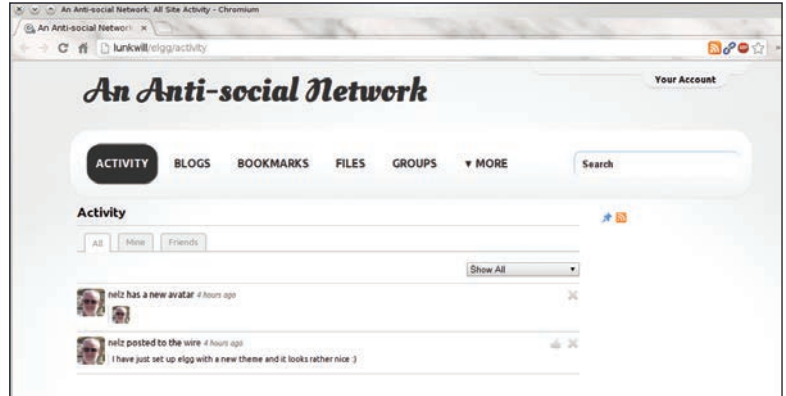
Elgg неплохо задокументирована, однако эти документы столь же неплохо спрятаны, и на них нет никаких ссылок с домашней страницы сайта *Elgg*. Перед установкой зайдите на <http://docs.elgg.org>. После установки *Elgg* вы увидите ссылки на Manual и FAQ внизу страницы Administration.

Теперь вам должен быть знаком метод управления плагинами. Доступ к установленным плагинам осуществляется по ссылке на странице администрирования. Новые плагины скачиваются в виде ZIP-файлов, для распаковки в директорию `mod`, после чего они появятся в вышеупомянутом списке. Темы тоже являются плагинами, и устанавливаются и настраиваются аналогично. Ссылки на репозитории тем и плагинов размещаются на начальной странице руководства администратора.

Помните, что, поскольку темы считаются плагинами, можно ненароком выбрать более одной за раз, что порой приводит к эксцентричным результатам. Полезная функция страницы плагинов — перемещение их вниз и вверх по списку приоритетности, для изменения порядка их загрузки.

Что такое LAMP?

Web-сервисы работают на системе, известной как LAMP, где первые три буквы означают Linux, *Apache* и *MySQL*, а P может быть PHP, Perl или Python, в зависимости от ваших предпочтений. *Apache* является стандартом де-факто web-сервера для Linux,



хотя есть и альтернативы — например, *Lighttpd* и *Cherokee*, которые часто легче и работают с этими сервисами. Однако *Apache* наиболее популярен, и у всех программ есть настройки для работы с ним, тогда как использование другого web-сервера может предполагать определенные труды, подробности которых можно найти на сайтах соответствующих проектов. Установить *Apache* проще всего из вашего менеджера пакетов (вы ведь уже позаботились об L в LAMP, верно?), Python и Perl часто устанавливаются по умолчанию, что оставляет нам PHP и *MySQL*, тоже устанавливаемые через менеджер пакетов.

Это далеко не все

Здесь мы рассмотрели три основные программы — во всем, что касается хранения файлов, электронной почты и социальных сетей, хотя все они предоставляют и дополнительные функции, такие как фотоальбомы, календарь, управление контактами и закладками. У каждой из них, как мы упоминали, есть альтернативы, но есть множество и других облачных программ. Например, *Gallery* (<http://gallery.menalto.com>) — специализированный фотоальбом, который предоставляет гораздо большую функциональность, чем дополнения к программам, рассмотренным здесь.

«Elgg характеризует список ее пользователей.»

Это типичное web-приложение на PHP, поэтому установка и настройка во многом такая же, как и у других программ, и у него всеохватывающий web-интерфейс администрирования для создания публичных и частных альбомов, а также множество плагинов, чтобы еще больше расширить возможности *Gallery*.

Есть также сервис обмена быстрыми сообщениями с открытым кодом, использующий *Jabber* (который, помимо прочих, применяется в *Google Talk*). *Asterisk*, о котором мы рассказывали раньше, поможет в настройке вашего собственного VoIP-шлюза, что особенно удобно для организаций, где более одного офиса.

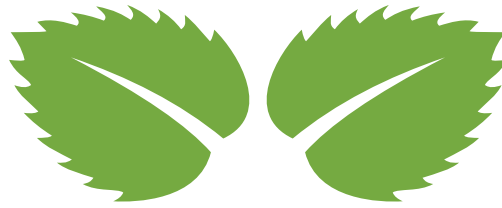
Коммерческие программы опережают свои альтернативы с открытым кодом только в одной важной области, и это — офисные программы. Хотя WebDAV-доступность *ownCloud* позволяет осуществлять редактирование где угодно и кем угодно при наличии правильного логина, все же она зависит от клиентского офисного ПО. В разработке находится проект под названием OX documents, у которого в настоящее время есть демо-версия на https://www.ox.io/ox_text. Интересно, что он поддерживает и формат ODF, и файлы форматов Word, и являет собой первый шаг на пути к созданию офисного онлайн-пакета с открытым кодом. Первый релиз должен вот-вот выйти, и когда в нем появятся электронные таблицы, мы сможем делать в собственном облаке практически все то же самое, что позволяет проприетарное облако, не теряя своей конфиденциальности. LXF

» Есть даже темы, которые заставляют *Elgg* выглядеть как сайт некоей популярной социальной сети.

ВЫЙТИ







Э значит Эбен

Грэм Моррисон посетил Raspberry Pi Foundation, чтобы порасспросить его основателя об интеллектуальной собственности, обновлении оборудования и многом другом.



Мало кто сегодня столь же востребован, как Эбен Аптон (Eben Upton). Будучи известным как создатель Raspberry Pi, он принимал самое активное участие в судьбе этого

мини-компьютера, неотрывно наблюдая за его проектированием, воплощением и невероятным взлетом к всенародному признанию. Параллельно фонд перенес производство в Уэльс, выпустил Model A и модуль камеры, и продолжает продвигать Raspberry Pi в новые сферы.

Воспользовавшись передышкой между шквалом предложений от потенциальных партнеров и международных СМИ, он уделил нам пару часов для интервью. Значительная его часть, где обсуждается все — от истоков проекта до управленческих сложностей — опубликована на www.linuxradar.com.

Linux Format: Много ли трудностей пришлось с успехом Raspberry Pi?

Эбен Аптон: Думаю, многое из того, что связано с логистикой и масштабами, взяли на себя RS и Element 14 (два изготовителя Raspberry Pi).

LXF: А вы заранее на это рассчитывали?

ЭА: Да, так и планировалось изначально, ведь еще до старта проекта мы осознавали, что его успех может оказаться нам не под силу, у нас были деньги на создание 10 000 образцов — четверть миллиона долларов... Наверное, этот момент наступил, когда мы выпустили первый образ карты

SD для устройства и получили 50 000 закачек, или глючную альфа-версию первой операционной системы, в жизнеспособности которой сомневались сами разработчики; тогда-то и стало понятно, что дело плохо [смеется], и пора подумать о нашей модели всерьез.

LXF: Это было в конце 2011?

ЭА: Да, и если бы мы тогда остановились на той первой модели, то весь прошлый год пришлось бы заниматься только заказами с того первого дня. И хотя нам кажется, что так и было, на самом деле это заняло месяца 3–4. Но нас бы это точно никуда не привело: в лучшем случае, сделали бы 100 000 за первый год, но никак не миллион. Так что, глядя с позиции Raspberry Pi, весь размах мы «наворожили» благодаря сотрудничеству с организациями, ему соответствующими. То есть с теми, кто занимается логистикой в масштабах Amazon. По данным RS, они делают по 44 000 отправок в день. И то, что это значительно для их оборота — не повод сомневаться в его общем объеме.

LXF: Выход этого устройства стал настоящим событием!

ЭА: Да, точно. Только мы жили этим около девяти месяцев, а они [RS и Element] — всего пару.

LXF: Лиз (пиар-директор Raspberry Pi и жена Эбена) тоже обеспечила предвкушение публики.

ЭА: Именно, так что для них, я думаю, это стало сюрпризом. И очень даже приятным сюрпризом —

для обеих компаний проект сделался хорошим подспорьем в бизнесе, очевидно, что они сегодня преуспевают. И выводят нас на нужный масштаб. Мы стали иначе смотреть на цели, которые прежде считали едва достижимыми.

LXF: А что это значило лично для вас?

ЭА: Много, конечно же... Не то чтобы это было менее увлекательно, но гораздо более серьезно. Потому как есть люди, которые этим зарабатывают. Изначально мы не думали создавать рабочие места, то есть нанимать кого-то помимо разработчиков — конструкторов, распространителей. Но сегодня уже много, порядка сотни, тех, для кого Raspberry Pi — основной источник дохода. И это заставляет задуматься, когда на вас вдруг найдет: «О, а давайте поменяем вот этот компонент на PCB». Ведь если за неделю мы уже сделали 100 таких, а потом решили зарубить этот продукт, то эти 100 единиц придется уничтожить. А при нынешних объемах, естественно, недельный выход составит 30, 40, а то и 50 тысяч единиц, что может нанести урон всему бизнесу, и потом люди начнут терять работу. Так что увлекательно, но и ответственно.

LXF: Любопытно, что вы задумали Raspberry Pi одновременно и как фонд, и как бизнес, сегодня такое нечасто увидишь.

ЭА: Я считаю, что если вам действительно важна окупаемость, то так и должно быть. Есть мнение, что благотворительностью не обойдешься — вот »



мы хотели быть благотворительной организацией, и являемся некоммерческой. Просто вырученные деньги вновь идут в дело. Я не получаю зарплату. Мне повезло — я могу себе такое позволить.

LXF: И ваша позиция не изменится?

ЭА: Когда-нибудь — возможно. Ведь работаю я, на самом деле, даже больше, чем полный день, — больше, чем для многих означает «на полный день» в этой организации. Но я до сих пор числюсь сотрудником Broadcom. И это огромная щедрость с их стороны. По-моему, всегда подозревалось, что это маркетинговый ход, а я считаю это подкормкой идеи. Нам просто повезло; но пришлось таки убеждать Broadcom, что это дело стоящее.

LXF: Так, как было, когда Линус Торвальдс работал с Transmeta?

ЭА: Да, примерно те же отношения. Вероятно, менее затягивающие, насколько я могу осознавать его огромный вклад в Transmeta. Но я уделяю около 90 % своего времени Raspberry Pi. Так что, да.

О РАЗГЛАШЕНИИ СХЕМЫ УСТРОЙСТВА PI: «Деньги за нашу интеллектуальную собственность уходят на Pi.»

Буду ли я когда-нибудь работать за зарплату? Возможно, если в какой-то момент проект перестанет быть рентабельным для Broadcom, и я сочту, что настало время менять приоритеты. В любом случае, даже если так и будет, никаких дивидендов здесь люди не получают, и все извлеченные доходы, что ни остается после выдачи зарплаты, вновь вкладываются в проект. И это радует, ведь это значит, что окупаются по крайней мере та его маленькая часть, что принадлежит нам, и у нас есть стратегический потенциал, чтобы экспериментировать... например, финансировать разработку камеры. Создаем Pi, на общественных началах, занимался Пит Ломас [Pete Lomas]

из Norcott (Technologies Limited), являясь, по факту, одним из попечителей фонда. Сама же плата камеры была коммерческим продуктом, за ее разработку мы платили.

LXF: Влияет ли это на возможность открытия схемы устройства?

ЭА: Схема открыта, кроме PCB, и это интересный вопрос. Откроем ли мы когда-нибудь схему PCB? Такое намерение всегда было, да оно и остается. Проблема в том, что чипов-то не купить. По сути, это еще одна проблема, требующая решения.

LXF: Вы поэтому не раскрываете PCB?

ЭА: Если мы ее раскроем, то сделать Raspberry Pi самому вам могут помешать лишь нехватка чипов и... нарекания... но, думаю, мы мало кого обидели, не выдав все целиком. А вот с нареканиями мы сейчас сталкиваемся. Теперь я представляю, до чего бы они дошли, если бы решение о публикации принималось сейчас. И при той огромной поддержке, оказанной нам Broadcom, я считаю, что они этого не заслуживают. Так что пока я рад, что все эти кирпичи летят только в мой огород.

Во мне недостаточно альтруизма, чтобы делиться своей интеллектуальной собственностью [IP], с учетом того, что деньги за нее удерживают Pi на плаву... Важно понимать, что Pi будет существовать и через 5 лет. А это солидный коммерческий аргумент — чтобы поставить производство Pi на поток или построить на его основе программу обучения.

LXF: И вы не хотите, чтобы к этому примешивался кто-то со стороны, создав свой собственный аналог...

ЭА: Именно. И по-моему, если бы дело было только в том, чтобы раскрыть все устройство и позволить производителям его клонировать, подорвав финансовую основу нашего проекта... было бы еще ничего. Но речь ведь не о том, чтобы срубить по 10 баксов с каждо-

го Pi, верно? Мы и так не много получаем с каждого, к тому же мы снизили цену для всех, и доходы, которые пошли бы кому-то в карман, просто обратились бы в дивиденды, в доход компаний. В конечном итоге, это привело бы к краху самого фонда, его возможностей в об-

разовании, в инвестировании открытых проектов без расчета на явную выгоду потребителей. Разве что в идейном плане почувствовать, какие мы молодцы: вот дернули за веревочку, и дверка открылась, — ну и все. А потом на Broadcom обрушились бы все эти нарекания за то, что они не предоставляют свои чипы.

LXF: Планируется ли усовершенствовать Pi? Не хотели ли вы сделать его модульным?

ЭА: Пока что нам некуда втиснуть дополнительную оперативку... Увеличить объем памяти мы не можем: SoC [system on a chip] не позволяет. По той же причине нельзя и добавить



О RASPBERRY PI 2

«Будет фатально для нас продавать в 2016-м все тот же Pi.»

потратил деньги на дивный новый мир — если мы собираемся туда попасть. Люди этого хотят, так что мы туда отправимся, и довольно скоро. Мы уже кое-то сделали совместно с компанией Collaboga, здесь, в Кембридже. Пек (Пекка Пааланен) [Pekka Paalanen], который работает на них и многое делает для *Wayland*, занимается разработкой *Weston*, эталонного компоновщика *Wayland*, лежащего в основе нашего аппаратного движка визуализации, благодаря которому мы получаем необычайно мощный HVS, инструмент масштабирования видео [video scaler], представляющий собой большой движок аппаратных

спрайтов. Он выдает их сразу помногу, а вы помещаете внутрь каждого окно и [хлопает в ладоши] шлепаете их друг на друга на экране. Пропускной способности памяти хватит на значительное количество таковых. Как только она иссякнет, есть резервный вариант компоновки вне экрана. И это здорово. Это позволяет целиком перетаскивать окна на 60 Гц. В условиях, где каждое окно является аппаратным спрайтом, это можно делать без сдвига пикселей.

Сам видеoinструмент размером примерно с ARM, даже больше, поскольку VHS требует большого буфера памяти для операций со спрайтами. При наличии средства для визуализации иерархий и менеджера компоновки окон нам необходимо ПО, способное их связать. Это хороший пример повышения качества использования исключительно за счет ПО, без впаивания лишних чипов. К тому же это открывает нам путь в чудесный *Wayland*.

LXF: Пожалуй, это первое применение *Wayland* в столь широком масштабе.

ЭА: Думаю, это первая самостоятельная не-GUI платформа для *Wayland*. Обычно он работает поверх GL, но GL для компоновки совершенно не годится из-за примитивных фильтров масштабирования. Даже при уменьшении размера приходится создавать кучу текстур, а у нас это делается так: допустим, мне нужно привести что-то к 3:1, и проблема в том, что это нечто среднее между 2:1 и 4:1. В GL вы получите либо текстуру на 2:1 с некоторым зашумлением, либо текстуру на 4:1 с зернистостью. Или можете попытаться «линейно интерполировать» их — станет получше, но не намного. В нашем случае (при тех же 3:1) каждый пиксель строится осреднением сетки 3×3 пикселей исходного изображения, для каждого фрейма; для вас бесплатно.

Узнать больше о работе с *Wayland* и прочитать еще 4000 слов нашего эксклюзивного интервью с Эбен Аптоном вы можете на www.tuxradar.com. **LXF**

производительности. На самом деле, по аппаратной части у нас стратегического плана нет.

LXF: То есть вы даже подсознательно не представляете себе *Raspberry Pi 2*?

ЭА: Думаю, будет очень печально, и даже фатально для нас продавать, скажем, в 2016-м все тот же *Raspberry Pi*. Видимо, придется что-то с этим делать, но пока не представляю, что... Реальная проблема в том, что я представляю себе платы, которые мог бы создать, за любую цену от \$25 до \$85... И те новые, которые я мог бы создавать за каждые добавочные \$10. Все они существуют пока только в моей голове. Но как найти такую, что будет на самом деле привлекательна, которая добьется такого же признания, как Pi... А козырь Pi — в соотношении цены и качества. Этого очень сложно добиться. О чем-то в промежутке \$25–35 уже, естественно, речи не идет, другую такую плату мне уже не сделать. На любом имеющемся чипе невозможно представить другую плату ни за \$25, ни даже за \$35, которая могла бы сравниться с Pi, а тем более превзойти ее.

Но после \$45 настает скачок. При \$55 я уже представляю, как получить кое-что получше, но, что интересно, этот сегмент пока пустует.

LXF: Как же вам удается удерживать преимущество?

ЭА: Мы усиленно работаем над программным обеспечением.

LXF: Значит, сейчас все усилия направлены на это?

ЭА: При частоте в 700 МГц [Pi] является необычайно мощным ускорителем медиа. Чип — это на 97% ускоритель.

LXF: И как это используется в установленном по умолчанию *Raspbian*?

ЭА: На данный момент, имеется видеовывод, USB-контроллер; то, что было до появления ARM. ARM сделал чип на 3% больше. Основную нагрузку несет именно ARM, некоторые элементы инфраструктуры, контроллер SDRAM и несколько крохотных периферийных устройств. Значительная часть системы находится по большей части в спящем режиме. И все равно, 700 МГц для ARM — это очень мощный процессор, хотя по общепринятым стандартам тяжеловатый. Но раз уж здесь мы повязаны, то делаем больший упор на ПО, чтобы выжать из него максимум. Поэтому много времени было уделено оптимизации системных компонентов, будем надеяться, для общего блага, системных компонентов Linux — того же *pixmap*. Улучшению таких вещей, как *memset* и *memset*.

Любопытная была дискуссия по поводу ускорения графики. У нас ускорителя Иксов нет, как нет и драйвера ускорителя. Но у нас есть множество компонентов чипа, множество подсистем, способных влиять на ускоритель X-сервера. И на самом деле, этого достаточно. С ПО для X все хорошо. Я даже сам удивляюсь — насколько. Просто ARM в нужный момент подстраивает нужный пиксель, правда, теперь, с налаженным *pixmap*, они как из автомата выстреливают. Настолько быстро, насколько это возможно при довольно низкой производительности ARM. А дискуссия была о том, нужно ли нам аппаратное ускорение графики. И вроде как мы сошлись на том, что не нужно. Что нужно, так это стремиться в дивный новый мир.

LXF: Потому что так проще?

ЭА: Думаю, да. Мне кажется, так больше вариантов... и при наличии выбора, я бы лучше



Внутри EFF

Майк Сондерс выясняет, как Electronic Frontier Foundation (EFF) защищает нас от плутовства мегакорпораций и слежки правительств.

Наша свобода делиться информацией, открыто выражать свои мысли, предлагать новые идеи и защищать свою частную жизнь — под угрозой. Правительства постоянно ищут все новые способы следить за нашими делами, а крупные предприятия постоянно пытаются подсадить нас на их продукцию. Частичка за частичкой, наша свобода разрушается.

Популярные СМИ часто проводят параллель между нынешней ситуацией и романом Джорджа Оруэлла «1984», который многие считают гиперболой. Однако задумайтесь на минутку: правительства прослушивают наши телефоны, читают нашу электронную почту и следят за нашими действиями онлайн. Социальные сайты и поисковые движки создают гигантские базы данных наших просмотров и наших покупок. И все это делается на фоне постоянной так называемой угрозы со стороны тер-

рористов, политиков и владельцев СМИ и супербогатой элиты, которая продолжает богатеть.

Смотри в оба

Как видите, стоит быть начеку — возможно, даже немного удариться в паранойю. Океания [вымышленное государство в романе-антиутопии Дж. Оруэлла «1984», — прим. пер.] строится не за один день, и полицейское государство тоже не появится за ночь. Но времена меняются, и нашей свободе угрожают со всех сторон; борьба с этим потребует настойчивости и решимости. И одной из групп, которые твердо стоят на нашей стороне, является EFF.

«Океания строится не за день, и полицейское государство — не за ночь.»

Цель Electronic Frontier Foundation — «защита ваших прав в цифровом мире», и его деятельность включает полный спектр действий по защите свободы: помощь в судебных разбирательствах; выпуск подробных докладов с рассказами об имеющихся угрозах; кампании по повышению информированности; разработка технологий защиты нашей онлайн-активности от любопытствующих глаз. Это некоммерческая организация, существующая на пожертвования, которая располагается в Сан-Франциско и имеет впечатляющий список сотрудников (загляните на www.eff.org/about/staff), в числе которых юристы, аналитики и активисты — и в основном придерживающаяся свободного ПО и культуры GNU/Linux.

Читайте нашу статью, чтобы узнать, как возник EFF, что он сделал и как ведет подготовку к предстоящим битвам.



EFF действует

Бросить вызов Секретной службе США — рискованное предприятие...

Но EFF сделала именно это, вскоре после своего образования в июле 1990 года. Секретная служба оказывала давление на мелкого издателя книг по видеоиграм в процессе поиска похищенного документа, который мог попасть не в те руки. В конечном итоге, улики для выдвижения обвинения не оказались, но издатель оказался на грани банкротства, поскольку его компьютеры конфисковали, он сорвал все сроки, и ему пришлось уволить сотрудников. А что хуже всего — Секретная служба уничтожила большую часть важных данных этого издателя.

Группа озабоченных проблемой защиты свободы умников — представлявших собой подобные же примеры нарушения прав в растущем онлайн-мире и цифровом мире — узнали об этом. И тогда Митч Кейпор [Mitch Karpog] (создатель Lotus, а позднее председатель Mozilla Foundation), Джон Перри Барлоу [John Perry Barlow] (бывший автор текстов *Grateful Dead*) и Джон Гилмор [John Gilmore] (бывший когда-то сотрудником Sun Microsystems) создали EFF, получив дополнительное финансирование от соучредителя Apple Стива Возняка [Steve Wozniak]. Их первой важной задачей стало поддержать издателя в судебном иске против Секретной службы.

В конечном итоге издатель выиграл два из трех исков — \$50 000 на возмещение ущерба и \$250 000 на судебные издержки. Но намного важнее было привлечение внимания общественности к вопросу о том, нужна ли электронной коммуникации такая же юридическая защита, что и иным формам коммуникации (например, телефонным звонкам). Суд постановил, что у Секретной службы не было права просматривать электронную почту издателя без специального ордера на просмотр электронной почты, создав прецедент, существующий по сей день.

EFF продолжил оказывать содействие во многих аналогичных случаях, что мы и увидим на следующих страницах, и масштаб его деятельности расширился, когда использование Интернета стремительно выросло и в 1990-х стали появляться другие проблемы. Сегодня Гилмор и Барлоу остаются в Совете директоров EFF, и к ним присоединились и другие известные лица, например, Брайан Белендорф [Brian Behlendorf], бывший ранее разработчиком *Apache*, интервью с которым вы можете прочитать в **LXF80**, («Ангел *Apache*», стр. 34), и Джон Бакман [John Bickman] (основатель музыкального сервиса *Magnatune*). Вместе они курируют команду из 44 человек с десятками тысяч помощников по всему миру.



► Митч Кейпор, создатель электронных таблиц Lotus 1-2-3, а также руководитель и соучредитель EFF.

«Секретная служба уничтожила часть ВАЖНЫХ ДАННЫХ.»

Руководство «Как победить в споре» — вырежи и сохрани

Множество благонамеренных типов полагают, что мы чересчур расшумелись и столь серьезные меры по защите не нужны. Вот их обычные аргументы и ответы на них.

1 «Я не боюсь Правительства X или Компании Y!»
Рады за вас, однако вам ведь неизвестно, какими будут X или Y в будущем. Возможно, сейчас у вас достойнейшее правительство, но что будет с данными, собранными о вас, когда к власти придет кто-то другой? Вам все так же будут нужны эти законы о надзоре и технологии слежения за коммуникациями, когда к власти придет партия, которая вам активно не нравится?
То же касается и компаний — пускай Компания Y исключительно порядочна в вопросах конфиденциальности и ваших данных, но как она поведет себя через 10 лет, когда в ней полностью сменится состав руководства?

2 «Если вы делаете что-то онлайн тайне от всех, может, просто не стоит этого делать?»
У нас у всех есть право на частную жизнь, и есть масса вполне законных и важных причин не желать, чтобы ваши действия в Сети стали достоянием гласности. Скажем, вы консультируетесь по проблеме со здоровьем и вовсе не хотите, чтобы к вам потом поперли рекламные объявления данной тематики. Вы можете захотеть купить какой-нибудь тур или подарок любимому человеку так, чтобы не засветиться раньше времени и не испортить сюрприз. Или вам нужно провести некие политические или исторические изыскания так, чтобы в штаб-квартире правительства не завывала сирена.

3 «Если Компания X начнет меня доставать, я просто перейду в Компанию Y.»
Если бы все в жизни было настолько просто... Это же не продуктовый магазин поменять — тут все сложнее. Что произойдет со всеми вашими книгами и музыкой, которые вы приобрели у X и которые работают только на устройствах от этой компании? Легко ли вам удастся вытащить все свои данные и импортировать их в Y? Что будет с данными, которые в X накопили о вас за все эти годы — сможете вы заставить компанию удалить их? Вряд ли все так однозначно.
4 «О нет! Вы лучше о детях подумайте!»
Спокойно. Пойду заварю вам чайку.

Решения, решения

Как EFF выбирает свои баталии.

Как EFF решает, кому оказать юридическую помощь? Ведь его время и финансы не бесконечны, а на свете есть бесчисленное множество областей, в которых он мог бы что-то сделать. Работники EFF проверяют адрес info@eff.org, на который может написать любой, кому нужна юридическая помощь; какое решение примет команда — использовать ли ресурсы фонда — зависит от нескольких ключевых моментов.

Во-первых, если у дела есть шанс оказать влияние на законодательство — создать прецедент для будущих судебных разбирательств — EFF, скорее всего, примет в нем участие.

Громкие судебные процессы, способные повлиять на будущее, помогают также и популяризации работы фонда. Во-вторых, EFF может помочь нескольким людям или группам в таких случаях, как составление

группового иска, то есть это тоже важный фактор. И, наконец, EFF особенно охотно помогает тем, кому просто не по карману большие судебные издержки.

Иногда EFF критикуют за то, что они не высказываются открыто и громко по какому-либо вопросу, особенно если имело место солидное событие, касаемое цифровых прав, и добрые пол-Интернета схлестнулись в словесных баталиях. Однако EFF должен вести очень благоразумную и взвешенную политику: «Худшее, что мы можем сделать — это высказаться вслух до того, как выработается юридическая стратегия». Так что когда в следующий раз вы будете наблюдать за нечистой игрой, направленной на ограничение свободы, со стороны правительства или компании, и при этом не услышите ни звука от EFF, значит, фонд, вероятно, ведет свое расследование.



Рейтинг технологий

Поддерживаемые EFF проекты ПО для защиты вашей свободы.

1 HTTPS Everywhere • www.eff.org/https-everywhere
Это расширение для браузеров *Firefox* и *Chrome* ставится везде, где возможно, использовать соединение HTTPS (безопасное), даже когда сайты по умолчанию настроены на HTTP. Это соединение шифрует данные, передаваемые с вашего компьютера на удаленный сайт. Не все сайты поддерживают HTTPS, или поддерживают не для всех страниц, так что для выбранных сайтов вы можете отключить это расширение, щелкнув по значку.

2 Panopticklick • <http://panopticklick.eff.org>
Блестящая (и, если честно, пугающая) web-страница демонстрирует вам, сколько информации ваш браузер оставляет на сайтах. Даже если у вас отключены cookies, сайт все равно может идентифицировать и отследить вас на основе комбинации данных из вашего браузера, таких, как пользовательский агент, версии плагина, размер экрана и установленные шрифты. Это «отпечатки пальцев браузера»,

а поскольку вряд ли у кого-то другого в Интернете они идентичны, вас очень легко накрыть.

3 Switzerland • www.eff.org/pages/switzerland-network-testing-tool
Известно, что некоторые интернет-провайдеры вводят в соединение своих клиентов, ограничивая применение технологий вроде BitTorrent. Switzerland позволяет проверить целостность соединения между двумя машинами в Интернете и предупредит, если пакеты вашего IP подверглись изменению.

4 TOSBack • www.tosback.org
Ну кто читает соглашения по сервисам на сайтах? Они всегда такие длинные, и напигованы всякими юридическими крючками, придуманными, чтобы сами юристы собой гордились. Этот сайт, хотя на данный момент он в стадии глобальной переделки, показывает, когда самые основные онлайн-сервисы меняют условия своих пользовательских

соглашений, выделяя части, которые могут быть важными для вас. И если вы, допустим, услышали, что Facebook изменил свое соглашение, но вам вовсе не хочется продирааться сквозь весь текст в поисках поправок, этот сайт вам поможет.



➤ Panopticklick — это эксперимент по замеру онлайн-отслеживания и защите от него.

Главные победы EFF

За всю свою историю EFF ввязывался не в одну крупную драку и выходил из них победителем. Вот самые яркие.

Против Apple

В середине 2000-х для сайтов со сплетнями об Apple было золотое времячко. Пресловутая скрытность Apple в сочетании с распространённостью разнообразных i-гаджетов означала, что любая утечка информации ценилась этими сайтами на вес золота. Интригующие факты и фото изредка просачивались из штаб-квартиры Apple, и замять скандалы бывало непросто.

Apple это надоело, и они подали в суд на некоторых онлайн-журналистов, пытаясь установить, кто стоял за утечками информации. EFF принял в этом деле участие, представляя интересы журналистов и доказывая их право сохранять свои источники в тайне. И они выиграли: решение было вынесено в пользу EFF, и давало защиту журналистам от претензий Apple. Большая победа для независимых СМИ.

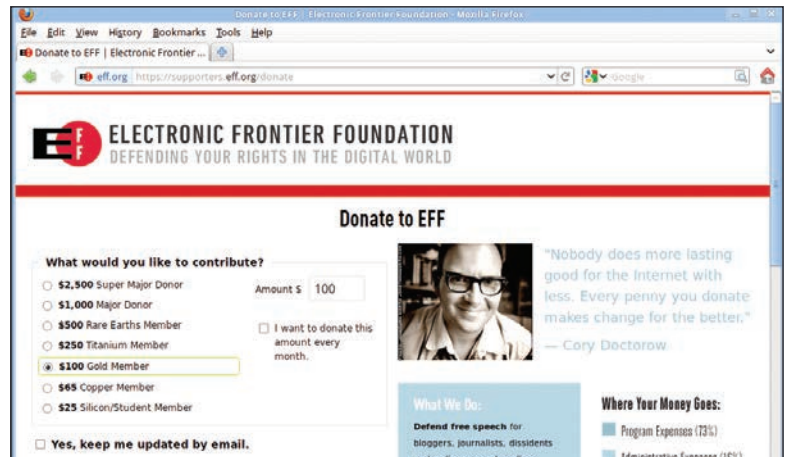
Против Sony

В 2005 в Sony начали выпускать музыкальные CD, на которых заодно имелись программы Windows для реализации защиты от копирования и Digital Rights Management. Эти программы устанавливали скрытые файлы, отправляли информацию об использовании CD через Интернет и делали компьютер потенциально уязвимым с точки зрения безопасности. Эта идея была катастрофична с самого начала, хотя Sony умудрилась отгрузить 22 миллиона CD с этими весьма спорными программами.

И буря не замедлила разразиться, хотя Sony сперва отбивала всю критику, заявляя, что ее программы безопасности не нарушали. Но через некоторое время Sony пришлось изъять CD, хотя и только после того, как EFF подала в суд, чтобы заставить компанию предпринять шаги по устранению ущерба, предлагая программы для удаления этого ПО и сообщая пользователям, где получить техподдержку по вопросам удаления этих программ.

Против правительственных агентств

Одним из исключительно мерзких предложений в начале 2000-х было предложение о «флаге широкоэмитальной трансляции [broadcast flag]» от Федерального агентства США по связи (US Federal Communications Commission, FCC — ФАС). По сути,



это был код статуса, который вещательные компании могли отсылать на домашние устройства записи — он запрещал устройствам записывать определенные программы. Тем самым потребители лишались контроля, хотя и приобрели оборудование и ПО для записи видео, и этот контроль полностью передавался медиа-империям. В суде EFF оспаривала возможность ФАС управлять происходящим в наших телевизорах и компьютерах по получении ими сигнала вещания, и суд согласился с этим. Это был важный шаг, утверждающий наше право использовать и контролировать приобретенную нами продукцию.

Одновременно уже долгие годы Департамент юстиции США (US Justice Department) использовал весьма спорные технологии слежки — например, определение местоположения мобильного телефона подозреваемого без специального ордера. EFF вступила в борьбу, доказывая судьям, что они не должны выдавать подобный далеко идущий и крайне опасный ордер без предварительного предъявления «достаточного основания» — то есть обоснованной вероятности того, что преступление совершается. Эта битва продолжается, и она крайне важна, чтобы защитить нас от всевидящего Большого Брата. **LXF**

► Работу EFF обеспечивают пожертвования — загляните на <https://supporters.eff.org/donate>, и вы тоже сможете получить членскую карту.



Все (лучшее/худшее) еще впереди...

Каковы самые главные угрозы?

Ребекка Йешке [Rebecca Jeschke] — директор EFF по связям со СМИ, а по совместительству — аналитик фонда по цифровым правам. Мы встретились с ней, желая узнать, что ждет нас в будущем...

LXF: Как вы считаете, о чем нам стоит переживать в ближайшие несколько лет?

РЙ: Одной из проблем, требующих внимания, являются растущие попытки правительства США собирать большие объемы данных. Мы уже знаем, что АНБ (Агентство национальной безопасности — National Security Agency, NSA) собирает с помощью своих новых шпионских приспособлений внушительные по объему данные о деятельности

в Интернет (<http://tinyurl.com/c8fk466>, <http://tinyurl.com/buttire>), и уже звучат требования сделать всю деятельность онлайн более доступной для прослушивания властями. (<http://tinyurl.com/cv5yn52>).

И правительству США вовсе не нужно самому заниматься сбором всей этой информации — достаточно взглянуть на юридическую поддержку CISPA, закона, который предоставит компаниям большие возможности по сбору «угрожающей» информации (например, из частных бесед пользователей) и передачи этих данных правительству без ордера — сюда входит и пересылка данных в АНБ. Нет нужды говорить, что хотя это и американская практика, она все равно повлияет на людей по всему миру.

LXF: А как обстоят дела с компаниями?

РЙ: Третьи стороны — например, Facebook и Twitter — уже много знают о нас, и вскоре будут знать еще больше. И очень важно защитить эту информацию от бездумного сбора и употребления. Мы не говорим, что к ней не должно быть доступа по закону, но правительство должно в большинстве случаев получить ордер, и за сбором информации должен наблюдать судья. Те, кто не хочет превращать свою жизнь в открытую книгу, должны быть уверены, что их законодатели знают об их принципах. У нас в США очень сильна оппозиция по CISPA, и мы полагаем, что именно это — одна из причин, по которой закон застрял в Сенате.



Следующее поколение KDE

KDE охватывает новые технологии. Стюарт Джарвис выясняет, почему KDE 5 не бывать.

В 2008 году KDE представил ребрендинг, который означал, что KDE 4 более нет. Вместо этого KDE стало именем сообщества, выпускающего ПО в трех областях: рабочие среды (или рабочие столы), такие, как Plasma Desktop и Plasma Netbook, платформа разработки (не без изысканности названная KDE Platform), и, конечно, сами приложения KDE.

Ребрендинг стал результатом дискуссий, которые велись в KDE в предшествующем году: как сделать приложения и библиотеки более привлекательными для тех, кто не использует рабочий стол, вне зависимости от того, предпочитают ли они Gnome, Windows или Mac OS. Тогда это было чисто маркетинговым изысканием и не привело ни к каким реальным подвижкам в создании или выпуске программ KDE, а в результате и особого воздействия не возымело. Вы до сих пор можете встретить тех, кто говорит о «KDE 4.x», даже в самом сообществе KDE.

Процесс, который был запущен этой сменой бренда, теперь начинает осуществляться. Разница между продуктами KDE будет теперь заключаться не только в названии; KDE усвоил уроки, полученные после выхода KDE 4.0. Когда он выпустил вместе совершенно новые бренды библиотек и рабочего стола, получилось два

продукта, которые не были ни нормально протестированы, ни стабильны для повседневного использования. Эти ошибки больше не повторятся. KDE 5 не будет, и новое поколение программ от KDE вряд ли появится разом.

Разделение библиотек

В следующем поколении технологий KDE есть нечто большее, нежели простое разделение рабочих областей и приложений. Библиотеки, которые будут именоваться KDE Frameworks 5 (KF5), сами по себе подверглись разделению, чтобы сделать

«Разделение делает KF5 привлекательнее для использования вне KDE.»

программы KDE более модульными. Цель — сделать приложения привлекательнее для изолированного использования и облегчить независимым разработчикам использование частей KF5.

Есть — или ощущается, что есть — давняя проблема использования приложений KDE вне рабочего пространства KDE, например, в средах Gnome или Unity. Открытость и повторное употребление кода, которое делает программы KDE совместимыми и мощными, означает, что

даже простейшие приложения могут иметь десятки, а то и сотни мегабайт прямых зависимостей. У пользователей рабочего пространства KDE, скорее всего, они уже установлены, однако пользователям чистой системы Gnome понадобится скачать солидный объем. Эти проблемы еще сильнее обостряются на таких платформах, как Windows — команде портирования приложений KDE в Windows даже пришлось создать нечто вроде менеджера пакетов на роль программы установки. Это работает, однако сама концепция крайне непривычна для пользователей Windows,

и даже небольшие приложения требуют зависимостей, на скачивание которых может уйти длительное время.

Чтобы преодолеть эту проблему, предпринимались — и до сих пор предпринимаются — попытки сделать KF5 как можно более модульной, чтобы приложения требовали минимума зависимостей. Инженер и исследователь открытого кода Кевин Оттенс [Kevin Ottens] немало потрудился над решением этой проблемы; по его словам, наметился неплохой прогресс, и низкоуровневые части неплохо поддаются разделению. В целом, Кевин видит немало преимуществ для пользователей приложений KDE, особенно на других платформах. «Приложения будут проще портировать на разные платформы

и форм-факторы, — говорит Кевин. — Мы будем экономить ресурсы. Каждое приложение сможет обращаться только к тому, что оно использует. Разделение сделает KF5 привлекательнее для использования вне сообщества KDE, а также повысит ценность Qt, поскольку у нас есть решения для функций, которых нет в Qt». Коротко говоря, пользователям Qt станет проще, чем раньше, включать некоторые функции KDE без весьма внушительного и сложного списка зависимостей для своих приложений.

Вверх по течению

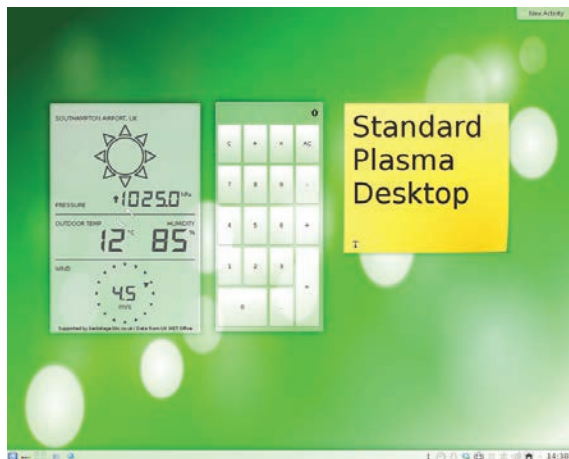
Другая ключевая цель процесса KF5 — передача некоторых функций «к истоку [upstream]», в Qt. Разработчик KDE Давид Фор [David Faure] отмечает, что «для KDE это не потеря». Будет ли разработчик поддерживать код в Qt или в библиотеках KDE, результат будет доступен разработчикам KDE. Есть тут и преимущества. Компоненты, прежде применимые только в приложениях KDE, могут начать использоваться намного шире, и потому их тестирование тоже станет более масштабным, с привлечением сторонних разработчиков.

«Некоторые вставки кода в Qt, например, **QSaveFile**, — говорит разработчик KDE Стивен Келли [Stephen Kelly], — привели к появлению ошибок, которые были также обнаружены в классе **KSaveFile**». Их обнаружение означает, что они будут исправлены в следующем релизе KDE 4.x, поэтому преимущества как для настоящего, так и для будущего очевидны. «Я также наблюдаю, как все больше народу начинает принимать участие в работе над KDE Frameworks и содействовать работе над Qt», добавляет Стивен.

Давид отмечает, что благодаря переносу функций в Qt больше не будут предлагаться альтернативы классов для одного и того же действия (как это было в прошлом). Это также должно облегчить портирование на платформы кроме Linux: ведь у разработчиков «будет гарантия того, что они полностью протестированы на всех платформах, а не только в Linux».

Давид объясняет, что перенос функций в Qt часто включает чистое переписывание кода, потому что API не всегда соответствует стандартам Qt и по причинам, связанным с лицензированием: работа с Qt требует лицензионного соглашения с тем, кто ее выполняет, не предоставляемого изначально авторами. «Это также отличный шанс улучшить реализацию, — добавляет Давид, выделяя ситуацию с обработкой типа MIME (типы MIME используются для описания типа контента файлов, чтобы они открывались в нужном приложении и обрабатывались соответствующим образом). — Реализацией было использование кэша текстового файла, но технология, лежащая в основе Freedesktop (sharedmime-info), предоставила нам более эффективный двоичный кэш». Давид уверен, что новый код обработки типа MIME принесет «серьезный рост производительности».

Перенос функций в Qt не означает отказа от специфичных для KDE адаптаций и стилей. «Механизм тем в Qt — слой QPA (Qt



» Рабочий стол Plasma переключается с **QGraphicsWidgets** (слева) на **QML** (справа). Их почти не отличить, поскольку задачей было не менять работу пользователя.

Platform Abstraction) — позволяет запускать специфичные для KDE функции через стандартные API Qt, — говорит Давид. — Нам не придется убеждать разработчиков Qt в целесообразности каждой нашей потребности».

В пример он приводит загрузку значков, когда Qt использует встроенное в память кэширование на каждый процесс. «С точки зрения KDE, мы смотрим на множество разных процессов, поэтому имеется общий для процессов кэш. Это типичное расхождение точек зрения разработчиков Qt (рассматривать одно приложение

за один раз) и рабочих пространств KDE (рассматривать множество работающих приложений). Плагин QPA в рабочем пространстве KDE загрузит плагин тем KDE, а тот предоставит движок для про-

граммы загрузки значков, который использует общий кэш в файле, употребляемом всеми процессами». Перенос ряда функций в Qt сопровождался трудами по применению непосредственно Qt вместо библиотек KDE. Там, где функции целиком перейдут в Qt, KF5 вставит новые Qt-эквиваленты прежних компонентов KDE.

Больше прока от Qt

Естественно, существуют ограничения на то, что желательно перенести в Qt: «Цель заменить всю систему локализации KDE Qt-эквивалентом пока не ставится, — говорит Стивен. — Его недостаточно для нужд приложений KDE. Но часть KDE Frameworks не особо нуждается в системе локализации, и эта часть портируется, чтобы по возможности применять систему Qt. Это важное уменьшение зависимостей при сохранении функциональности». Другие элементы по-прежнему зависят от необходимых функций, которые добавляются в Qt. По словам Стивена, использование **QDateTime** вместо **KDateTime** в первую очередь требует, чтобы классы даты и времени Qt имели функции обработки часовых поясов. «Цель в том, чтобы не потерять важных функций в KDE Frameworks или приложениях KDE, непосредственно

«Новые Qt-эквиваленты старых компонентов KDE применят в KF5.»

Что мы видим на экране: X, Wayland и Mir

X (или, более официально, *X Window System*) долго служила технологией по умолчанию для отображения на экране графики в системах Linux и Unix. Она мощная и отличается сетевой прозрачностью, что было важно в первые дни рабочих станций Unix, когда пользователь мог сидеть за консолью и запускать приложения на удаленном компьютере.

За последнее время в X было отмечено значительное количество недостатков, и началась работа над протоколом *Wayland* (и сопутствующими библиотеками) ей на замену. Предполагается, что *Wayland* будет проще и передаст большую часть контроля менеджерам окон, например, *KWin* в KDE. Этот прямой контроль позволит *KWin* убедиться,

что содержание окна выводится на экран в должное время и в должном месте.

В марте этого года в Canonical объявили, что Ubuntu будет вместо *Wayland* использовать свой сервер отображения, *Mir*. Однако KDE и Gnome дали понять, что в следующем поколении сервера отображения сохранится *Wayland*.

используя компоненты *Qt*. В общем, что хорошо для *Qt*, хорошо и для KDE. KDE может ожидать от *Qt* той надежности, к которой мы привыкли, зная, что она постоянно хорошо контролируется».

Портирование в Qt 5

Qt 5 была разработана с упором на совместимость по исходнику с *Qt 4*. В результате портирование большинства приложений KDE в *Qt 5* должно быть довольно простым. Однако есть в рабочем пространстве KDE область, требующая немалых трудов по портированию: это *KWin*, менеджер окон. «*Qt* обещает 99 % совместимости кода, — говорит разработчик *KWin* Мартин Гресслин [Martin Graesslin]. — Нужно поднастроить всего лишь 1 % зависящего от платформы кода. О да, этот 1 % — *KWin*».

Некоторые важные отличия *Qt 5*, влияющие на *KWin* — это переход на API QPA, ликвидировавший многие специфические для платформы вызовы, которые использовал *KWin*. Еще одно важное изменение — переход *Qt 5* с XLib на XCB, что потребовало переписать обработку событий в *KWin* (XCB — это привязка языка C к протоколу X).

Переход на *Qt 5*, помимо дополнительной работы, несет в себе и ряд преимуществ. QML все больше используется в *KWin*, что позволяет создавать более привлекательные интерфейсы для таких эффектов, как текущие окна или раскладка рабочего стола (которые показывают, соответственно, все окна или все рабочие столы на экране в уменьшенном масштабе).

«*KWin 4.11* больше не использует удаленные функции *Qt*, и можно скомпилировать *KWin* для *Qt 5* и KDE Frameworks 5», говорит Мартин. Тем не менее, обработка событий пока не портирована, поэтому порт *KWin* в *Qt 5* пока не готов. Цель в том, чтобы *KWin 4.11* мог работать с использованием *Qt 5*, однако *Qt 4* по-прежнему остается целевой (и рекомендуемой) платформой.

Подготовка к Wayland

Еще одна солидная область работы в *KWin* для следующего поколения рабочих пространств KDE — его подготовка к использованию *Wayland* в качестве замены X. Это зависит от порта в *Qt 5*, потому что Мартин хочет использовать в *Qt 5* модуль *QtWayland*. «Это создаст оболочку в стиле *Qt* вокруг библиотеки *Wayland*, — объясняет Мартин. — Для нас это весьма интересно, ведь нам все равно пришлось бы написать такую оболочку, поскольку *Wayland* — это библиотека C. Главная сложность здесь в том, что *KWin* разработан и написан исключительно для X11, что требует сильного внутреннего рефакторинга — задача, работа над которой велась в течение последних нескольких лет». Меняются также функции, предоставляемые *KWin*. «В мире *Wayland* компоновщик (*KWin*) берет на себя множество задач, которые должен выполнять X-сервер: например, вывод результата в окна».

Поддержка *Wayland* в *KWin* должна появиться к концу года. По словам Мартина, это зависит от темпов портирования в *Qt 5* и состояния модуля *QtWayland*. Сначала поддержка *Wayland* будет тестироваться в Plasma Active (рабочем пространстве KDE для планшетов), а затем, если там она заработает, другие рабочие пространства тоже перейдут на *Wayland*.

Быстрый переход на *Wayland* не является приоритетом. «Мы не хотим сломать рабочий стол, поэтому в этом вопросе останемся консерваторами, — уверяет Мартин. — На меня никто не давит, и я не собираюсь спешить только потому, что Gnome уже осуществил переход или Unity работает на *Mir*».

Но даже в такой ситуации уже налицо преимущества всех этих усилий по портированию. «Благодаря портированию XCB *KWin* стал быстрее», говорит Мартин, добавляя, что «пересмотр приводит либо к росту производительности, либо к дополнительным функциям — например, лучшей работе с краями экрана в *4.11*».

Обновление Plasma

Конечно же, мотивом для работы над KF5 является поддержка нового поколения приложений и рабочих пространств KDE. Начиная с KDE 4.0, рабочие пространства использовали библиотеки Plasma в качестве своей базы, и поэтому немалая работа ведется также над следующим поколением Plasma.

Одно из самых крупных изменений, ожидающих Plasma — отказ от использования основанными на X виджетами *Qt Quick* (часть *Qt*, отвечающая за создание интерфейсов пользователя с помощью декларативного языка QML). Рабочие пространства Plasma полностью перейдут на Plasma Quick. «Это будет *Qt Quick* плюс ряд компонентов интеграции для тем, взаимодействие с компоновщиком, интернационализация, доступ и разделение данных, конфигурации и оборудования», говорит разработчик Plasma Себастьян Кюглер [Sebastian Kuegler]. По словам Себастьяна, портирование компонентов рабочего пространства в QML уже проводится, и многие виджеты уже портированы, в том числе System Tray, Pager, Notifications, Device Notifier, Battery, Lock/

➤ Новая область включения [containment] рабочего стола Plasma упрощает позиционирование виджетов, показывая тенью конечное расположение виджета после перетаскивания.



Qt, основа программ KDE

Qt — кроссплатформенная среда разработки, ведущая начало с 1991 года. Она была выбрана в качестве основы для KDE, когда в 1996 году этот проект запускался. Как правило, за новыми версиями *Qt* следуют новые версии программ KDE. *Qt 5* внес в эту среду серьезные изменения, в частности, в компоненты интерфейса и рендеринг графики. Теперь для создания интерфейса пользователя предпочитают *Qt Quick* (используя декларативный язык QML). Это позволяет рабочим пространствам Plasma 2 в KDE

иметь собственную раскладку и поведение, определенные в QML, облегчая настройку рабочих пространств и независимость рабочего пространства от машины, на котором оно работает. Не надо компилировать область включения, что упрощает ее разделение и использование на разных компьютерах.

Qt также претерпела недавние изменения в руководстве. Изначально разработанная независимой компанией Trolltech, *Qt* была приобретена фирмой Nokia и использовалась в их проектах Symbian

и MeeGo. С переходом Nokia на Windows Phone в 2011, применение *Qt* в компании угасло, и *Qt* продали в Digia. Примерно тогда же было основано Открытое управление *Qt* [Open Governance], и в соответствии с ним разные компании и организации могут вносить свой вклад прямо в *Qt*, а кодовая база принадлежит независимому проекту Qt Project. Это серьезно облегчает организациям типа KDE, заинтересованным в экосистеме *Qt*, возможность определять разработку и развитие *Qt*.



► **Портирование приложений в Qt 5 должно быть относительно простым. Изначальный порт Konqueror показал лишь мелкие дефекты рендеринга. (Изображение от Стивена Келли/KDAB)**

Logout, Weather, Wallpaper и поддержка Containment. «Чтобы обеспечить возможность портирования и поддержки, предпочтение отдается чистым QML-виджетам, — говорит Себастьян. — Для сложных случаев, когда невозможно обойтись QML, мы используем комбинированный C++ и QML-апплет».

Когда ранее в этом году вышел KDE 4.10, экспериментальная область включения рабочего стола, портированная в QML, стала доступна пользователям. «Большая часть новой области включения в точности такая же, как в текущей настройке по умолчанию, — поясняет Себастьян. — Так и задумано. Мы не хотим радикальных изменений в рабочем пространстве и организации работы пользователей, а постепенно совершенствуемся в процессе итераций».

Но две вещи все-таки изменились в экспериментальной области включения QML: позиционирование/размер и визуальная чистота. Виджеты и их фреймы объединены, чтобы не загромождать рабочую область, и упростились позиционирование. «При перемещении мы показываем ореол там, где окажется апплет после перемещения, — объясняет Себастьян. — Это делает перемещение виджетов более предсказуемым и разрешает пользователю совершить его за один прием. И в конечном итоге апплеты оказываются с точностью до пикселя выровнены друг относительно друга».

Новое поколение приложений

Основная выгода от перехода на QML для области включения в том, что теперь намного проще разрабатывать настраиваемые области включения (раскладки рабочего стола) для разных областей применения. «Теперь очень легко создать собственную область включения, изменить что-то еще и разрешить распределенный доступ к ним, — говорит Себастьян. — Поскольку виджеты и области включения независимы от архитектуры, это, естественно, работает на любых устройствах». В Plasma 2 вместо использования разных кодов C++ для рабочих пространств настольного ПК, нетбука и планшета, все они определяются как области включения QML, работающие на единой базе C++.

Plasma 2 должна обеспечить большее удобство работы пользователя, несмотря на некоторые первоначальные различия во внешнем виде. Себастьян утверждает, что многое в графике Linux изменилось к лучшему. Главное, что преимущества от применения новых технологий вроде Qt Quick (которая умеет использовать для рендеринга OpenGL и графическое оборудование) и Wayland позволяют Plasma выводить «60 идеальных фреймов на экран каждую секунду, обеспечивая работу пользовательского интерфейса как по маслу».

Хорошая новость для разработчиков приложений заключается в том, что исходник KF5 будет в основном совместим с библиотеками KDE 4. А значит, при портировании приложения во Frameworks 5 изменения будут минимальны. Основным преимуществом станет то, что благодаря разделению библиотек каждое приложение получит возможность иметь меньше прямых зависимостей. Это должно сделать отдельные приложения привлекательнее для использования в Gnome и Unity, в особенности на таких платформах, как Windows и Mac OS, а также даст возможность создавать компактные системы, где требуется минимум приложений.

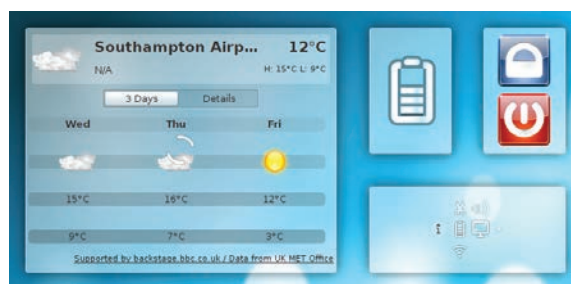
Стивен уже попытался портировать некоторые приложения KDE на Qt 5. «В порядке эксперимента я портировал некоторые приложения KDE — Dolphin, Konqueror и части PIM KDE — на Qt 5. Мне нужно будет еще поработать над этим, но уже видно, как легко мы можем дойти до момента запуска кода и поиска оставшихся ошибок в порте». Есть также возможность переместить интерфейсы в QML. «Портирование существующего пользовательского интерфейса в QML — это опциональный шаг в порте Qt 5, который можно осуществить или до, или после портирования в Qt 5», говорит Стивен.

Будущее уже наступило

Судя по всему, будущее поколение программ KDE будет готово по этапам: сначала достигнет зрелости KDE Frameworks 5, затем Plasma 2, и, наконец, будут вноситься изменения в приложения, чтобы извлечь все преимущества новых технологий.

Изменения в KDE Frameworks 5 уже появляются, так же как и поэтапные изменения в KWin и Plasma, которые оказывают положительное влияние на пользователей, в основном через расширение применения QML, которое делает интерфейсы приятнее с виду и более устойчивыми. К концу этого года можно будет использовать KWin на Wayland, но, по словам Себастьяна, «базовый релиз Plasma 2 появится примерно через год». И все же на данный момент создается впечатление, что изменения будут гораздо менее разрушительными, чем при переходе на KDE 4.0, и модернизация глубоких компонентов в программах Linux и KDE сделает комбинацию более устойчивой в будущем и способствует переносимости программ KDE на новые и будущие платформы.

Будущее для KDE — это нечто большее, нежели просто среда рабочего стола. Модульная структура KDE Frameworks 5 нацелена на то, чтобы предоставлять компоненты, пригодные для отдельного использования (что будет крайне привлекательно для любого разработчика Qt) и при этом обладающие способностями к расширению и увеличивающие ценность Qt как среды приложений. Plasma 2 принесет на традиционные компьютеры и планшеты более плавно функционирующие рабочие области и облегчит создание настраиваемых рабочих областей для новых устройств. Приложения KDE будут легче портироваться, иметь меньше зависимостей, и их будет проще развернуть где угодно. LXF



► Многие из существующих виджетов Plasma портированы в QML, что значительно облегчает их адаптацию и распределенный доступ.

Феномен Ubuntu



Алексей Федорчук рассказывает о перевороте в мире FOSS, который был вызван появлением Ubuntu, и размышляет над его причинами.

В одной из заметок исторического цикла (LXF155) речь шла о возникновении и первых шагах дистрибутива Ubuntu и его сородичей. И рассказ тот обрывался на моменте, когда Ubuntu обрел всенародную известность и популярность в широких народных массах. В чем далеко превзошел всех предшественников. Но история никогда

не кончается — и впереди было самое интересное. То, что позволяет сегодня оценить феномен Ubuntu во всей его полноте. Однако для этого нам опять придется вернуться к истокам — хотя настоящая заметка относится не к историческому циклу, а скорее к околополитическому. То есть написана и с гневом, и с пристрастием.

Немного ретроспективы

Итак, обратим вспять время и посмотрим, как выглядел мир Linux накануне появления Ubuntu на арене истории, то есть к осени 2004 года. Надо сказать, что картина сложилась вполне благодная:

» Red Hat целиком переключился на коммерческие продукты, а в свободное от этого время экспериментирует в своей песочнице, именуемой Fedora, при участии сложившегося вокруг сообщества волонтеров;

» SUSE, недавно купленная компанией Novell, пытается расширить свое присутствие на американском рынке, и потому вынуждена идти на... ногу в ногу с Red Hat; что, в частности, проявляется в учреждении собственной песочницы — openSUSE;

» Mandriva балансирует между банкротством и получением немислимых правительственных контрактов (или все-таки субсидий?), хотя не прочь запустить свои щупальца и в сопредельные страны с давними своими, еще со времен Mandrake, приверженцами, такими как Россия и Бразилия;

» разработчики Debian ведут — правда, уже шепотом — разговоры о мировом господстве на всех платформах (в том числе чужих, вроде разных BSD'ей, а то и вовсе несуществующих, типа Hurd), чем обещают осчастливить все прогрессивное человечество;

» пользователи Slackware, под мудрым руководством Великого Патрика, продолжают изучать матчасть, благодаря чему время от времени поставляют кадры «продвинутых» пользователей существующих дистрибутивов, а то и разработчиков дистрибутивов новых (например, Zenwalk);

» пользователи Gentoo, поднабравшись опыта, задумываются, а куда бы им переползли, но пока стойко продолжают перекомпилировать систему по будням, отдыхая за настройкой опций компиляции по праздникам;

» Герард Беекманс [Gerard Beekmans] продолжает регулярно выпускать новые издания своей LFS — вокруг него по-прежнему группируется могучая кучка приверженцев, развивающих его дело виришь, в виде Beyond LFS;

» создатели юзерофильных дистрибутивов с псевдокоммерческим уклоном, таких, как Vector Linux, MEPIS, Xandos, Lindows/Linspire, потеряли надежду массово развести лохов своими Linux'ами, которые «виндее все виндей», но пока еще рассчитывают удержать лохов уже окученных;

» разнообразные нишевые дистрибутивы и дистрибутивы «для себя»... да кто же уследит, что происходит в их мирках; разве что некоторым из них, как Arch Linux, удастся попасть в запасной состав вышеперечисленной сборной — правда, как потом выяснится, ценой отказа от своего стиля игры.

В общем, «жизнь идет, прорабы строят, парикмахеры стригут»: крупные (по меркам FOSS-бизнеса) воротилы ворочаются, мелкие барыги — барыжничают, гики — гикствуют, строители светлого будущего на одном отдельно взятом десктопе — продолжают его строить, мало внимания обращая на несовершенство окружающего мира.

На пресловутых «простых» пользователей все дружно забыли. Разговоры о Linux'е с человеческим лицом, да еще на каждом десктопе, ведутся чисто по привычке. Выражение «виндокапец» превращается в мантру, смысл которой забылся. Упоминание виндового апокалипсиса становится ритуальной фразой, подобной напоминаниям о неизбежности победы коммунизма в мировом масштабе. В светлом будущем, разумеется, в отдаленной перспективе.

Часть недавних приверженцев лозунга «Каждой домохозяйке — по теплomu Linux'у!» понемногу перекалфицируется в оправдомы... пардон, в сисадмины. И начинает зарабатывать

Замечания личного плана

У автора этих строк есть несколько сайтов обще-UNIX'овой направленности и несколько сайтов дистрибутив-ориентированных. Так вот, статистика их посещаемости показывает, что как на первых, так и на вторых среди всех заходов с UNIX-машин однозначно лидирует Ubuntu. В отношении сайтов первой группы это легко объ-

яснимо: как сказано в статье, убунтуцев количественно больше, чем всех остальных линуксоидов, вместе взятых. Но их интерес к иным дистрибутивам, ни с какой стороны с Ubuntu не пересекающимся, являет собой свидетельство здорового любопытства. Сильно утраченного пользователями стариков от дистростроения...

на жизнь знаниями и умениями, приобретенными за время пламенного энтузиазма. Другие, возвращаясь к Windows, становятся офисными менеджерами и клерками. И вспоминают период пламенного энтузиазма как увлечение юности. Те же, кто видел свою роль в амплуа не пропагандиста, но популяризатора, просто отходят в сторону. И если и продолжают свою деятельность — то, скорее, по привычке.

Иными словами, идет нормальная цивилизованная жизнь, внешне кипящая, но на самом деле устаканившаяся. И ничто, казалось, не в силах нарушить сложившееся благолепие. Если бы не два фактора.

Первый — те самые «простые» пользователи, о которых Linux забыл. Но которые не забыли о Linux'е. Ибо были не так уж про-

сты, и нашли в нем среду для эффективного решения своих профессиональных задач. Или для применения к своим очень серьезным хобби — серьезным до той грани, где хобби смыкается с профессией.

То есть те пользователи, которых я не так давно выделил в группу применителей <http://alv.me/?p=1549>.

Второй же фактор — появление главного героя моего сегодняшнего рассказа, дистрибутива Ubuntu, играющего роль «простого султана Брунея». Благодаря которому, прямо или косвенно, наши применители могли перестать задаваться вопросом: «Что ты сделал для Linux'а?». А получили не только право, но и возможность спросить: «Что Linux сделал для тебя?» (см. LXF95).

Те же и герой

«Под звон мечей и зловещее пение стрел в огне пожаров вышел на арену мировой истории русский народ», написал В. В. Мавродин в книге о возникновении Древнерусского государства. Наш же герой появляется на арене мира Linux скорее под клацанье клавиш и подвывание кулеров. И представляется как самый совершенный и окончательный пользовательский Linux-десктоп, с помощью которого любая кухарка сможет управлять персональным компьютером — возможно, даже не под наблюдением комиссара... то есть, пардон, сисадмина. Тем самым смешивая расклад, описанный в прошлом разделе.

Правда, выяснилось это не сразу: «действующие» пользователи Linux встретили появление Ubuntu... да никак они его не встретили. Ибо помнили еще и 1999 год, обещавший приход Linux'а на каждый пользовательский десктоп. И первую волну юзерофильных дистрибутивов, каждый из которых представлялся как «Linux с человеческим лицом» (можно подумать, что до этого у Linux'а было не лицо, а... еще одна спина). И то, как эти человеколюбивые дистрибутивы меняли имена, исчезали или влачили жалкое существование, не нужные никому, подчас даже своим создателям.

Так что те самые действующие пользователи, интересующиеся новыми дистрибутивами, поначалу пророчили и Ubuntu ту же

»

Откуда есть пошла

Все началось с того, что была выпущена Windows 95. К которой, как и к Ubuntu, поначалу никто не относился серьезно: она воспринималась как платформа для запуска игрушек. Даже для всамделишной офисной работы резонные люди консервативного склада отдавали предпочтение старой, не очень доброй, но досконально известной Windows 3.1/WfW 3.11, тогда как прогрессисты склонялись к OS/2. Что же до серверов на Windows 95 — такое могло привидеться

разве что в кошмарном сне с большого перепоя.

Нет, у Microsoft была в загашнике и самая настоящая ОС — Windows NT, от которой по прямой линии происходят все варианты всех современных Windows. Но как серверная платформа она и близко не была тогда конкуренткой не только UNIX'ам, но даже OS/2. А на рабочих станциях применение NT тормозилось интерфейсом, унаследованным от Windows 3.1 — последний в счи-

танные месяцы после выхода 95-й стал казаться старомодным.

Однако, быстро оккупировав домашние компьютеры, Windows 95 постепенно утвердилась на рабочих местах различных контор. А затем... затем Microsoft в очередной раз всех напарила, выпустив Windows NT 4 с интерфейсом в стиле modern, то есть а la Windows 95. И именно с нее началось распространение NT-серверов и рабочих станций.

судьбу. Должен сознаться, среди них, наряду со многими, был и автор этих строк. Однако то был один из тех нередких случаев, когда провидцы и ясновидцы, даже будучи очевидцами, оказались не правы (повторяю, это и ко мне относится). Ибо не учли, что организатор всего этого безнадёжного предприятия, Марк Шаттлворт [Mark Shuttleworth], окажется способным на весьма неординарные меры для продвижения своего произведения. Не пренебрегая, однако, и мерами вполне тривиальными.

Самой неординарной мерой была организация заказов на комплекты дистрибутива совершенно безвозмездно, то есть даром — с бесплатной же почтовой доставкой по всему миру, в том числе по России. И это, как ни странно, действительно работало — даже в самых удаленных городах и весях нашей необъятной родины.

Правда, в связи с этой акцией скептики поначалу иронизировали: нынче каждый линуксоид имеет возможность бесплатно заказать себе подставку под пивную кружку. А ведь зря иронизировали: в результате о Linux'e узнала масса людей, прежде о его существовании и не подозревавших. И немало представителей этой массы дистрибутив хотя бы опробовали. А кое-кто из опробовавших так к нему и прикипел.

Это была одна из причин почти мгновенного роста популярности Ubuntu. Вторая же, как я говорил — вполне тривиальна: интенсивная «работа над ошибками», и не только своими. Ubuntu изначально позиционировался как очередной Linux с человеческим лицом, с одной стороны, и концентратор самого свежего софта — с другой. В плане первого пункта были учтены все ошибки прежних попыток «очеловечивания» Linux'a. И в итоге разработчикам удалось если не найти оптимум между «настройкой с папальником и осциллографом» и «молчаливыми визардами для полных идиотов», то вплотную к нему приблизиться.

Направление работ по второму пункту очевидно: использование самых свежих версий софта всегда потенциально чревато ошибками в оном, и ошибки эти следовало исправлять. Или не допускать — путем сознательного ограничения «степени свежести»: ведь программы, в отличие от осетрины, бывают свежести весьма разной. И в итоге в Ubuntu не стало никакого особого гипермодерна: он основывался на репозиториях Debian тестируемой [tested] ветки, пригодность к использованию которой в десктопных условиях общепризнанна.

В результате уже через год, к осени 2005, обнаружилось, что Ubuntu — вполне зрелая система, пригодная к применению «искаропки» пользователем любого уровня. Разумеется, не без некоторых шероховатостей, касавшихся в первую очередь локально-зависимых вещей, но это было вполне естественно: обеспечить равную поддержку всех языков, от зулусского

до русского, за столь короткий срок физически невозможно. Да и лечилось все это достаточно просто. В том числе и руками утопающих.

Все сказанное выше было причиной того, что за пару лет количество пользователей Ubuntu достигло не просто некоей значимой цифры — оно превзошло число пользователей всех прочих дистрибутивов.

Не менее, чем количество пользователей Ubuntu, показателем их состав в сравнении с «более иными» дистрибутивами.

Так, в многочисленных опросах о первом дистрибутиве Linux на протяжении первой половины нулевых годов неизменно, и с большим отрывом, лидировал Mandrake/Mandriva. Но те же опросы о текущем дистрибутиве показывали, что после успешного старта с Mandriva изрядное число пользователей перетекало на другие дистрибутивы.

Для второй же половины нулевых годов картина совершенно переменялась. Первое место в опросах о первом дистрибутиве прочно занял Ubuntu. И в то же время процент пользователей, оставшихся верными этому выбору, был неизменно высок.

Кроме того, было (и есть) немало действующих пользователей Ubuntu, для которых этот дистрибутив был не первым, не вторым и даже не пятым. Тех, кто прошли и ручную настройку Slackware, и тотальную компиляцию Gentoo, и роман «Ядро и мир» от FreeBSD, а кое-кто — и сборку LFS. И чье сердце успокоилось в казенном доме — на тихом и уютном Ubuntu.

Интересно также, что среди пользователей Ubuntu высок процент тех, кто не имеет к компьютерам ни малейшего отношения — ни по долгу службы, ни по велению души. А разве что по жизни вынужден ими пользоваться. Тогда как среди пользователей иных дистрибутивов процент этот исчезающе мал. Более того, среди моих личных, реальных и виртуальных, знакомых (а круг и первых, и особенно вторых у меня весьма широк) вообще нет людей, не работающих в околокомпьютерных сферах или просто не интересующихся компьютерами как хобби, которые использовали бы какой-либо дистрибутив Linux'a. Разумеется, если этот Linux — не Ubuntu.

Так что буквально за пару лет Марку Шаттлворту, его фирме Canonical, примкнувшим к ним независимым разработчикам и не в последнюю очередь — активным пользователям, создателям сайтов и авторам блогов убунтийской тематики, удалось превратить, казалось бы, рядовую «человеко-мордастую» подделку в самый популярный и распространенный дистрибутив планеты.

Не будем пока оценивать это в терминах великого советского поэта, автора знаменитой поэмы про «хорошо и плохо»,

«Провидцы, даже будучи очевидцами, оказались не правы.»

а примем как медицинский факт. И посмотрим, что же из этого получилось.

Первые последствия

Первым и, пожалуй, самым заметным, следствием появления Ubuntu был параллельный рост популярности Gnome. Да-да, дорогие мои читатели — поклонники и болельщики этой рабочей среды: своей популярностью она обязана не команде разработчиков, не своим несравненным достоинствам (каковые в разное время трактовались диаметрально противоположным образом), и даже не усилиям Red Hat, продвигавшей ее со дня появления. А исключительно широкому распространению Ubuntu, где она выступала в качестве основной — а для начинающих пользователей и единственной рабочей среды. Ибо поначалу о существовании других сред они не подозревали.

Напомню, что первоначально в качестве объектов бесплатной рассылки выступали собственно Ubuntu и ее «образовательный» вариант — Edubuntu, обе — с Gnome в качестве рабочего окружения. Kubuntu с KDE включился в этот процесс существенно позднее и участвовал в нем не очень долго, а Xubuntu, кажется, не общился к нему вообще.

В сочетании с шоковым действием, оказанным на старых KDE'шников первыми версиями 4-й ветки этой среды, это привело к тому, что Gnome по популярности с ней почти сравнялся. Казалось бы, частное явление: изменение соотношения пользователей разных рабочих сред. Однако, как мы увидим позднее, оно тоже явится причиной ряда событий.

А вот второе следствие было более общим: лавинообразный рост источников информации о Linux'е. Ибо к середине нулевых годов писательский пыл линуксоидов первого призыва подыссяк. И причины того понятны. Проблемы, которые волновали поколения пользователей, начинавших свой путь в Linux еще в прошлом тысячелетии, и были поводом для сочинения всякого рода FAQ'ов, How-to'ев, Tips'ов и Hint'ов, ушли в прошлое. Во второй половине нулевых годов уже почти не требовалось ни ручной правки Иксовых конфигов, ни построения программных RAID'ов и систем LVM «с нуля», ни прикручивания файловых систем, отличных от умолчальной ext2, ни, в наших условиях, кириллизации. Все эти вопросы решались «искаропки», требуя в худшем случае небольшой косметической доводки. И в результате из прежних линуксописателей на вахте остались только те, для кого это занятие стало чем-то вроде профессии или очень серьезного хобби.

Однако на смену пришел эшелон новых пользователей, открывавших для себя вещи, казавшиеся ранее элементарными — и спешивших поделиться своими новообретенными знаниями. А поскольку это совпало по времени с массовым распространением блогов и прочих социальных сетей, возможностей поделиться было хоть отбавляй.

Можно сколько угодно иронизировать над сочинениями, хорошо описываемыми старым не очень цензурным анекдотом про Вовочку, вернувшегося 1 сентября из первого класса с новыми познаниями — и ваш покорный слуга, в силу природной язвительности, отдал дань этому занятию. Однако время расставило все по своим местам. Сайты и блоги вовсе тривиального содержания канули в небытие, более иные же развились в полноценные ресурсы. И ныне решение любой проблемы, имеющей отношение к Linux'у вообще, скорее всего может быть найдено на ресурсах убунтийской тематики.

Другое дело, что на таких ресурсах далеко не всегда делается различие между общелинуксовыми (и даже общеюниксовыми) решениями и решениями дистрибутив-специфическими. Но что

поделаешь: это следствие все той же популярности Ubuntu. Для многих пользователей которой понятия Linux, Ubuntu и, как уже было сказано чуть выше, Gnome (теперь Unity) соединены символами тождественного равенства.

Еще одно частное, но очень важное для конечного пользователя следствие распространения Ubuntu — появление в Иксах качественных шрифтов. Точнее, не столько даже самих шрифтов, сколько механизмов их экранного воспроизведения. Если раньше

» для практической работы широко применялись пиксельные шрифты;

» для просмотра в браузерах и офисных пакетах приходилось прибегать к шрифтам от классового врага (причем, желая соблюсти букву лицензии, нужно было понимать толк в извращениях);

» а умельцы вручную патчили freetype для включения поддержки интерпретации байт-кодов и субпиксельного хинтинга, то в Ubuntu качественный рендеринг шрифтов обеспечивался из все той же коробки.

И еще очень важный момент: Ubuntu, развивая традиции систем быстрого развертывания, довел до логического завершения безальтернативную «установку в пять кликов», милую сердцу начинающего пользователя. И в то же время сохранил идущую от Debian'a альтернативную текстовую инсталляцию (ныне, правда, только в виде образа **mini.iso**), допускающую широкое ручное вмешательство со стороны пользователя, твердо знающего, что он делает.

Список усовершенствований, предложенных в Ubuntu пользователю любой степени предварительной подготовки, можно продолжать еще долго. Поэтому закругляюсь очередным предварительным выводом: нововведения Ubuntu привели к повышению планки «юзерофильности» в хорошем смысле слова; или, если угодно, к понижению порога вхождения для начинающего пользователя. Что, по закону обратной связи, повлекло за собой рост числа этих самых начинающих. Так или иначе, с появлением Ubuntu мир Linux изменился, и это — та самая объективная реальность, с которой отныне должны были считаться все дистрибьютеры.

Реакция

Итак, Ubuntu понадобилось всего года два для того, чтобы добиться несравненной популярности среди узкого круга широких народных масс. Той самой, к которой на протяжении полутора десятков лет тщетно стремились и Red Hat в свою еще десктопную пору, и Debian во время своих самых широких имперских притязаний, и Mandrake с Mandriva при всей своей перманентной фронтирности. Как же прореагировали на это явление разнообразные дистрибьютеры всевозможных категорий сложности?

В первом приближении ответ очень прост: по-разному. Для начала, появление Ubuntu, в силу ее развитой инфраструктуры,

«Нововведения Ubuntu привели к понижению порога вхождения.»

Новый поворот

В 1997 году — а кто не помнит, это был год рождения массового российского Интернета, — некоторые московские провайдеры впервые стали предлагать хостинг не только на UNIX-машинах, но и на NT-серверах. Причем последний стоил существенно дороже. А мотивировалась дороговизна очень просто — непривычностью интерфейса для web-мастера. Судя по тому, что эта услуга в конце 90-х пользовалась большим спросом, аргумент действовал.

спровоцировало волну клонов: сначала официальных и полуофициальных вариантов со своими рабочими средами, затем — локализованных версий и версий, ориентированных на национальную специфику, а также специализированных систем (см. LXF155). В сущности, если не любая кухарка, то почти любой квалифицированный пользователь в состоянии собрать свой дистрибутив на базе Ubuntu. Другое дело, что как раз квалифицированные пользователи понимают бессмысленность этого занятия...

Иной была реакция со стороны ветеранов дистрибутирования. Первый вал упреков посыпался со стороны разработчиков Debian'a. Каковой сводился к двум основным пунктам: а) заимствованию всего из прародительского дистрибутива и б) несовместимости с ним на уровне пакетов и конфигов. Несмотря на то, что это положения во многом взаимоисключающие, некоторый повод к подобным упрекам Ubuntu давал.

Да, Ubuntu действительно почти целиком основан на пакетной базе из репозитория Debian — *tested* и частично *unstable*. Да, в ряде случаев пакеты из Debian'a не желали устанавливаться в Ubuntu, и чуть в большем количестве случаев — наоборот. Да, постепенно Ubuntu приобретал все большую дистрибуцифичность, а с переходом на схему инициализации *upstart* и внедрения Unity в качестве основного десктопа вообще отдалился от предка. Однако с течением времени ситуация по ряду позиций поменялась.

До сих пор «официальная» часть репозитория Ubuntu в основном (кроме собственных разработок, типа тех же *upstart*'а и среды Unity) повторяет репозитории Debian'a. Но существует и неофициальная часть инфраструктуры Unity — репозитории PPA (Personal Package Archive) и инструмент для работы с ними — Launchpad. Так вот, PPA-репозитории — неисчерпаемый клад пакетов самого разного назначения. И все новинки свободного софтверостроения в первую очередь появляются именно в них. Так что для упрека в паразитировании Ubuntu на Debian'e не остается никакой почвы. К тому же нынче в отношении двоякой совместимости пакетов для Debian'a и Ubuntu достигнут консенсус. Что же до дистрибуцифичности — тут уж ничего не поделаешь: любой активно развивающийся дистрибутив рано или поздно приобретает свою специфику.

С другой стороны, Ubuntu косвенно дал Debian'у очень немало. Во-первых, здоровая спортивная злость подстегнула разработчиков последнего, и они существенно сократили релиз-цикл. Во-вторых, видимо, из той же спортивной злости, были реализованы наконец некоторые задумки, обещанные и ожидаемые... даже не три года, а существенно больше: например, графический инсталлятор. А в-третьих и главных, популярность Ubuntu вызвала рост интереса и к родительской системе. И хотя я уже говорил о приверженности убунтуцев к своему дистрибутиву, все же немалое их число мигрировало по тем или иным причинам на Debian.

Так что к рубежу нулевых и десятых классовые противоречия между этими дистрибутивами можно было считать вчерашним днем. А временами возникающие конфликты — отнести скорее всего к противостоянию отцов и детей, ведь от этого конфликта не было избавлено еще ни одно человеческое поколение.

Однако к тому же времени — рубежу нулевых и десятых — отнесится обострение конфронтации на другой линии: RHEL/Fedora против Ubuntu. Ибо в Ubuntu замахнулись на святое — на сервера и прочий корпоратив, начав выпуск «долгоиграющих [LTS]» релизов. Не то чтобы Ubuntu Server вдруг в одночасье стал прямым конкурентом для серверов на RHEL. Более того, отношение к Ubuntu в амплуа сервера было еще более скептическим,

чем поначалу — к Ubuntu в роли пользовательского десктопа. По крайней мере, на Linux-ресурсах было хорошим тоном иронизировать по этому поводу. Кстати сказать, кое-где иронизируют и по сей день.

Но в Red Hat сидят люди серьезные, и им было не до иронии. Может быть, потому, что они вспомнили историю, начавшуюся в 1995 году. Она рассказывается во врезке — для тех, кому не довелось жить в то интереснейшее время.

Так вот, Ubuntu тоже начал свой путь с оккупации пользовательских настольных компьютеров. В том числе десктопов школьников и студентов. А поскольку, как я уже говорил, пользователи Ubuntu, в отличие от Mandriva, продемонстрировали завидное постоянство своих привязанностей, резонно было ожидать, что со временем эти самые школьники и студенты принесут его и на рабочие места, которые Red Hat с давних пор полагал своей вотчиной. Надо было принимать меры — и они были приняты в двух направлениях. Здесь я остановлюсь только на первом, нетехнологическом.

Оно выразилось в агитации и пропаганде, достойной лучших учеников товарища Ульянова в скобках Ленина. Когда все наше заранее объявляется прогрессивным, а все не наше — устаревшим и маргинальным. Большевицкий лозунг — «Кто не с нами — тот против нас!» — неожиданно прозвучал в исполнении

тех, кто считал себя (и считает до сих пор) оплотом свободы.

Не обошлось и без прямых уколов по адресу Ubuntu, дескать, недостаточно внимания уделяющего разработке ядра и вообще системных компонентов Linux.

А уж что до мелких подковырок относительно перекрашивания обоев или перетаскивания кнопок — так это просто смешно. Особенно если не забывать о том, что стратегия «перекрашивания обоев и перетаскивания кнопок», говоря очень фигурально, за несколько лет дала Ubuntu больше пользователей, чем всем остальным дистрибутивам — самые передовые патчи ядра за двадцать лет.

«Здоровая спортивная злость подстегнула разработчиков.»

Заключение

Меня терзают смутные подозрения, что к теме, затронутой в данной заметке, мне придется возвращаться еще не раз. Так что пора подводить предварительный итог и попытаться в первом приближении ответить на вопрос: в чем же заключается феномен Ubuntu?

Ubuntu начинал как сугубо десктопный дистрибутив для конечного пользователя, и, несмотря на наличие аналогов в виде СБР, ориентированных на ту же нишу, очень быстро преуспел на этом поприще. Однако, не останавливаясь на достигнутом, он тихо и незаметно расширяет сферу своей деятельности в двух противоположных от десктопа направлениях.

Первое — это серверные решения, реализуемые в виде периодически выходящих «долгоиграющих [LTS]» версий. Второе же — прямо противоположное: разного рода гаджеты, планшеты и прочие смартфоны. И если в серверной сфере Ubuntu тащился в хвосте не только за Red Hat и SUSE, но даже за прародительским Debian'ом, то здесь он оказался в числе передовиков производства. В том числе и потому, что Ubuntu одним из первых всерьез занялся адаптацией самого себя для альтернативных процессоров — ARM'ов всякого рода. Причем как организованно, так и частным порядком.

«Кратко резюмирую сегодняшний базар»: если раньше пользователь в основном приспособивался к миру Linux'a, то с появлением Ubuntu он впервые почувствовал, что и Linux-мир стал приспособливаться к нему.

Определить феномен Ubuntu короче у меня не получилось. LXF

СВОБОДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ВИРТУАЛИЗАЦИЯ: ЭКОНОМИЯ НА СТОИМОСТИ IT-ИНФРАСТРУКТУРЫ ДО 90%



Единая инфраструктура
на базе свободного
программного обеспечения

Нет лицензионных платежей —
расходы только на внедрение
и техническую поддержку

Минимальные затраты
на оборудование
за счет виртуализации



[www.linuxcenter.ru/shop/
linux-software/office/kitezh](http://www.linuxcenter.ru/shop/linux-software/office/kitezh)

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center

Управители окон: извлечения из истории

Алексей Федорчук извлек из закровов истории рассказы о менеджерах окон, с которыми он работал или которые ему довелось видеть.

Ряд последних заметок исторического цикла ([LXF168](#), [LXF169](#), [LXF172](#)) был посвящен истории X Window System и ее свободных реализаций, XFree86 и X.org. Однако ни слова не было сказано об истории того, как протоколы, спецификации и реализации претворялись в те самые графические интерфейсы, с которыми непосредственно имеет дело пользователь.

Терминологическое введение

Протоколы, спецификации и реализации... претворялись они в виде двух классов программ — оконных менеджеров, иногда именуемых также диспетчерами окон (WM — Window Manager), и интегрированных графических рабочих сред (Graphic Desktop Environment), которые называют также средами рабочего стола (DE — Desktop Environment) или, в просторечии, десктопами.

Чтобы в дальнейшем не путаться среди многочисленных реализаций графических интерфейсов, дам краткую вводную установку. Оконный менеджер — это программа, обеспечивающая внешний вид и функциональность элементов графического интерфейса. Под видом интерфейсных элементов понимается оформление окон, включая панель заголовка, рамки, линейки прокрутки, управляющие кнопки и меню. Функциональность же — возможность минимизации, максимизации, сворачивания окон, их перемещения, изменения размера, вызова различных наборов контекстных меню.

По большому счету от оконного менеджера ничего иного и не требуется. Однако, как мы скоро увидим, со временем развитые представители этого класса обрастали наборами привязанных к ним утилит и приложений, а также собственными средствами настройки и наборами предопределенных тем. Откуда оставался один шаг до интегрированных рабочих сред, то есть десктопов.

Десктопы также включают средства оформления окон и управления ими, то есть оконные менеджеры, собственные (как в KDE и *Xfce*) или заимствованные (как в Gnome и *LXDE*). Однако средства собственного конфигурирования, наборы тем и стилей, штатные утилиты и приложения входят в них уже в обязательном порядке. Хотя количество последних может быть различным — от всеохватного в KDE до весьма скромного в *Xfce* или совсем уж бедного — в *LXDE*. Важно, что все штатные программы десктопов характеризуются единством интерфейса, настраиваемого собственными конфигураторами среды.

Настоящая статья, как и следующая за ней, будет посвящена истории менеджеров окон. В статьях же дальнейших я расскажу о драматической истории десктопов и о разворачивавшихся вокруг них «священных войнах».

Я не берусь рассказать про возникновение и развитие всех оконных менеджеров, когда-либо существовавших на свете — это задача, непосильная для одного человека. Поэтому ограничусь только теми представителями семейства, с которыми работал непосредственно или которые внимательно рассматривал на предмет практического применения.

Предтечи управителей

Практическая работа в X Window System без менеджера окон почти невозможна, или, по крайней мере, очень неэффективна. Тому, кому доводилось видеть «голые Иксы», понятно, почему: это просто серое поле с курсором мыши в виде крестика. И никакое щелканье мышинными кнопками не влечет за собой ни малейшего результата.

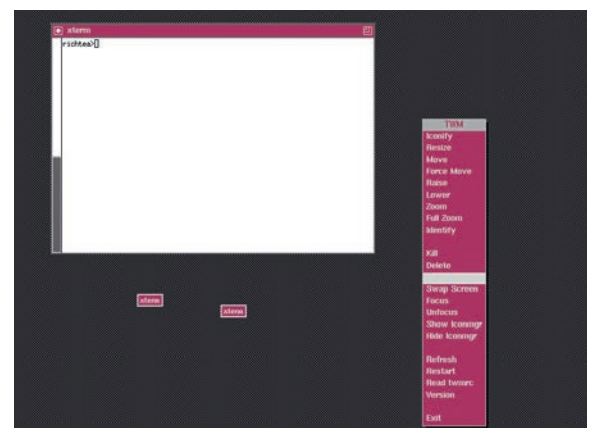
Конечно, в пользовательском конфиге Иксового сеанса можно прописать автоматический запуск каких-либо приложений — например, собственного эмулятора терминала, *xterm*, и работать

в его командной строке как в текстовой консоли. Можно было открыть при старте Иксов много терминальных окон для одновременной работы. И я знаю применителей, которые вполне серьезно по-

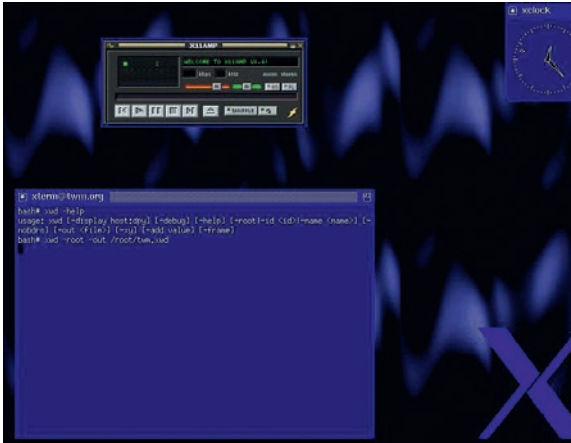
лагали это главной сакральной функцией графического интерфейса — ибо ни в чем, кроме терминалов, не нуждались. Можно было, наконец, открыть много-много приложений — но радости это приносило мало: исключительно иксовыми средствами окна их нельзя было ни перемещать, ни масштабировать, ни даже закрывать, не говоря уж о сворачивании в пиктограммы.

Так что можно предполагать, что оконные менеджеры возникли одновременно с первыми реализациями X Windows System,

«X Window System без оконного менеджера неэффективна.»



► Рис. 1. *twm* — до недавнего времени стандартный менеджер окон во всех дистрибутивах.



► Рис. 2. Внешний вид *twm* от создателя.

однако память о них затерлась. И первое о них упоминание обнаруживается только в X10 (конец 1985 года) под именами *xwm* и *hwm*, но сведений о том, что они собой представляли, мне обнаружить не удалось. По косвенным данным можно предполагать, что управлялись они не мышью, а с клавиатуры, комбинациями с участием клавиши Meta, и не имели средств конфигурирования.

В том же 1985 году компания DEC разрабатывает оконный менеджер *uwm* (*Ultrix Window Manager*). Он предназначался для ее собственной реализации UNIX для платформы VAX — Ultrix (в последующем, True64 UNIX), в которой Иксы не использовались. Однако *uwm* был быстро портирован на них, и уже в X11R3 стал стандартным средством управления окнами. Это был первый оконный менеджер, в котором с помощью файла конфигурации можно было настроить поведение кнопок мыши и привязать к ним меню управления окнами — функции, которые нынче кажутся столь тривиальными, что в гипермодернистских средах типа Gnome Shell и Unity они почти редуцировались.

У истоков управления окнами: *twm*...

Следующим этапом в развитии оконных менеджеров стал *twm*, разработанный Томом Ластранжем в 1987 году и включенный в качестве стандартного компонента в Иксы, начиная с X11R4 (декабрь 1989 года). Откуда он и попал в XFree86, появившуюся, как помнит читатель, в феврале 1991 года (**LXF168**).

В отличие от ранее упомянутых оконных менеджеров, *twm* могли бы наблюдать многие из ныне действующих линуксоидов. Хотя развитие его прекратилось, *twm* до недавнего времени в качестве стандартного средства управления окнами входил практически во все сборки Иксов — как в XFree86, так и в Xorg. А некоторым довелось и повидать его: именно *twm* запускался по умолчанию в ответ на команды **startx**, если в пользовательском конфигурационном файле Иксов не было определено

twm — что в имени его?

Создатель *twm*, Том Ластранж [Tom LaStrange], разрабатывал этот оконный менеджер для себя — и, естественно, назвал его собственным именем: первоначальной расшифровкой аббревиатуры было Tom's Window Manager. Такая практика в те годы была обычной (вспомним, например, Joe — Joe's Own Editor, то есть Личный Редактор Джо, Джозефа Аллена [Joseph Allen], его создателя) и отражала не манию величия или стремление увековечить свое имя. А напротив, как бы говорила: эту программу я сделал для себя. Подразумевая в скобках: а подойдет ли она вам — не знаю.

При включении *twm* в штатный комплект Иксов Том передал права на свое производство X-Консорциуму, стоящему в то время у руля управления графическими интерфейсами в UNIX'ах. И *twm* перестал быть

его личным инструментом, сделавшись всенародным достоянием (на условиях X-лицензии, разумеется). К тому же новые разработчики добавили в него функцию объединения заголовков окон в единую панель с закладками (позднее нечто подобное будет реализовано во *FluxBox*'е, а сама идея закладок нашла применение в бесчисленном количестве прикладных программ; по слухам, не так давно закладки появились даже в *Internet Explorer*). Так что *twm* с полным на то правом был переименован в *Tabbed Window Manager*.

А через четверть века после своего возникновения получила распространение иная расшифровка имени *twm*: *Timeless Window Manager*. Что, применительно к случаю, я перевел бы как Оконный Менеджер Всех Времен (а возможно, и народов).

ничего другого. Ныне место «умолчального» (он же «аварийный») оконного менеджера в ряде дистрибутивов Linux занял *IceWM*. Однако и *twm* до сих пор сохраняется во многих сборках: например, его можно обнаружить в стандартной установке openSUSE.

Основным средством управления окнами в *twm* окончательно стала мышь. С ее помощью, по щелчку правой кнопкой, вызывалось контекстное меню рабочего стола для запуска приложений и прекращения их работы, выполнения системных действий, таких, как выход из сеанса или рестарт машины, манипулирования открытыми окнами. Впрочем, управление окнами можно было осуществлять и действиями в строке заголовка — опять же с помощью мыши.

Собственного средства конфигурирования или каких-либо тем и стилей *twm* не имел: все настройки осуществлялись правой единственной и весьма простого по устройству конфигу — **twmrc**. Что, тем не менее, позволяло добиваться весьма причудливого и эффектного внешнего вида.

Современный оконный менеджер, и тем более десктоп, трудно представить себе без виртуальных рабочих столов — некоего аналога виртуальных терминалов консольного режима. Однако в *twm* их еще не было. Зато допускалось применение виртуального разрешения экрана, и во времена, когда преобладали мониторы с физическим разрешением 640×480, а режим 800×600 считался предметом роскоши, это было более чем востребовано (см. врезку).

»

Разрешение реальное и виртуальное

Виртуальное разрешение экрана — функция не оконного менеджера или рабочей среды, а X-сервера, и задается в его конфигурационном файле. Оно может быть в полтора-два раза больше максимального физического разрешения монитора. Собственно, верхний его предел определяется только объемом видеопамати.

При включении виртуального режима на экране видна только часть рабочего пространства (например, четвертинка, если виртуальное разрешение

задано вдвое большим, чем физическое). Для доступа к невидимым его участкам достаточно подвести курсор мыши к правому или нижнему краю экрана, чтобы плавно переместиться как бы за его пределы.

Ныне, в эпоху больших LCD-мониторов, виртуальное разрешение экрана устанавливается редко, обычно для каких-то специальных задач, и многие современные линуксоиды даже не подозревают о его существовании. Однако во времена,

когда самым ходовым размером экрана было 14, много 15 дюймов, это была палочка-выручалочка при использовании таких нагруженных интерфейсными элементами рабочих сред, как KDE.

Кстати, оверлейный режим Gnome Shell, которые создатели «третьегонома» продают как последний писк прогресса, не что иное, как гальванизованная идея виртуального разрешения. Воистину, если не все, то многое новое — это основательно испорченное старое.

В сущности, уже в *twm* почти в современном виде сложилась парадигма работы в графической среде, которая и поныне применяется в большинстве оконных менеджеров и рабочих сред. И хотя развитие его прекратилось очень давно, еще в прошлом тысячелетии, он оставил несколько прямых потомков и немало косвенных. А идейное его влияние в той или иной мере прослеживается абсолютно во всех оконных менеджерах и рабочих средах.

...и его линия

Как уже было сказано, *twm* не поддерживал виртуальных рабочих столов. Однако эта функция появилась в его ближайших прямых потомках — *vtwm* (Virtual TWM) и *vtwm* (Tom's Virtual Tab Window Manager — опять же разработка Ластранжа для личного пользования). Которые, в сущности, только ею и отличались от родителя.

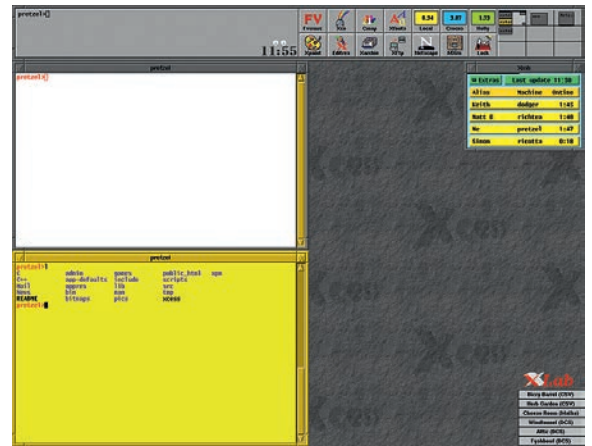
Из косвенных потомков *twm* наибольшее распространение и известность приобрел *FVWM*, что изначально расшифровывалось как Feeble Virtual Window Manager (то есть «хилый виртуальный менеджер окон»), однако в дальнейшем значение первой литеры забылось, и она стала восприниматься символически. Он был создан в 1993 году Робертом Нэйшном [Robert Nation]. И сначала — также для личного применения. Однако, будучи заодно автором эмулятора терминала *gxvt*, Нэйшн начал распространять их совместно — и *FVWM* был принят народом на ура.

Вскоре Роберт прекратил заниматься *FVWM*, и его на посту основного разработчика сменил Чарльз Хайнс [Charles Hines], который внес изменения в формат конфигурационного файла, дополнив его рядом новых возможностей. Получившийся в результате оконный менеджер стал известен в народе как *FVWM2*, хотя до сих пор оба названия часто употребляются как синонимы.

FVWM, особенно в варианте *FVWM2*, прославился своей истине безграничной настраиваемостью — как функционала, так и внешнего вида. Правда, достигалась она кропотливой работой конфигурационного файла, причем далеко не очевидной, требующей вдумчивого чтения документации. В этом отношении его можно сравнить с позднейшим KDE: хотя в нем настройка выполняется через собственный графический интерфейс, однако он тоже далеко не прозрачен и требует освоения. Впрочем, в обоих случаях затраченные усилия окупаются неизменно превосходным результатом.

Вследствие своей конфигурабельности *FVWM2* завоевал большую популярность, и до сих пор широко распространен

«Затраченные усилия окупаются превосходным результатом.»



➤ Рис. 4. Внешний вид *FVWM* безгранично настраивается...

в узких кругах терпеливых и аккуратных применителей. И почти неизвестен за их пределами. Однако он сыграл большую роль в становлении следующей волны оконных менеджеров.

По стопам Windows 95

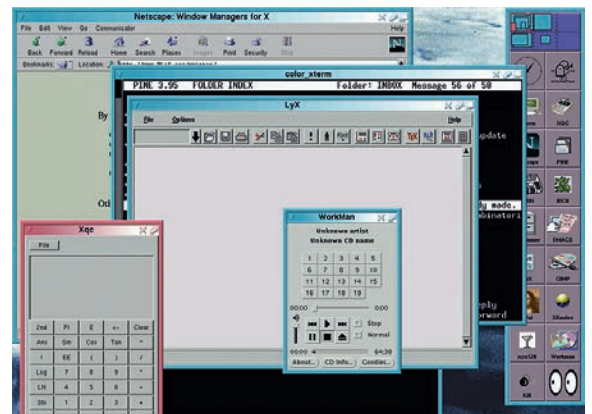
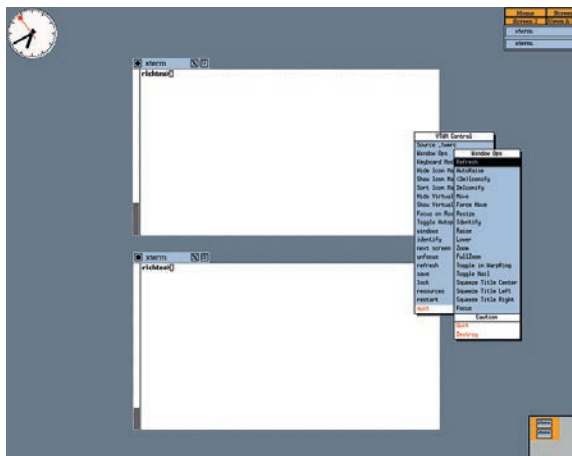
Летом 1995 года появляется Windows 95 со своей сакраментальной кнопкой Пуск (она же Start). И сразу обретает бешенную популярность среди windows-профов (windows-профи в это время продолжают применять Windows 3.1/3.11 для офисных задач и NT 3.X — для задач всамделишных).

По закономерной случайности в это же примерно время Linux делает первую попытку обратиться лицом к пользователю в корпоративном его исполнении (**LXF150**), а энтузиасты-линуксоиды начинают первую волну пропаганды своей любимой системы частным порядком, среди широких народных масс. А так как последние уже вкусили от прелестей кнопки Пуск, предпринимаются попытки обеспечить их таковой и в графических оболочках Иксов.

В результате в 1996 году на свет божий появляется *fvwm95* — весьма причудливая имитация интерфейса Windows 95. Она была образована из несколько облегченного

го *FVWM* с прикрученной к нему панели задач в win-стиле и, разумеется, кнопкой Пуск. Собственных средств конфигурирования по-прежнему нет, но настраивать стало легче, стало веселее. Потому что конфигурируемых параметров стало меньше, по сравнению с прототипом. В общем, на меня некогда этот оконный менеджер произвел впечатление откровенной пародии — причем

➤ Рис. 3. В *vtwm* появляются виртуальные десктопы.



➤ Рис. 5. ...но настройки *FVWM2* еще безграничной.

Про win-профи и win-профов

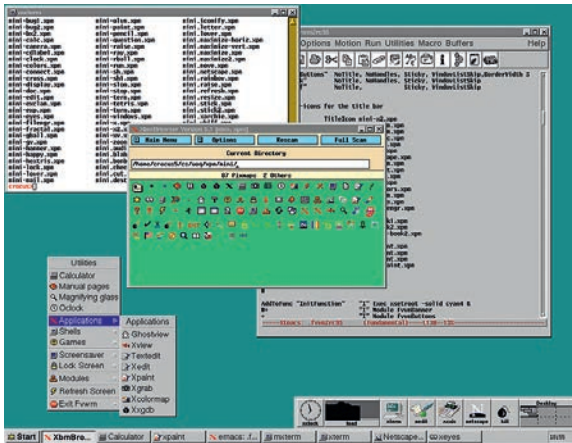
Впреки устоявшемуся мнению, победное шествие Windows той линии, которую в фольклоре именovali «оболочкой дешевой», началась не сразу с появлением Windows 95. Свидетельством чему — появившаяся в «Компьютере» колонка тогдашнего ее главного редактора, Георгия Кузнецова, о профи и профах. В которой прозвучала пророческая фраза (цитирую по памяти): «Профы поустанавливают

Windows 95 дома, потом притащат ее на работу, и профи придется с этим разбираться».

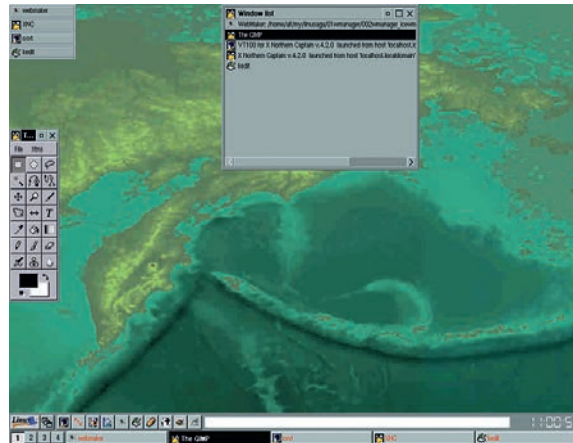
Так оно и случилось. Хорошо помню время, когда на моей тогдашней службе проходила массовая замена внедренной в качестве «фирменного стандарта» OS/2 на Windows 95 и затем на Windows 98. И проходила она по пожеланиям трудящихся. А поскольку среди трудящихся велик был процент

не только простых докторов наук, но и академиков с членкорами, пожелания эти по силе своей были близки к армейскому приказу.

Кстати, именно тогда мы впервые массово применили Linux в условиях производственного десктопа. Оказалось, что самый быстрый и простой способ искоренения OS/2 — загрузка Linux'a с дискеты и запуск в нем команды `dd if=/dev/zero of=/dev/hda`.



► Рис. 6. *fvwm95* — имитация или пародия?



► Рис. 8. *IceWM*? Собственная тема? Не вопрос...

и на *FVWM*, и на Windows 95. Судя по тому, что он очень быстро сошел со сцены, я был не одинок в своем мнении.

Однако дело кнопки Пуск не пропало. Его подхватили... и так далее, по Ульянову в скобках Ленину, разработчики других GUI, начиная с *IceWM* и кончая героями... Но о героях речь пойдет в одной из последующих статей. А здесь поговорим об *IceWM*.

Если *fvwm95* выглядел поделкой, слепленной на скорую руку на потребу win-профам, *IceWM*, разработка Марко Мачека [Marko Macek], начатая им в 1997 году, производила впечатление продукта, сделанного не только с умом, но и с любовью. Кроме того, он был написан «с нуля», а не основывался на коде *FVWM*, хотя без идейного влияния последнего и не обошлось (в частности, в плане гибкости конфигурирования).

Основных особенностей, выделявших *IceWM* в ряду прочих оконных менеджеров (а я выше упомянул далеко не всех его современников — одних только эпигонов *FVWM* тогда существовало с полдюжины), изначально декларировалось три:

► интуитивно понятный интерфейс (кто видел умолчальный *FVWM*, поймет, о чем речь) с возможностью гибкой индивидуальной настройки (в этом отношении если *IceWM* и уступает *FVWM*, то не намного);

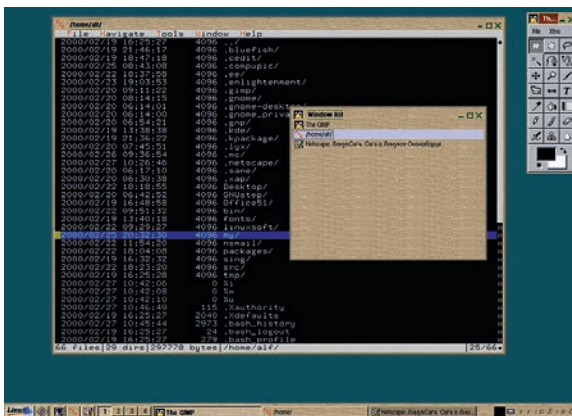
► доступность полного функционала интерфейса с клавиатуры, без использования мыши, что для большинства современников было затруднительно: в те времена мне приходилось слышать высказывания, что в Иксах без мыши работать вообще невозможно; так вот, *IceWM* был живым их опровержением;

► быстродействие и минимизация потребления ресурсов.

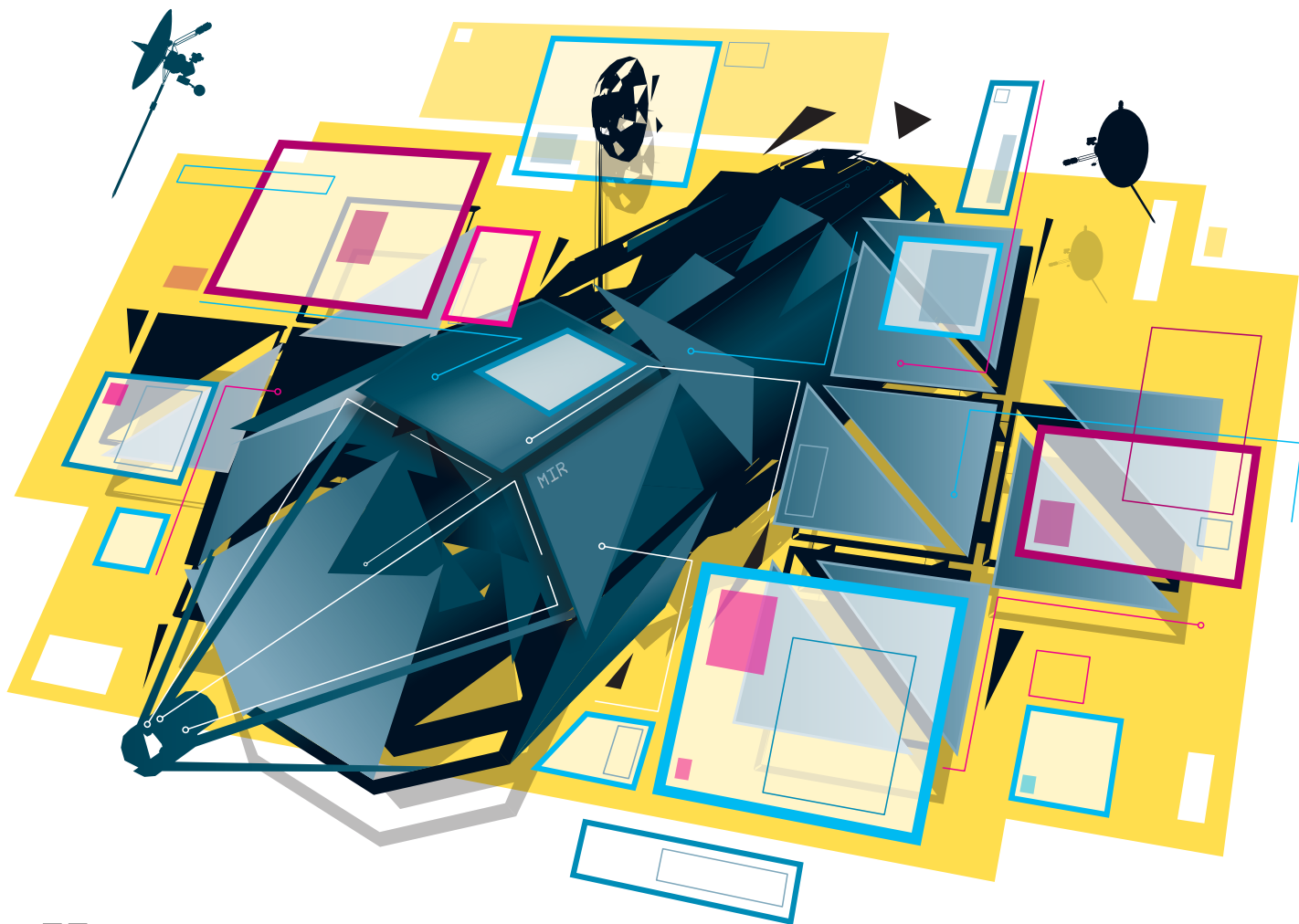
Все эти цели были достигнуты — плюс еще ряд дополнительных. Например, хотя изначально предусматривалась только настройка *IceWM* традиционным путем правки конфигов (их довольно много, но они просты и понятны), очень быстро он был дополнен средством собственной настройки — графической утилитой *IcePref*. А простота создания тем для *IceWM* привела к тому, что таковых было создано много: кроме умолчальной стилизации под Windows 95, в сети можно было найти множество тем, воспроизводящих внешний вид OS/2, Motif и многих других систем. При наличии желания и толики свободного времени не составляло труда сделать и собственную тему.

Оконный менеджер *IceWM* продолжает свое развитие до сих пор: очередные его версии выходят не часто, ибо кардинально улучшать в нем уже нечего, но зато регулярно — в соответствии с необходимостью подгонки под новые версии библиотек, с которыми он связан зависимостью. И, как уже было сказано, в сборках X.org некоторых дистрибутивов *IceWM* сменил *twm* в качестве «умолчального» оконного менеджера, который может быть запущен вне зависимости от наличия или целостности любых других графических сред.

Однако, как мы увидим в следующей статье, свет клином не сошелся на линии *FVWM* и Windows-подобии. Напротив, наиболее яркие представители семейства оконных менеджеров представляли собой линии иные. **LXF**



► Рис. 7. *IceWM* — с умом и любовью.



Что за штука...

Mir

Марко Фиоретти представляет тот самый противоречивый проект в области графики, обещающий привести Ubuntu на вершину мира.

В И что же за штука *Mir*, помимо бывшей российской космической станции?

О В сфере свободного ПО это графический сервер, который компания Canonical, владеющая Ubuntu, разрабатывает как альтернативу *Wayland* (<http://unity.ubuntu.com/mir>).

В Про *Wayland* вы рассказывали в LXF142. Не резюмируете еще разок?

О Легко. Системы, подобные Unix, используют для отображения графики отдельный сервер, независимый от ядра. *Wayland*, находящийся сейчас в разработке, призван сменить популярнейший графический сервер для Linux, называемый *X.Org* или просто *X*. В *Wayland* довольно много новшеств, в том числе, как и в *Mir*, компоновщик и подсистема обработки ввода.

В По порядку, пожалуйста. Что такое компоновщик?

О Это менеджер окон — он следит, чтобы окно каждого приложения создавалось в отдельном буфере, а затем соединяет их, зачастую добавляя 3D и другие визуальные эффекты. Эталонный компоновщик *Wayland* называется *Weston*.

В Спасибо. А про обработку ввода можно тоже объяснить?

О Ввести текст в компьютер можно массой способов: например, прикасаясь к сенсорному экрану, одновременно нажимая клавиши (особенно для безалфавитных языков), через распознавание речи и почерка, а также при помощи специальных приспособлений для пользователей с ограниченными возможностями. Программы

обработки ввода конвертируют все эти аппаратные события в текстовые байты. В современных интерфейсах их положение в графическом стеке имеет большое значение.

В Почему, собственно говоря, это так уж важно?

О Изменение системы ввода, как того требует *Wayland*, может изменить требования его совместимости с другими программами, что гарантирует стабильную работу. Прямое взаимодействие с инструментариями GUI, например, *Qt* или *GTK*, может добавить системе гибкости, но нужно приложить немало совместных усилий. Переместить систему обработки ввода вниз по стеку означает (грубо говоря), что она должна быть совместима только с графическим сервером.

В Хорошо, но что же дурного с графическим сервером *Wayland*, коли Ubuntu (и только Ubuntu) пошел другим путем?

О В качестве замены *X*, *Wayland* в основном нацелен на настольный вариант. Canonical же, напротив, стремится к полной конвергенции (при приоритете мобильных устройств), считая, что чем скорее она настанет, тем лучше.

В Полной конвергенции? Что это означает в бытовом плане?

О Это означает, что приложения и пользовательские интерфейсы должны быть не только максимально отзывчивы, но и просты в освоении и использовании, будучи едины для всех устройств. То есть графический сервер должен делать только то, что от него требуется, и интегрироваться со всем остальным, от обработки ввода до оболочки рабочего стола и менеджера окон.

В И что же, с *Wayland* это оказывается невозможно?

О Ну, в Canonical считают так. Конечно, в открытом коде можно выполнять любую отладку и настройку того, что в сторонних протоколах закрыто, но это лишняя работа. Canonical хотят также максимально автоматизировать тестирование ПО. А если обновления, способные что-то повредить, приходят извне, это затрудняет дело. Наконец, в начале 2013 в Canonical решили,

О Некоторые разработчики просто отвергли тот довод, что Canonical не сможет получить желаемое, доработав *Wayland*. Другие указали на то, что некоторые технические претензии Canonical, тот же статус обработки ввода, уже не столь насущны, как несколько месяцев назад.

В А что подразумевалось под «управленческими причудами»?

О Главу Ubuntu, Марка Шаттлворта, обвиняют в желании столь тотально контролировать Ubuntu, что ему некогда убеждать других. В особенности раз он так стремится стать ключевым игроком на мобильном рынке.

В Но в мире свободного ПО такое случалось не раз, и в итоге обернулось пшиком...

О На этот раз суть в том, что Canonical достаточно велика, чтобы сделать большую часть работы самостоятельно, без поддержки сообщества, рискуя значительно усилить раздробленность внутри свободного ПО.

В В каком это смысле, усилить раздробленность?

О Во-первых, графическому серверу потребуются убедить производителей оборудования, таких как AMD, Intel и Nvidia, либо создать оптимизированные открытые драйверы, либо представить достаточно информации, чтобы это могли

В А с другой?. Если Mir почему-либо провалится?

О Обратная проблема. Скорее всего, Canonical будет портировать только в *Mir*, *Qt5* и *GTK3*, без поддержки более ранних версий. Кроме того, многие сторонние разработчики не смогут или не захотят применять множество заплат, годных только для одного дистрибутива. Поэтому поддерживать свободное ПО универсальным как для Ubuntu, так и для его производных, станет значительно сложнее.

В И это будет плохой новостью для таких проектов, как Kubuntu.

О Верно. Если, или когда, *Mir* приживется на десктопах настольно, чтобы Ubuntu смог уйти с *X.org*, этим дистрибутивам, возможно, будет проще вернуться к Debian, чтобы использовать сторонние приложения. Однако повторю еще раз, это только гипотетический и худший сценарий, который сбудется, только если графического стека *Mir* хватит для создания современного свободного рабочего стола.

В То есть предполагается, что до этого «дня» еще далеко? А когда Mir будет готов?

О На данный момент, *Mir* все еще в стадии альфа. Согласно некоторым заявлениям, возможно, он войдет в состав Ubuntu 14.04, но это очень жесткие сроки.

В Жесткие? Почему вы считаете этот срок нереалистичным?

О Поскольку это сложная программная среда, ее можно будет использовать, только когда будут готовы и совместимая с *Mir* оболочка Unity, и соответствующие графические драйверы.

В А, да, еще же графические драйверы! Это большое дело, не так ли?

О Да, особенно для сервера, который обязан обеспечивать работу на всех устройствах. Относительно настольной версии, Canonical ведет переговоры о поддержке с AMD и Nvidia, но на это потребуется время. Плюс в том, что какое бы решение они не нашли для *Mir*, оно сработает и в *Wayland*.

В А какова ситуация с мобильными графическими чипсетам?

О *Mir* должен быть полностью совместим с существующими драйверами для Android, с той же стабильностью, производительностью и энергопотреблением.

В Последний вопрос: каково лично *ваше* мнение об этом проекте Canonical?

О Даже если в Canonical действительно пытаются создать собственный Android для мобильных платформ, существует еще множество задач, для которых нужны «традиционные» системы. Так что я бы больше побеспокоился о том, насколько эта гонка затормозит переход с *X* в целом. Время покажет, насколько решение Canonical оправданно. **LXF**

«Wayland нацелен на настольный вариант. Canonical же стремится к конвергенции.»

что *Wayland* изрядно отстал от своего же запланированного графика, и вкладываться в него — наименее верный путь достичь желаемого в кратчайшие сроки.

В А помимо Canonical никто не делает ставки на конвергенцию?

О Нет, конечно, есть и другие. Похожую цель преследуют те же Plasma Workspaces из KDE: создать универсальную оболочку рабочего стола плюс оконный менеджер, чтобы автоматически адаптировать свое поведение под любое устройство. Разница в том, что Canonical хочет больше и скорее, и считает, что сможет всего добиться самостоятельно.

В А в мире свободного ПО от Mir не в восторге, я полагаю?

О «Не в восторге» — это мягко сказано. В первых откликах упоминались: «решение несуществующей проблемы», «управленческие причуды» и «у Canonical синдром NIH» (“Not Invented Here” — «Изобрели не у нас») [упорное нежелание пользоваться чужим опытом, — *прим. пер.*].

В Не могли бы вы прокомментировать такую реакцию?

сделать другие. Такие же проблемы есть и с настольными окружениями и инструментарием GUI. *Mir* станет третьей платформой, которую необходимо поддерживать, помимо *X.org* (на которой держится столько, что она не может просто взять и исчезнуть, как бы всем ни хотелось) и *Wayland*.

В Понятно: есть риск, что Canonical навяжет всем Mir, сделав его единственным сервером, который поддерживается из коробки?

О Да, это худший (долгосрочный) сценарий: из-за плотной интеграции *Mir* с очень популярным дистрибутивом никто не захочет работать над *Wayland*.

В Как Android и Google? Открытие, но не в той степени, как другие продукты, или по крайней мере не для всех сразу?

О Отличный пример для объяснения того, что может произойти.

В И каковы будут последствия для другого свободного ПО?

О С одной стороны, успех *Mir* может значительно затруднить портирование любых программ на базе Ubuntu (а значит, и на основе *Mir*) в другие дистрибутивы.



По рецептам доктора Брауна

Д-р Крис Браун

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

24 минуты

Время от времени мне звонят по телефону (и вам, я уверен, тоже) люди, для которых английский язык явно не родной; они заявляют, что неким туманным образом связаны с Microsoft, и пытаются внушить мне ложное чувство опасности, утверждая, что получили отчет об ошибке с моего компьютера и их долг — помочь мне ее исправить. Обычно я довольно бесцеремонно бросаю трубку, но вчера был в озорном настроении и водил за нос одного парня целых 24 минуты. Нимало не огорчившись известию, что у меня вообще нет Windows, он объяснил, что мой IP-адрес стал широко известен в итоге моей деятельности в браузере. (Разумеется, когда IP-адрес становится широко известен, это Очень Плохо.)

Оказалось, что он пытается раскрутить меня на установку программы для удаленного управления рабочим столом под названием *Ammy Admin* (прежде чем меня придут брать, спешу прояснить, что, насколько мне известно, *Ammy Admin* — абсолютно легальная программа). Будучи направлен на сайт *Ammy Admin*, я не смог не заметить жирной красной надписи на главной странице, которая ссылалась на страницу с объяснением, что если вам звонит некто, чей родной язык явно не английский, и сообщает, что неким туманным образом связан с Microsoft, то это жулик. Я озвучил парню данный текст, и тот сразу повесил трубку.

Я рискнул предположить, что этот тип перековался на честный способ заработка, но не уверен, что попал в точку. А если вам удавалось продержаться дольше 24 минут, я с радостью прочту ваши рассказы. chris.linuxformat@gmail.com

Эзотерическое системное администрирование из причудливых заворотов кишок серверной



Исследование сети

Nmap Узнать, что делается в сети, поможет одна из лучше всего в мире документированных утилит с открытым исходным кодом.

Оглядываясь назад, трудно поверить, что в 65 выпусках «Рецептов доктора Брауна» утилита *Nmap* упоминалась лишь мимоходом — ну, по крайней мере, до этого раза.

Вкратце, *Nmap* — средство «исследования сети». *Nmap* может найти все компьютеры в заданном блоке IP-адресов, определить открытые на них порты и собрать информацию о них — определить, какие версии сервисов на них запущены, и даже предположить, какая ОС установлена. Эта информация очень пригодится злоумышленникам, ищущим уязвимость для атаки, и уж конечно, они пользуются *Nmap*. Но ее можно использовать и в других целях: например, проверить, что брандмауэр работает так, как ожидается, или попробовать выполнить учебное вторжение.

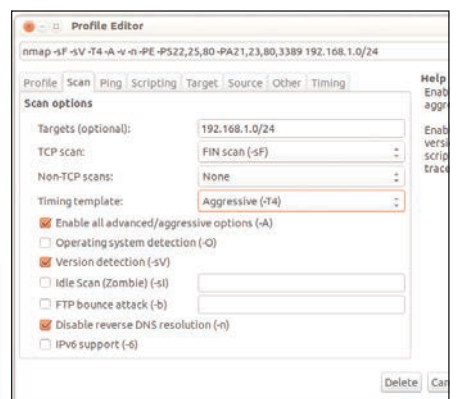
Nmap написан Гордоном Лайоном [Gordon Lyon] (известным также как Fyodor). Утилита вышла в 1997 году и до сих пор в активной разработке.

Простое сканирование может выглядеть так:

```
$ nmap scanme.nmap.org
Starting Nmap 5.21 ( http://nmap.org ) at 2013-05-02 16:24 BST
Nmap scan report for scanme.nmap.org
(74.207.244.221)
```

```
Host is up (0.17s latency).
Not shown: 998 closed ports
PORT STATE SERVICE
22/tcp open  ssh
80/tcp open  http
```

Параметр **-A** запускает обнаружение версии и ОС. Например, сканирование нашего сайта — `# nmap -A -T4 www.linuxformat.com` сообщит нам, что там запущен *ProFTPD 1.3.1* на порте 21, *Apache 2.2.8* на порте 80, *OpenSSH* на (нестандартном) порте 2020 и, вероятно, *Ubuntu*. Учтите, что можно получить лучшие результаты, запустив *Nmap* от имени `root`: при некоторых сканированиях выполняется подмена заголовков TCP, а на этом уровне нужны права `root`. **Не сканируйте свою корпоративную сеть без явного позволения. У вас могут быть серьезные неприятности!**

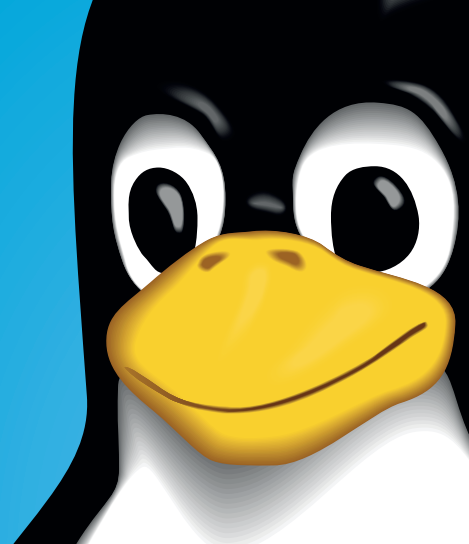


► Командные строки с *Nmap* бывают довольно запутанными — в качестве GUI возьмите *Zenmap*.

Триумф документации

Одно из лучших качеств *Nmap* — документация. Даже `man`-страницу приятно читать (о многих ли `man`-страницах можно сказать то же самое?). На сайте <http://nmap.org/book/toc.html> есть под-

робное руководство с примерами и задачами, а его печатная версия была (в течение короткого времени) самой продаваемой книгой на Amazon. Это ли не достижение?



Используйте свободное ПО — сэкономьте годовой бюджет!

Операционная система GNU/Linux поможет вам с **наименьшими затратами** решить проблему лицензирования программного обеспечения, навсегда избавиться от компьютерных вирусов и повысить надежность вашей компьютерной сети.



**С нашей
помощью
вы сможете**

**Сконцентрироваться
на своем бизнесе,**
не отвлекаясь на вопросы
поддержки своей
ИТ-инфраструктуры



**Забывать о вирусах,
угрозах безопасности**
и необходимости
лицензирования
программного обеспечения



**Оптимизировать
затраты**
на лицензирование ПО
за счет максимально
возможного использования
свободного ПО

ГНУ/Линуксцентр предлагает:

- Лицензирование во ФСТЭК.
- Внедрение наиболее дружественных вариантов ОС GNU/Linux и прикладных решений на базе свободного ПО.
- Абонентскую поддержку вашей сети.
- Обучение сотрудников вашей компании.

Наш опыт внедрения свободного программного обеспечения в организациях различного профиля поможет выбрать **оптимальное сочетание свободного и коммерческого программного обеспечения**, подходящее именно для вашей компании, а также поможет избежать технических и организационных проблем при внедрении свободного ПО.

Решите проблемы лицензирования ПО и поддержки компьютерной сети с помощью профессионалов!

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center

Уникален ли ваш уникальный идентификатор?

Универсальные идентификаторы Добрый Доктор показывает, как сгенерировать нечто гарантированно уникальное.

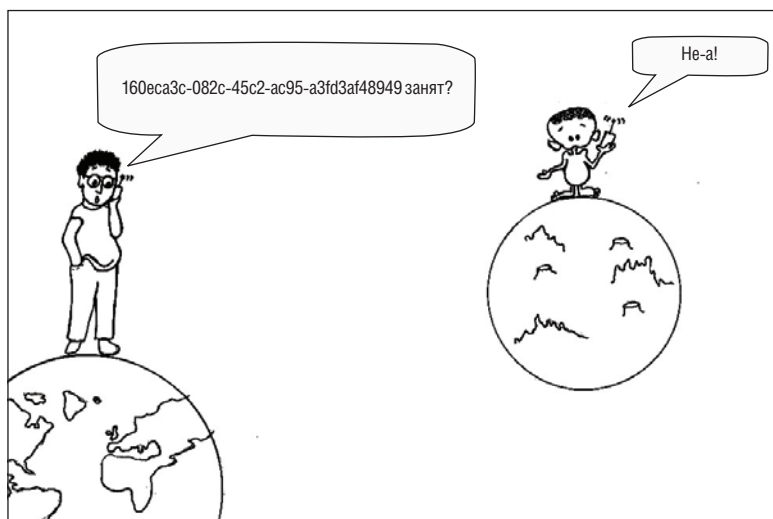
Есть много ситуаций, когда необходима гарантия уникальности. В Великобритании у всех граждан есть личные номера государственного страхования, у наших друзей в США — номера социального страхования. Номера паспортов тоже уникальны. У книг есть уникальные идентификаторы ISBN. Должны быть уникальными IP-адреса (не считая частных адресных пространств), и т.д.

Обычно механизмы обеспечения уникальности включают некий центральный орган (или иерархию таковых), который выдает или утверждает идентификаторы. (Подумайте о том, как отводятся IP-адреса и имена доменов.) Это недешевый бизнес, то есть получение нового идентификатора занимает минуты, часы, дни или даже недели. Насколько мне известно, идея генерации уникального идентификатора без центрального реестра впервые появилась в операционной системе Domain/OS в конце 1980-х и получила более формальное определение в виде универсальных идентификаторов (UUID), которые мы все знаем и любим, в статье «Распределенные сетевые вычисления» Фонда открытого ПО в начале 1990-х.

Если вам не приходилось видеть UUID, то он представляет собой 128-битное число, представленное в виде 32 шестнадцатеричных чисел, разбитых на пять групп 8-4-4-4-12, разделенных дефисами. Например:

```
4ecf3183-096d-4ded-80b0-53c2f96f564f
```

► Помните, что вы тоже уникальны. Как и все остальные.



Мне кажется, что универсальность уникальных идентификаторов трудно доказать. Представьте себе картинку из будущего: линуксоид беседует с маленьким зеленым человечком с планеты, вращающейся вокруг Капеллы В. «Смотри, — говорит линуксоид, тыкая в экран своего ноутбука. — Мы пользуемся UUID для уникальной идентификации разделов на наших дисках. Эти идентификаторы уникальны во всей Вселенной». «О, клево, — отвечает маленький зеленый человечек, — мы тоже!» Он присматривается и хмурит брови. «Хм-м... этот мне вроде знаком». Он сверяется с базой данных. «Да, мы уже пользовались им... на тюбике варенья из авокадо». — Укоризненный взгляд на линуксоида. «Знаете, все же следовало проверить...»

Да, но как?

Даже если сузить поиск планетой Земля, вопрос о гарантии уникальности UUID все равно актуален. Ну, UUID первой версии генерировались путем объединения MAC-адреса компьютера, который генерирует UUID, с моментальным снимком момента времени с точностью в 100 наносекунд. Оказывается, что первые 24 бита, которые производитель присваивает MAC-адресу сетевой карты, выдаются Институтом инженеров по электротехнике и радиоэлектронике (IEEE) — это так называемый OUI (Organisationally Unique Identifier — организационно уникальный идентификатор), и он — ну, организационно — уникален. Затем производитель (вероятно) назначает 24 последних бита уникальными для каждой карты. Если не генерировать более одного UUID в диапазоне 100 нс, уникальность этого подхода гарантируется. В Linux UUID первой версии можно сгенерировать командой

```
uuidgen -t  
13a1c098-b23d-11e2-8eab-001d095409e5
```

Выполните ее снова, и вы увидите, что последние 12 цифр не изменились. Это мой MAC-адрес, а чтобы узнать, какой организации этот OUI, зайдите на <http://standards.ieee.org/develop/regauth/oui/public.html> (не буду вас томить: 001d09 — это Dell).

В UUID версии 4 для 122 из 128 бит используются просто случайные значения (остальные шесть бит закреплены в стандартной структуре UUID — подробности см. в RFC4122).

Получить UUID четвертой версии в Linux можно командой

```
uuidgen -r  
aca7197d-1e01-4e2f-b0f0-b3a232409ea9  
или из ядра, командой  
cat /proc/sys/kernel/random/uuid  
d54f567e-f2b4-4d46-94e9-bbcdbe6c199
```

Уникальность UUID версии 4 нельзя полностью гарантировать, хотя перебор в лоб представляет собой убедительный аргумент. Цитируя Википедию, после генерации двух триллионов UUID (примерно по 300 на каждого жителя планеты) вероятность совпадения равна 0,000000000004, что довольно мало. Чтобы повысить вероятность одного совпадения до 50 %, каждый должен сгенерировать по 600 миллионов универсальных идентификаторов.

Так что же сказать об использовании UUID для гарантии уникальности имен разделов на диске в Linux? На ум приходит выражение «из пушки по воробьям». К тому же **/etc/fstab** из-за них ужасно захламлен!

Литературный конкурс с UUID

У меня есть для вас задание. Придумайте корректный UUID, который можно прочесть как предложение на английском. Для этого задания подойдет любая последовательность из 32 шестнадцатеричных цифр, и буквы можно заменять на цифры, напри-

мер, цифра 0 вместо буквы «o» и цифра 1 вместо «l». Вот пример (разумеется, правописание здесь несколько вольное):

```
Offalbad-cafe-f00d-1ed2-adeadbadc00c
```

Если у вас выйдет лучше, напишите мне на chris.linuxformat@gmail.com.

Как притвориться знатоком раздела подкачки

Раздел подкачки Хотите перессорить системных администраторов? Спросите их о разделе подкачки. Какого размера он должен быть? Нужен ли он вообще?

Одно из действий при установке Linux — отведение на диске места для раздела подкачки [swap]. Вообще-то большинство установщиков впадут в истерику, если этого не сделать. Но мнения о том, какого размера раздел подкачки должен быть и должен ли быть вообще, различаются.

Начнем с прояснения, зачем нужен раздел подкачки. В Linux много процессов, и всем подавай память. Через устройство управления памятью ядро пытается предоставить каждому процессу адресное пространство требуемого размера. На практике доступный процессам диапазон логических адресов делится на страницы (в Linux обычно по 4 К), каждая из которых связывается с физической страницей в ОЗУ. Когда общее количество страниц, необходимое всем работающим процессам, превысит количество доступных физических страниц, ядро освободит часть страниц, сбросив их содержимое на диск — как раз в раздел подкачки.

Таким образом, максимальное доступное адресное пространство — это сумма объема физической оперативной памяти и размера раздела подкачки. Само ядро тоже использует память, но эта память в раздел подкачки никогда не сбрасывается.

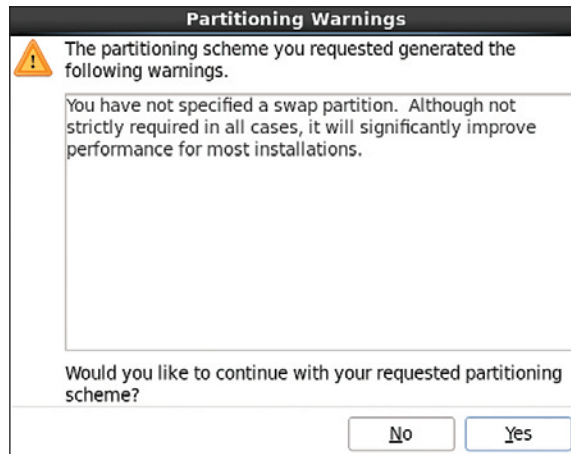
Каким должен быть размер раздела подкачки? Просто ответа на этот вопрос нет. В давние времена компьютеров, поддерживавших подкачку страниц по требованию (поднимите руки, кто помнит VAX), обычным правилом было делать раздел подкачки втрое больше объема ОЗУ (конечно, тогда 1 МБ ОЗУ было роскошью). Но ныне память очень дешева (я добавил 8 Гб на свой сервер всего за £40) — и можно наставить ее в компьютер столько, что тот обойдется без подкачки. То есть раздел не нужен вовсе.

Пробуксовка памяти

Однако с разделом подкачки можно добиться такой производительности, которая иначе невозможна. Пусть вы хотите одновременно запустить 6 виртуальных машин (ВМ), каждую с 1 Гб памяти, но физической памяти у вас всего 4 Гб. Подкачка хорошо работает тогда, когда система может обеспечить «рабочий набор» страниц (к которым достаточно регулярно обращаются запущенные процессы), фактически отображаемый на ОЗУ. При этом система может довольно интенсивно использовать подкачку, но перемещений страниц между разделом подкачки и ОЗУ сравнительно мало. В моем примере с запуском нескольких ВМ возможны задержки между переключением виртуальных машин, но как только рабочий набор страниц для новой ВМ отобразился, производительность должна стать отличной. Если рабочий набор страниц не вмещается в ОЗУ, подкачка работает хуже: системе приходится сбрасывать в подкачку страницу А, освобождая место для страницы В, только затем, чтобы сразу освободить его для страницы С (или, возможно, снова для А). Ситуация, в которой система тратит все больше времени на переброску страниц и все меньше — на полезную работу, называется «пробуксовкой» и оказывает негативное влияние на производительность.

Как это работает

Но довольно теории; перейдем к практике. Чаще всего при установке системы создается один раздел подкачки. Установщик добавит запись в `/etc/fstab`, наподобие



Установщики Linux притворяются, будто начнут капризничать, если не выделить раздел подкачки. Просто игнорируйте их!

```
/dev/sda6 none swap defaults 0 0
```

Во время загрузки выполняется команда `swapon -a` (например, в Red Hat это делает скрипт `/etc/rc.sysinit`). Она активизирует все разделы подкачки, заданные в `/etc/fstab`. Большинство пользователей принимают сценарий по умолчанию, а впоследствии уже не берут это в голову.

При желании можно расширить подкачку на дополнительные разделы. Пусть, например, вы добавили второй диск (sdb) и выделили под раздел подкачки `/dev/sdb2`. Сначала нужно инициализировать раздел для подкачки, следующей командой:

```
# mkswap /dev/sdb2
```

Теперь можно приказать ядру подхватить новый раздел:

```
# swapon /dev/sdb2
```

Однако, чтобы сделать изменения постоянными (то есть чтобы новый раздел активировался во время загрузки), нужно добавить в `fstab` строку

```
/dev/sdb2 none swap defaults 0 0
```

В Linux допускается задавать для разделов подкачки приоритеты. Если у вас два диска, работающих с одинаковой скоростью, вы можете сбалансировать нагрузку подкачки между ними, задав их приоритеты одинаковыми. Но если один из них более быстродействующий, можно распределить приоритеты с учетом данного неравенства. Например, если вы предпочитаете, чтобы для подкачки использовался `/dev/sdb2`, а не `/dev/sda6`, поместите в `fstab` такие строки:

```
/dev/sda6 none swap pri=2 0 0
```

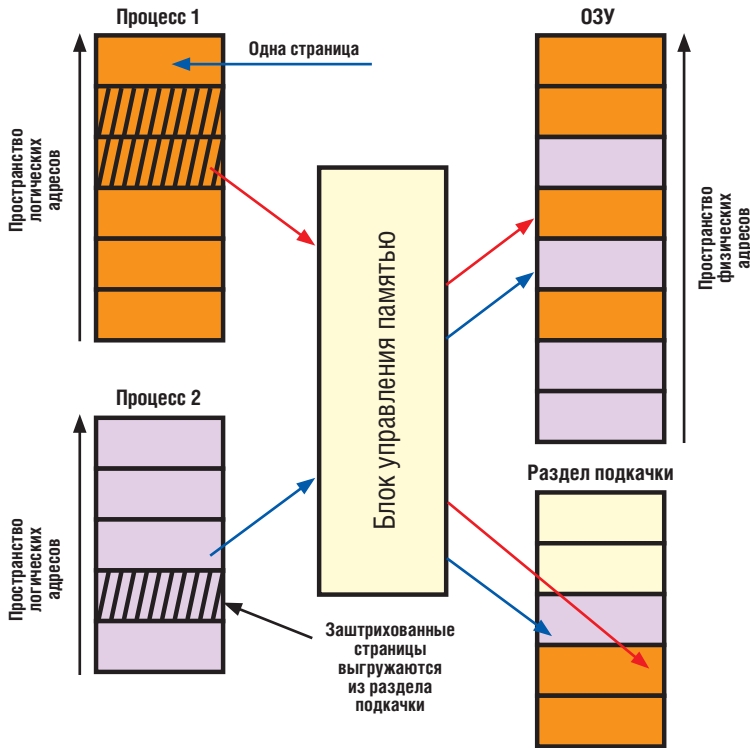
```
/dev/sdb2 none swap pri=3 0 0
```

»

Подкачка или страничная подкачка?

В традиционной терминологии есть различие между страничной подкачкой (когда на диск сбрасываются отдельные страницы) и просто подкачкой (когда на диск сбрасывается весь процесс). Насколько

я знаю, в Linux этого различия нет — там даже нет подкачки в традиционном понимании этого слова. Тем не менее, обычно мы говорим о подкачке, а не о страничной подкачке.



У каждого процесса есть иллюзия доступа к непрерывному пространству логических адресов, но реальность сложнее.

Для подкачки также можно использовать сброс в файл. На учебных курсах я говорил студентам, что подкачка в файл медленнее, чем прямо на раздел, но не думаю, что это так в современных ядрах Linux. Добавить файл подкачки явно проще, чем расширять или переразбивать физические диски, и это может стать простым краткосрочным решением при нехватке памяти.

Это делается так. Сначала нужно создать файл и убедиться, что весь его объем был выделен на подкачку. В Linux есть интересный трюк, за счет которого в файле могут быть «дырки», но для файлов подкачки это не работает, так что убедитесь, что данные были записаны в весь файл. Для этого скомандуйте

```
# dd if=/dev/zero of=/myswapfile bs=1M count = 1024
```

У нас получится файл в 1 ГБ, полный нулей. Затем подготовим файл для использования в качестве области подкачки:

```
# mkswap /myswapfile
```

Теперь можно добавить в `/etc/fstab` еще одну строку — `myswapfile none swap sw,pri=1 0 0` и активировать ее так:

```
$ sudo swapon -a
```

Наконец, можно проверить, какие пространства подкачки используются:

```
$ swapon -s
```

Filename	Type	Size	Used	Priority
/dev/sda6	partition	7811068	0	2
/myswapfile	file	1048572	0	1

Заметьте, что мы выставили приоритеты пространств подкачки таким образом, что раздел подкачки (`/dev/sda6`) более приоритетен, чем файл (`/swapfile`). Кстати, заглянув в файл `fstab` в своей системе, вы почти наверняка увидите, что раздел подкачки задан с UUID, а не с простым именем устройства.

Идет ли подкачка?

Есть несколько программ, которые покажут, какую часть пространства подкачки вы используете. Во-первых, это `Vmstat`. Это одна из немногих программ, показывающих, насколько активно страницы сбрасываются в подкачку и извлекаются оттуда. Потом есть старая добрая `top`:

```
$ top -n 1 -b | grep total
```

```
Tasks: 212 total, 1 running, 209 sleeping, 0 stopped, 2 zombie
```

```
Mem: 4048324k total, 3810436k used, 237888k free, 143032k buffers
```

```
Swap: 8859640k total, 6860k used, 8852780k free, 705668k cached
```

Команда `free` покажет во многом ту же самую информацию:

```
$ free -m -t
```

	total	used	free	shared	buffers	cached
Mem:	3953	3744	209	0	141	699
-/+ buffers/cache:	2902	1050				
Swap:	8651	6	8645			
Total:	12605	3750	8854			

Вывод команды показывает, что свободно только 5% памяти. Надо ли переживать по этому поводу? Не факт. Обнаружив свободную память, ядро использует ее в качестве буферного пространства для кэширования трафика файловой системы, что ускоряет доступ к часто используемым файлам. Но взгляните на цифры повнимательнее: вы увидите нечто странное. Система сбросила в подкачку 6 МБ, хотя свободно еще 209 МБ памяти. Почему она сделала это вместо возврата страниц из кэша?.. Что ж, ее поведение можно настроить, изменив остроумно названный параметр `swappiness` (от 0 до 100), который задает баланс ядра между возвратом страниц из кэша страниц и сбросом в подкачку памяти процессов. Если `swappiness` равен 0, то ядро сбрасывает данные в подкачку только в крайнем случае. Значение 100 означает агрессивную подкачку. По умолчанию параметр равен 60, как вы видите из команды

```
$ sysctl vm.swappiness
```

```
vm.swappiness = 60
```

Чтобы изменить значение, скомандуйте

```
$ sudo sysctl -w vm.swappiness=100
```

```
vm.swappiness = 100
```

Чтобы изменение сохранялось после перезагрузки системы, откройте файл `/etc/sysctl.conf` и добавьте следующую строку:

```
vm.swappiness = 100
```

Однако позвольте повторить совет, который я уже давал в этой статье. Те, кто задал значение по умолчанию, лучше понимали, что происходит, чем вы и я. Изменяйте его, только твердо зная, что вы делаете, и обладая объективным способом измерить изменение в производительности, вызванное вашей поправкой.

Наконец, в настольных системах есть еще одна причина выделения пространства подкачки, не имеющая ничего общего с подкачкой. Если попросить систему перейти в спящий режим, она запишет текущее содержимое памяти на диск, а затем отключит память (и все остальное). Для этого используется пространство подкачки. И если вы хотите, чтобы ноутбук можно было перевести в спящий режим, создайте раздел подкачки размером не меньше объема оперативной памяти.

Итак, в следующий раз, попав на собрание системных администраторов, не поднимайте вопрос о пространстве подкачки. На него нет «правильных» ответов. К тому же есть вопросы и поважнее, например, «Чья очередь выставлять пиво?» **LXF**

Знайте свою историю

Unix вступила в жизнь на компьютерах PDP7 и PDP11, не поддерживавших виртуальную память. Вместо нее применялся обходной прием — бит “sticky bit”, бывший частью режима доступа к файлу. Если он устанавливался для исполняемого файла, Unix понимал, что файл часто используется и должен по возможности остаться

[stick around] в памяти после завершения работы программы. Появление виртуальной памяти сделало этот бит излишним. Однако он нашел работу по режиму доступа к каталогам Linux, определяя, кто может удалять файлы из каталога. В Linux этот бит все еще можно задать для файлов, но эффект, по-моему, нулевой.

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший
свою квалификацию
физик.

Сохранить и проанализировать

Лучший способ спрогнозировать будущее — изобрести его.
Алан Кёртис Кэй

Человечество производит данные... Нет, не так. Человечество данными просто фонтанирует, а эти данные имеют привычку теряться. И первое, о чем попыталось озаботиться человечество — это сохранить их. Неважно какие — главное, сохранить, и по возможности все. Так родились базы данных, которые при разумной их эксплуатации сохраняют вброшенную в них информацию и отдают конкретный ее сегмент по запросу.

Казалось бы, счастье наступило — все можно сохранить и складировать, но что с этими горами не шибко связанных данных делать? Для этого нужны специалисты, а специалисты говорят не на проверенном временем языке баз данных SQL, а на языках анализа данных, например, R — фактически стандарт среди специалистов по статистической обработке (к сожалению, не в России). СУБД без встроенных в них специализированных инструментов анализа, да того же R, по всей видимости, в обозримом будущем вымрут как динозавры или, в лучшем случае, забьются в нишу баз данных одного файла, уже плотно занятую *SQLite*.

Проект *SciDB* от прародителя *PostgreSQL* Майкла Стоунбрейкера [Michael Stonebraker] изначально планировался как база данных, где данные не просто хранятся, но анализируются. Сейчас у нее появился интерфейс к R, так что все идет по плану, в конце которого *SciDB* захватит мир информации.
e.m.baldin@inp.nsk.su

В этом месяце вы научитесь...



Украшать LibreOffice 68
Эндрю Грегори научит вас создавать стили в *LibreOffice*, чтобы ваша изящная проза смотрелась круче.



Стартовать RPi 70
Намучились со включением Raspberry Pi? **Грэм Моррисон** рассказывает, как загрузиться и исправить ошибки.



Выставлять квоты 74
Нейл Ботвик умеет ограничивать аппетиты пользователей, которые норовят монополизировать диск.



Размножать MySQL 76
Одна база данных запросто обслужит несколько серверов, как показывает **Джонатан Робертс**.



Применять CUDA 80
Бен Эверард поможет вам скинуть на графическую карту интенсивное перемалывание числовых данных.



Вникать в Erlang 84
Андрей Ушаков методично ведет нас к идеалу многозадачных версий функций.



Монтировать видео 88
Введение в монтажные программы от **Петра Семилетова** позволит вам осознать, за что хвататься.



Создавать 3D-игры 92
Только для отважных! **Андрей Прахов** начинает эволюцию игровых персонажей с червяка.

Код в учебниках

Строки исходного кода помещаются в цветные плашки. Если строка кода не помещается в колонку, ее остаток переходит на следующую строку, в той же плашке:

```
procedure TForm1.Button1Click
begin
  mniWordWrap.Checked := false
```

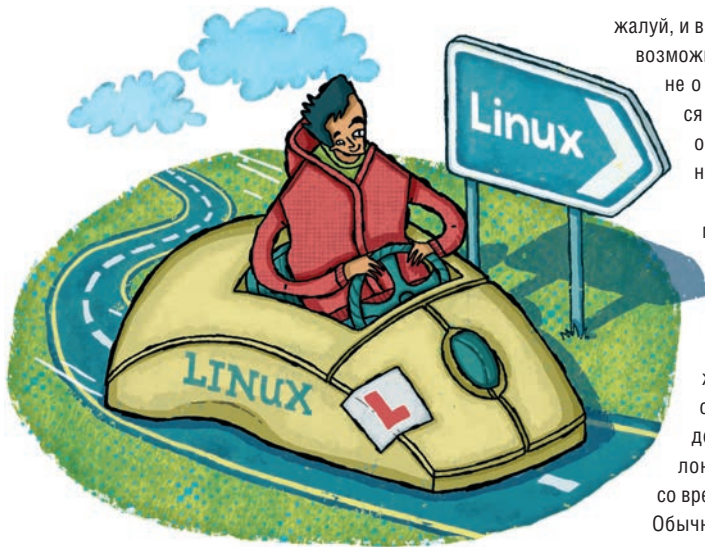
LibreOffice: Ключ

Эндрю Грегори выходит за рамки примитивно-люмпенских основ обработки текстов, придавая своим документам чуть больше изящества.



Наш эксперт

Эндрю Грегори
Вечный новичок Эндрю занимается макиажем слов с 2003 года, а полочки Linux учиняет с 2005 года.



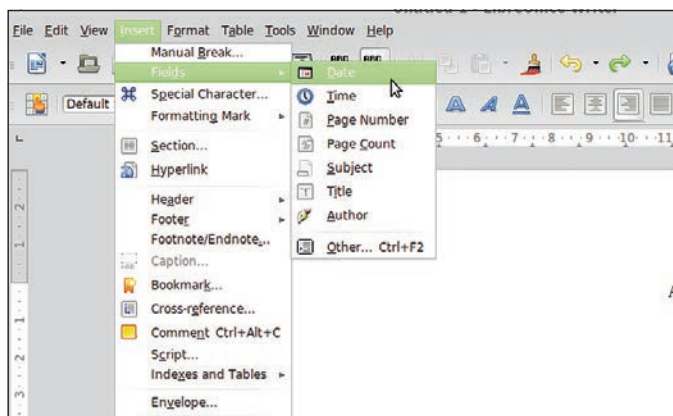
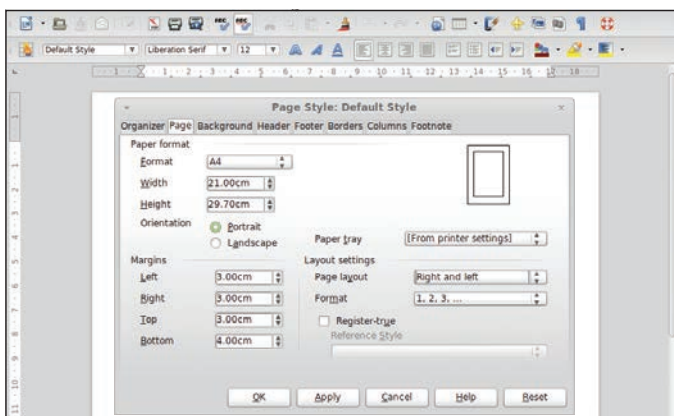
Офисный комплект *LibreOffice* содержит целую тучу функций, но для подсчета тех, какими мы пользуемся регулярно, хватит пальцев одной руки. Копирование и вставка, проверка орфографии, количество слов... вот, по-

жалуй, и все. И это позор, поскольку в *LibreOffice* гораздо больше возможностей сделать ваш документ элегантным. Речь здесь не о включении графических изображений: если требуется объединять картинки с текстом для создания броских оригинал-макетов, лучше применить специализированное приложение для верстки — *Scribus*.

Между тем, *LibreOffice* предлагает ряд полезных возможностей, облегчающих создание документа, который выглядит внятно и гармонично: это стили и шаблоны. Используя стили, не приходится терять время на форматирование каждого элемента, будь то сноски, заголовок или основной текст. Единоразово задав стиль, применяйте его там, где надо (та же схема используется в CSS при выборе стиля в HTML-документах). Выбранный стиль фиксируется в шаблонах, и оформление ваших работ не будет меняться со временем.

Обычно постепенная эволюция идет на пользу, но с точки зрения дизайна она враг единообразия. Согласованность писем, очерков, извещений и т.д. придает вашей документации профессиональный вид и подчеркивает содержание документов; а стало быть, в установленном *LibreOffice* стоит создать пару-тройку шаблонов для разных нужд.

Создаем шаблон



1 Зададим размер страницы

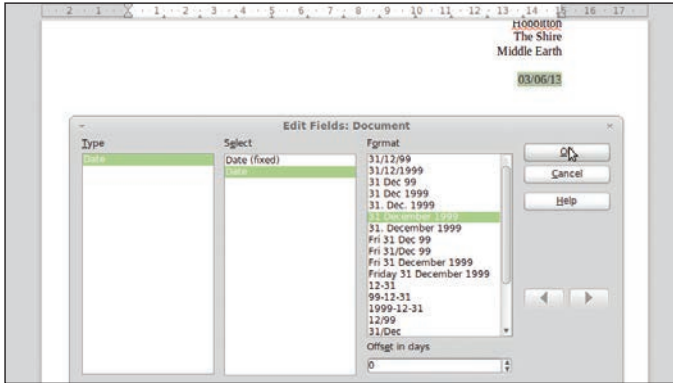
Для создания нового документа жмем File > New > Text Document [Файл > Создать > Текстовый документ]. Затем выбираем Format > Page [Формат > Страница] для вызова диалога создания стиля. A4 — хороший стандартный размер для страницы, но если вы создаете шаблон письма, поля по умолчанию слишком малы. Попробуйте установить поля по 3 см и отступ на 4 см снизу.

2 Добавим разные элементы

Добавьте имя и адрес и отформатируйте, как делается в письмах (обычно в правом верхнем углу), затем ниже проставьте дату. Так как наш урок посвящен экономии времени с помощью шаблонов, жмем Insert > Fields > Date [Вставка > Поля > Дата], чтобы дата загружалась автоматически.

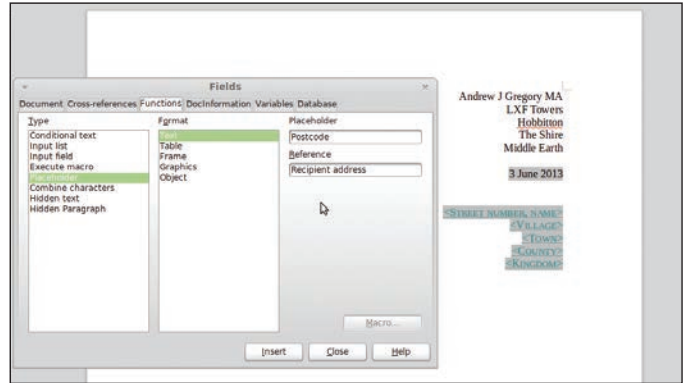
» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

К СТИЛЮ



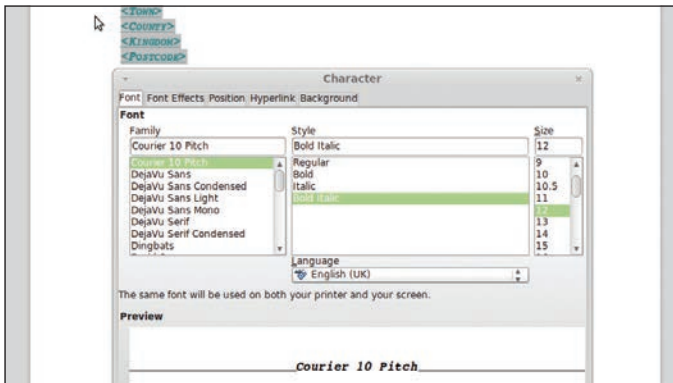
3 Наш первый стиль

Текущее отображение даты DD/MM/YY нам не нравится, поэтому ждем Edit > Fields [Правка > Поля]. Единственное доступное поле — Date (т. к. это все, что мы добавили), зато уж из имеющихся стилей есть что выбрать. Не выходя отсюда, выберите Date в графе Select [Выбрать]; в противном случае шаблон всегда будет показывать дату своего создания.



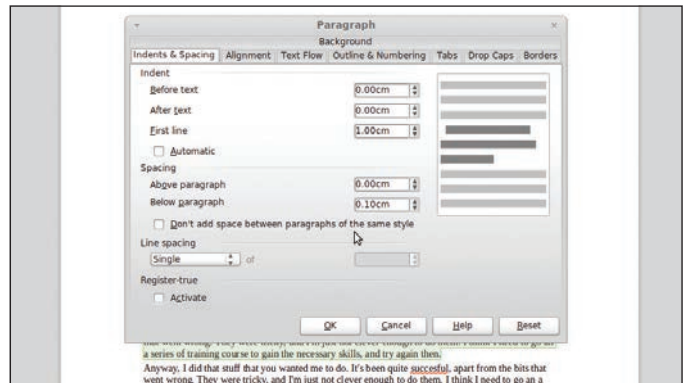
4 Добавление переменных данных

Ваши письма всегда идут от вашего имени, но адресованы разным людям, поэтому нельзя, да и ни к чему, включить в шаблон адрес получателя. Зато можно предусмотреть поле для его заполнения, с помощью Insert > Fields > Other > Functions > Placeholder [... > ... > Другое > Функции > Заполнитель]. Добавьте поля адреса одно за другим, каждый раз нажимая Insert [Вставить].



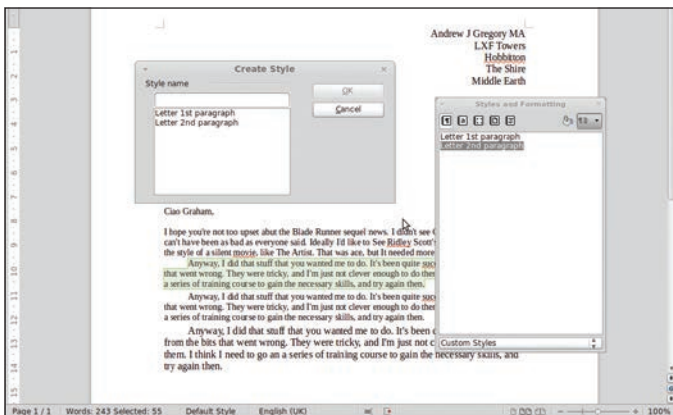
5 Выделения

Добавленным полям адреса можно задать любой стиль, на собственный вкус. Но чтобы вы не забыли заполнить предусмотренное поле, оно будет появляться в угловых скобках, выделенное другим цветом. Теперь пора создавать текст письма и задать ему желаемый стиль.



6 Стили абзаца

Как и в *Linux Format*, сделаем первый абзац прижатым к левому краю, а все последующие пусть будут с красной строки — т. е. с отступом. Для этого выделите второй абзац, перейдите в Format > Paragraph > Indents & Spacing [... > Абзац > Отступы & Интервалы]. Задайте первой строке [First Line] — отступ в 1 см. Мы еще прибавили 1 мм под абзацем, чтобы текст был попросторнее.



7 Сохраним стиль

Вызовите диалог Styles and Formatting [Стили и форматирование] (Format > Styles And Formatting), выделите стиль абзаца (когда будете удовлетворены шрифтом, кеглем, жирностью, отступом, интервалом и т. д.) и щелкните на иконке вверху справа. Выбираем Choose New Style From Selection [Создать новый стиль по выделенному фрагменту] и присвойте удобное имя (например, «Второй раздел письма»). Когда стиль абзаца задан, для присвоения стиля выделенному фрагменту требуется лишь дважды щелкнуть на названии стиля в диалоге Styles and Formatting (или нажать для этого F11).

И, наконец, идем в File > Save as Template [... > Сохранить как шаблон]. Единственный другой шаблон, установленный по умолчанию — MediaWiki; он выглядит как файл, но на самом деле это директория, которую нужно выбрать, прежде чем LibreOffice позволит выполнить сохранение.

Поздравляю с не напрасно прожитыми десятью минутами жизни! **LXF**

Raspbian: Как его

Установка другого дистрибутива из Linux — пожалуй, самый безопасный вариант, а попутно вы узнаете кое-что новое.

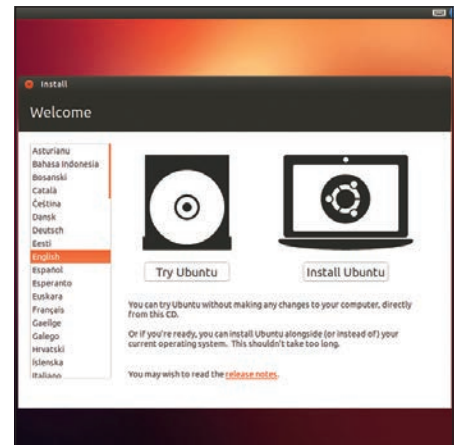
Шаг за шагом: Linux

1 Делимся функциями

Если вы никогда раньше не пользовались Linux, не волнуйтесь. Пользоваться им ныне так же просто, как и любой другой операционной системой, а во многих аспектах еще проще. Не нужно гоняться за драйверами, и новые приложения всегда устанавливаются через Linux-эквивалент App Store. И так как вы собираетесь устанавливать Linux и пользоваться им на Raspberry Pi, SD-карту стоит создавать в Linux-окружении. Это не влияет на установку, но дает вам отличную возможность проверить ее, прежде чем вставлять карту в Raspberry Pi. Мы советуем воспользоваться дистрибутивом Ubuntu, поскольку он идеален для начинающих, но наши инструкции подойдут почти ко всем

остальным дистрибутивам — просто замените Ubuntu Software Centre на соответствующий менеджер пакетов и не обращайте внимания на специфику рабочего стола. Linux — это еще и отказоустойчивый вариант, так как его можно запустить с Live CD, ничего не устанавливая. Вставьте компакт-диск и загрузите систему с оптического привода. Через несколько мгновений выберите из меню Try Ubuntu [Попробовать Ubuntu] вместо Install [Установить]. Загрузится рабочий стол Ubuntu, причем на компьютер ничего не устанавливается.

» Даже если Linux не установлен, можно загрузить рабочий стол Linux с помощью Live CD.

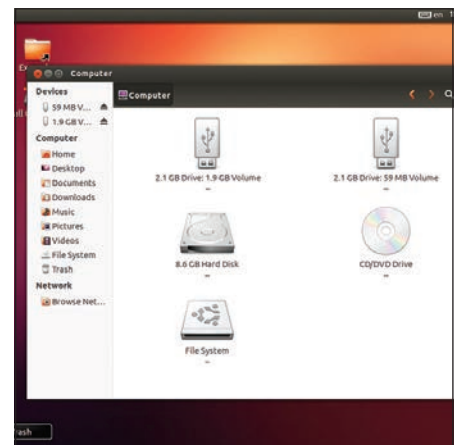


2 Будьте готовы

Проблема с установкой с помощью Live CD в том, что вы не сможете загрузить образ Raspbian. В сеансе рабочего стола для хранения данных выделяется недостаточно оперативной памяти, и образ придется откинуть на какое-нибудь внешнее устройство (но не на SD-карту для Raspberry Pi). Тем, у кого установлен Linux, беспокоиться об этом не нужно, и они могут просто загрузить последнюю версию образа на свой жесткий диск. После загрузки образа проверьте состояние SD-карты. Вставьте ее в слот читающего устройства, и она должна появиться на рабочем столе с отдельным окном для каждого раздела на диске. При установке Raspbian все данные на карте будут утеряны,

поэтому убедитесь, что на ней нет ничего, что вам хотелось бы сохранить. Так как Ubuntu загружает содержимое каждого раздела независимо от вида его форматирования, при необходимости можно проверить каждый свободный байт карты.

» Когда вы вставите флэшку, в Ubuntu будет показано содержимое всех найденных разделов.



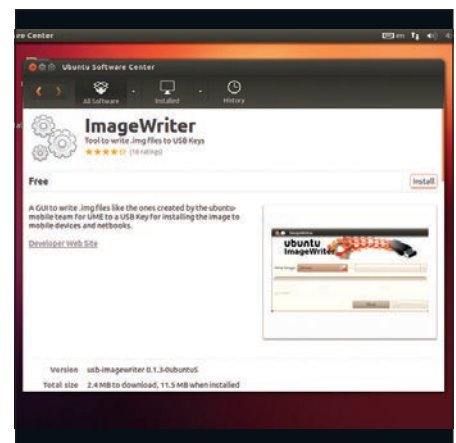
3 Установка ImageWriter

Для записи Raspbian мы воспользуемся графической утилитой *ImageWriter*. Ее можно установить из программы *Software Centre*, которая запускается щелчком по иконке с корзиной на панели запуска. В строке поиска наберите 'imagewriter'. Результат поиска должен быть один. Дважды щелкните по нему, и в следующем окне будет показано, что он доступен из источника 'universe'. Это дополнительный репозиторий программ, и по умолчанию он не включен, поэтому для доступа к нему нужно нажать кнопку Use This Source [Использовать этот источник]. Подождите, пока закончится обновление внутреннего списка пакетов, затем очистите строку поиска и снова

наберите 'imagewriter'. Вы увидите, что пакет обновлен, а когда выберете его, появится кнопка Install [Установить]. Нажмите на нее, и пакет будет загружен и установлен автоматически.

Вы можете спросить, почему же это сработало с Live CD. Ответ — доступной памяти достаточно для установки приличного числа пакетов, ее не хватит только на целый образ Raspbian. После установки *ImageWriter* и монтирования SD-карты все готово для записи образа Raspbian на SD-карту.

» *ImageWriter* можно установить и запустить с Live CD Ubuntu, а значит, вам даже не нужен ПК с Linux.



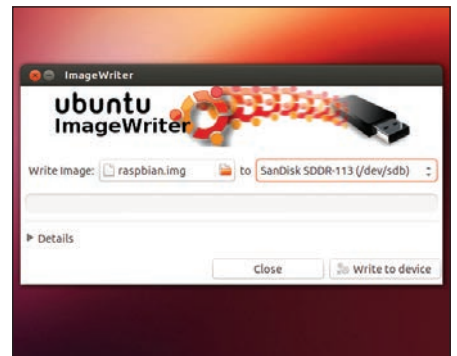
УСТАНОВИТЬ

4 Запись образа Raspbian

ImageWriter нужно запускать при подключенной SD-карте, иначе он не запустится и пожалуется, что не может найти ни одного диска. Когда появится главное окно программы, в нем нужно указать один-два параметра. Первый — путь до образа Raspbian, который нужно записать на флэшку, а второй — устройство, на которое нужно записать образ. Второй параметр важнее, потому что если подключено несколько устройств — например, если вы считываете образ с внешнего диска при его записи на SD-карту — то неверный выбор устройства приведет к тому, что данные на нем будут перезаписаны. И Windows, и OS X страдают от той же проблемы, но Linux по крайней мере

не позволит вам выбрать ничего кроме внешних USB-устройств хранения данных. Также вам покажут имя и производителя устройства, чтобы вы были уверены, что выбираете нужное. Выбрав устройство, нажмите кнопку Write to device [Записать на устройство]. Если вы выбрали верное устройство, светодиод активности на SD-карте должен замигать, сигнализируя, что данные записываются. Если нет, спешно закрывайте окно *ImageWriter*, чтобы прервать процесс.

Запись может затянуться, это зависит от карты и USB-портов. На наш образ ушло около 15 минут, но все это время индикатор записи информировал нас о ходе процесса; и когда запись закончится, можно проверять SD-карту.



► *ImageWriter* нужны только файл образа и место, куда его записать, но он не запустится без вставленной карты.

5 Проверка SD-карты

В отличие от Windows и OS X, в Linux можно прочитать оба раздела, созданные в процессе записи. Первый отформатирован с файловой системой FAT Windows, и его размер — почти 60 МБ. При загрузке Raspberry Pi с флэшки этот раздел считается первым, затем управление передается второму разделу. Второй раздел занимает львиную долю от 2 Гб и содержит корневую файловую систему Linux. Поскольку, когда вы в следующий раз вставите SD-карту в ПК с Ubuntu, оба эти раздела будут смонтированы, вы сможете повнимательнее рассмотреть находящиеся на них файлы. Раздел с Linux будет очень похож на настольную версию Ubuntu, которая у вас может быть запущена,

потому что и Ubuntu, и Raspbian происходят от одного «прародительского» дистрибутива Debian. Например, домашний каталог содержит собственный каталог пользователя, где он может хранить свои файлы и настройки. В Raspbian заранее настроен один пользователь — 'pi' (хотя в запущенном дистрибутиве это легко изменить), и поэтому, нажав на Home, вы увидите именно его каталог и файлы. Закончив, размонтируйте устройство в файловом менеджере и вставьте карту в Raspbian Pi.

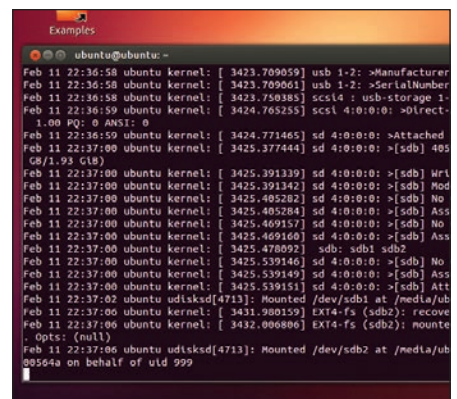


► Linux может читать файловые системы Windows, Linux и OS X, не требуя установки каких-либо дополнительных программ или драйверов.

6 Безаварийная установка

Расскажем о еще одном способе установки Raspbian на SD-карту, удобном как запасной вариант. Но при этом легко случайно перезаписать свои данные, и мы советуем прибегать к нему, только если все другие способы отказали. Мы воспользуемся командной строкой и командой **dd**. Она принимает «сырые данные» и копирует их — байт за байтом — на другое устройство. Стоит указать неверное устройство — и ваш жесткий диск с ценными фотографиями на нем будет затерт. Чтобы выбрать правильное устройство, сначала отключите SD-карту и найдите и запустите программу *Terminal* в Ubuntu. Откроется интерфейс знаменитой командной строки Linux, но в нем на самом

деле не все так сложно. Наберите **tail -f /var/log/syslog** и вставьте SD-карту. Эта команда выводит на экран системный журнал; найдите строку, напоминающую **sdb:sdb1**. Она означает, что система обнаружила новое устройство и присвоила ему файл устройства в файловой системе **sdb (sdb1 — первый раздел sdb)**. Должны быть и другие строки, так как компьютер пытается прочитать файловую систему и смонтировать его. Если она смонтирована, размонтируйте ее из графического интерфейса, а затем наберите **sudo dd bs=1M if=raspbian.img of=/dev/sdX**, заменив имя файла образа и имя устройства на **/dev/sdX** своими. Теперь образ будет записан на SD-карту без помощи графического интерфейса. LXF

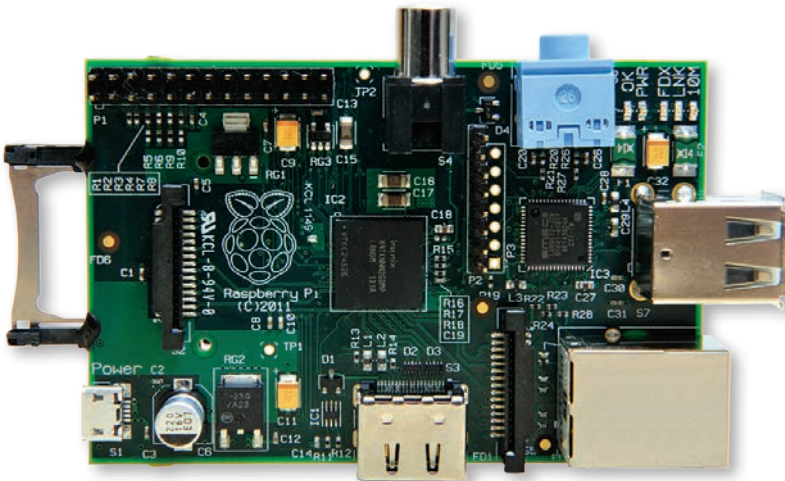


► Команду **dd** можно прервать, нажав **Ctrl+C**.



Raspberry Pi. Как

В девяти случаях из десяти это вам не понадобится — Pi просто загрузится. Но сюда стоит заглянуть, если что-то пойдет наперекосы.



Большое отличие Raspberry Pi от вашего компьютера в том, что в Pi нет BIOS. Именно BIOS вы видите первым при включении компьютера — старомодную картинку или текст с информацией о процессоре, памяти и жестких дисках, который исчезает слишком быстро, чтобы вы успели его прочесть. В случае проблем с компьютером, с помощью BIOS их часто можно устранить, а если память или процессор вышли из строя, благодаря BIOS компьютер начнет издавать звуки и мигать светодиодами. Не располагая возможностями BIOS, в Raspberry Pi решить такие проблемы не очень просто, и хотя в первый раз ваш Pi должен загрузиться без проблем, чем больше вы будете с ним играть, тем вероятнее, что однажды у вас возникнут проблемы с загрузкой. Их решению и посвящены эти две страницы.

Что означают светодиоды

Единственной подсказкой о том, как идет дело, станет яркий ряд светодиодов рядом со звуковым входом и USB-портами. На модели В пять светодиодов, а на модели А — всего два, но уцелели метки-названия остальных светодиодов.

- » **Светодиод 1 (LED1):** Зеленый с меткой ACT: доступ к SD-карте
- » **Светодиод 2 (LED2):** Красный с меткой PWR: напряжение 3,3 В присутствует
- » **Светодиод 3 (LED3):** Зеленый с меткой FDX: подключена полнодуплексная сеть (LAN)
- » **Светодиод 4 (LED4):** Зеленый с меткой LNK: связь/сетевая активность (LAN)
- » **Светодиод 5 (LED5):** Желтый с меткой 100: подключена сеть 100 Мбит (LAN)

Так как у модели А нет сетевого интерфейса, то трех последних светодиодов нет на ее материнской плате, и метки немного отличаются от более ранних версий модели В, хотя функции светодиодов идентичны.

При первом подключении Pi к источнику питания должен загореться красный LED2. Это значит, что на устройство подается

питание должным образом, и этот светодиод должен оставаться включенным в течение всего времени работы Raspberry Pi. Даже когда нет сетевого подключения или не подключена SD-карта, этот светодиод должен гореть. Если он мерцает или гаснет, то у вас есть проблемы с питанием устройства, и первым делом нужно проверить кабель и источник питания. Если SD-карта подключена, следующим должен загореться соседний светодиод, который сигнализирует, что происходит считывание данных со вставленной SD-карты.

Последовательность загрузки

Сначала этот светодиод загорится, затем погаснет; через некоторое время снова загорится и погаснет, по мере считывания кода загрузки с SD-карты. Если дело до этого не доходит, значит, проблема либо в коде ядра, который не был корректно записан на карту, либо в том, что ваша карта не работает с Pi. Проверьте, что карта вставлена до упора, а если используете адаптер для карт microSD, проверьте, что он подсоединен. Также стоит проверить, что контакты на Pi пружинят и выглядят одинаково, поскольку сообщалось о случаях проблем с этими контактами.

Кроме того, можно определить, на каком этапе загрузки процесс остановился. Вот список различных режимов мигания светодиода ACT/OK и того, что они означают — впрочем, этот список основан на прошивке как минимум середины 2012 года, и мы взяли его с форумов Raspberry Pi. На практике нам показалось, что проще игнорировать режимы мигания, если вы не готовы погрузиться в глубины процедуры загрузки, и хотя это, возможно, и весело, но может и обескуражить, если вы хотите просто поиграть со своим новым устройством.

- » **3 вспышки: loader.bin** не найден
- » **4 вспышки: loader.bin** не запущен
- » **5 вспышек: start.elf** не найден
- » **6 вспышек: start.elf** не запущен

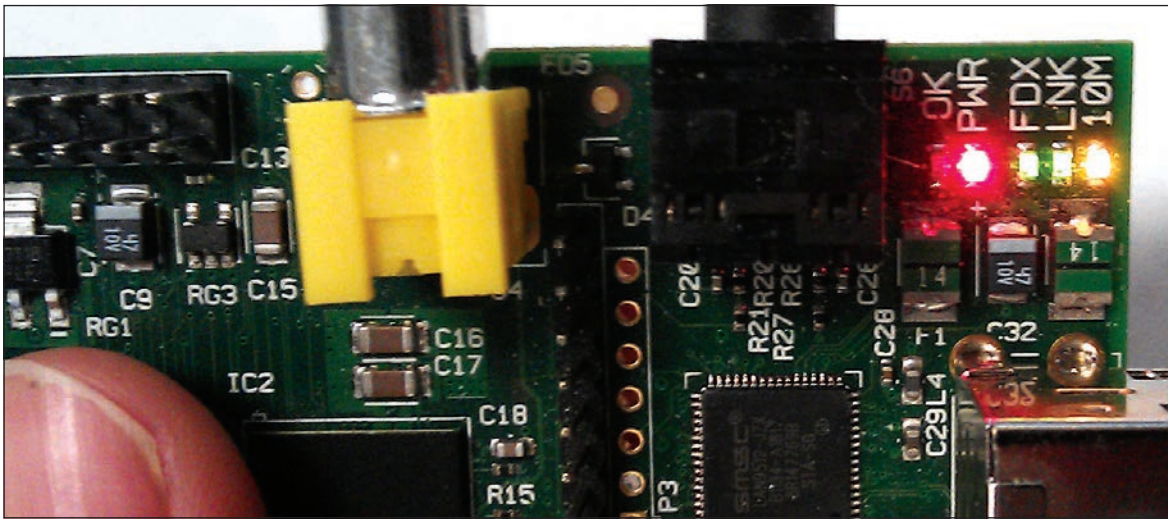
В октябре 2012 года прошивка обновилась, и мигания стали означать следующее:

- » **3 вспышки: start.elf** не найден
- » **4 вспышки: start.elf** не запущен
- » **7 вспышек: kernel.img** не найден

Проблемы можно исправить, найдя эти файлы и убедившись, что они в порядке. Проверьте их контрольную сумму и убедитесь, что контрольная сумма файлов на карте совпадает с контрольной суммой исходных файлов.

Но так как эти ошибки вероятнее всего означают то, что либо SD-карта не считывается правильно, либо операционная система на Raspberry Pi не записывается правильно, советуем взять новую SD-карту и записать образ другим способом. Можно даже взять у коллег другую карту и попробовать ее. Однако у нас были проблемы с несколькими кард-ридерами — обычно со встроенными в ноутбуки и нетбуки — и поэтому мы бы посоветовали вам сначала попробовать подключить внешний кард-ридер. Они дешевы, и нередко продаются в комплекте с самими картами. Если к Pi подключен монитор, в последних версиях также можно увидеть

ЛЕЧИТЬ ОШИБКИ



► Мигание светодиодов на Raspberry Pi гипнотизирует, но это также ключ ко множеству проблем, которые вы можете решить.

быстро меняющийся текст на экране загрузки. Если после этого текста все останавливается, проблема опять же в источнике питания.

Сеть

Если Raspberry Pi успешно инициировал начальную процедуру загрузки, операционная система начнет активно считывать содержимое SD-карты. До конца загрузки первый зеленый светодиод должен гореть почти постоянно, так как с SD-карты считывается много данных почти без пауз. Скорость считывания данных отчасти зависит от скорости вашей SD-карты. Почти сразу после начала загрузки Linux также должен мигнуть четвертый светодиод (LNK), а затем примерно на полсекунды загорятся остальные светодиоды, связанные с сетью. В это время инициируется сетевой стек и устанавливается связь с сетью Ethernet. Состояние этих светодиодов аналогично состоянию таких же светодиодов на любой сетевой карте Ethernet, и проблемы с ними означают скорее проблемы с сетью, чем с настройками Raspberry Pi.

Оранжевый светодиод означает полдуплексное подключение. Когда сети только появлялись, было важно, используется ли полдуплексное или полдуплексное подключение, потому что в первом случае данные можно было передавать только в одну сторону в один момент времени. Это, разумеется, влияет на скорость и эффективность, но ныне встречается относительно редко. Почти всегда этот светодиод будет гореть и обозначать полдуплексное подключение, при котором данные можно отправлять и получать одновременно. Так работают почти все современные сети.

Светодиод LNK — сетевой эквивалент светодиода ACT доступа к SD-карте, поэтому при передаче и приеме данных из сети он мигает точно так же, как при считывании и записи данных на карту. Если светодиод загорается, можно предположить, что соединение между устройством и коммутатором, управляющим

вашей сетью, установлено. Но это не обязательно означает, что вы можете подключиться к Интернету или проверить почту. С помощью этого светодиода можно следить за отправкой данных в сеть без активного участия Pi, но это лучший признак того, что сеть на Pi работает как полагается. Если этот светодиод мигает, но подключение не работает, проверьте настройки локальной сети, сначала убедившись, что Pi получает IP-адрес от роутера; а затем — настройки в самой операционной системе. Она может настраиваться автоматически почти в любых условиях, но иногда, если сеть немного необычная — например, у вас два домена или несколько роутеров — могут возникнуть проблемы. Единственный способ решения этих проблем — попробовать подключить Raspberry Pi к самой простой и наиболее видимой части сети и задать те настройки, которые вам нужны.

Последний светодиод обозначает скорость сети. Если он горит, то Pi установила скорость 100 Мбит/с. Это 1000000 бит данных в секунду или 100 мегабит, стандарт, также известный как Fast Ethernet (быстрый Ethernet). Это шаг вперед по сравнению с очень медленными 10 Мбит/с (светодиод не будет гореть), но немного медленнее самого быстрого на сегодня стандарта в 1000 Мбит/с, также известного как Gigabit Ethernet. Причина, по которой Raspberry Pi не поддерживает Gigabit Ethernet, в том, что порт Ethernet на самом деле подключен через шину USB 2.0, а в USB есть ограничение максимальной ширины канала, которое ограничивает потенциальную скорость порта до 100 Мбит/с. Однако этой скорости более чем достаточно для всевозможных ситуаций, так как всего несколько лет назад почти на всех устройствах было столько же. Вы заметите ухудшение производительности, только если превратите Raspberry Pi в высокопроизводительное хранилище NAS или захотите одновременно смотреть несколько фильмов с высоким разрешением. В во всех остальных случаях скорости Ethernet будет достаточно, и вы сможете продолжать наслаждаться своим Raspberry Pi. **LXF**

Админу: Пределы

Управляя отведением места на общем компьютере, **Нейл Ботвик** показывает, как задать квоты файловой системы для разных пользователей.



Наш эксперт

У **Нейла Ботвика** по компьютеру в каждой комнате, но по соображениям безопасности он ни за что не скажет вам, где находится центральный сервер.

Диски становятся все больше, но растут также и объем и количество файлов, которые мы на них храним. Помню, как покупал свой первый жесткий диск для Amiga 500. Я попросил диск на 20 МБ, но мне посоветовали модель объемом 42 МБ, сказав, что 20 МБ быстро заполнятся. До этого у меня были только дискеты по 800 КБ, и подобный совет выглядел полным абсурдом.

Если вы единственный пользователь компьютера, управлять дисковым пространством довольно легко: не храните слишком много и удаляйте все, что утратило ценность. С общим компьютером не все так просто. Если кто-то загружает или иным способом создает файлы в своих домашних каталогах, есть риск, что домашний раздел заполнится и вы не сможете сохранить ни одного файла, потому что другой пользователь захватил все пространство. Нам нужен способ ограничить пространство, доступное каждому пользователю. Можно бы создать отдельный раздел для домашнего каталога каждого пользователя, но это громоздкий и негибкий вариант. Что делать, если у какого-то пользователя избыток свободного места, а вам остро необходимо сохранить пару огромных файлов? Что делать, если надо добавить нового пользователя — захотите ли вы переразбивать диск? Но есть и более удобный способ — квоты файловой системы.

Что такое квоты?

Квота — просто-напросто ограничение дискового пространства, доступного конкретному пользователю в конкретной файловой системе. Существует два основных типа квот: квота на блоки ограничивает количество блоков файловой системы, предоставляемых пользователю, определяя доступное ему пространство, и называется также квотой на дисковое пространство. Квота на индексные дескрипторы [inode] ограничивает количество дескрипторов, фактически ограничивая количество файлов, которые может иметь пользователь, и называется также квотой на файлы.

Кроме того, квоты делятся на жесткие и мягкие. Мягкая квота — это предел, который можно превысить, возможно, на короткое время; но при этом обычно выполняется какое-нибудь действие, например, отправляется письмо с предупреждением. Жесткая квота абсолютна. Любая попытка сохранить данные, превышающие объем жесткой квоты, закончится неудачей.

Квота задается для файловой системы. Если у вас есть отдельный раздел `/home`, для него и нужно это делать. Если нет (как

Новое о квотах в ext4

Мы упоминали, что в XFS информация о квотах хранится в метаданных файловой системы, тогда как в остальных файловых системах используется `aquota.user` или `aquota.group`. В последних версиях ядра от 3.6 и выше поддерживается эта же возможность для ext4. С ней никаких изменений в файле `/etc/fstab` производить не нужно. Вместо этого включите квоты командой `tune2fs`:

```
tune2fs -Q usrquota,grpquota /dev/sda5
```

Можно задать квоту для пользователя или группы, или обе, как здесь, и квоты будут автоматически включены при монтировании файловой системы. По вине одной из ошибок файловая система может не монтироваться, если код квоты ядра собран в виде модуля. Эта ошибка исправлена в 3.8, и прежде чем пользоваться этой возможностью, лучше подождать, пока в вашем дистрибутиве появится данная версия ядра.

в конфигурации Ubuntu по умолчанию), установите квоты на корневой раздел `/`. Учтите, что благодаря правам доступа файлы каждого пользователя уже хранятся в их собственных домашних каталогах. Нам нужна лишь файловая система с этими каталогами.

Выполнение любой административной команды требует прав доступа `root`, поэтому либо зайдите в консоль от имени `root`, либо подставляйте `sudo` перед каждой командой. В наших примерах предполагается, что у нас есть отдельный раздел `home` на `/dev/sda5`. Приспособьте эти команды к своей системе.

Включение квот

Пусть у вас есть диск в 500 Гб и четыре пользователя. Если установить для каждого из них квоту в 100 Гб, то останется еще много места для файлов операционной системы, и файловая система не будет заполнена. Сначала нужно установить программу, которая — вы удивитесь — сидит в пакете под названием `quota`. Затем откройте `/etc/fstab` и добавьте `usrquota` к параметрам файловой системы с разделом `/home` — у вас должно получиться нечто вроде

```
/dev/sda5 /home ext4 usrquota,noatime 0 0
```

Если у вас ext3 или ext4, удобно воспользоваться журналируемыми квотами, которые избавляют от необходимости проверять квоты после падения системы, добавив параметр `usrjquota=aquota.user` таким образом:

```
/dev/sda5 /home ext4 usrjquota=aquota.user.noatime 0 0
```

Теперь перемонтируйте файловую систему с новыми параметрами, перезагрузив систему или выполнив команду

```
mount /home -o remount
```

Создайте файл базы данных, необходимый для квот, и включите квоты командами

```
quotacheck -avu
```

```
quotaon -av
```

Выставляем лимиты

В пакете `quotas` должен быть скрипт, который запускает эти команды при загрузке, и вручную их придется запустить только

Групповые квоты

Все, что мы пока делали, задавалось на уровне пользователей, но квоты можно задать и для групп. Воспользуйтесь командой `grpquota` вместо или наряду с предыдущей командой в `/etc/fstab` и замените `-u` на `-g` во всех предыдущих командах. Групповая квота применяется к суммарному

дисковому пространству всех пользователей в группе. Если вы хотите выделить всем пользователям группы одинаковый объем дискового пространства, можно задать его для одного из них и затем скопировать настройки другим членам группы с помощью `setquota -p`.

ДОЗВОЛЕННОГО

Превышение квот

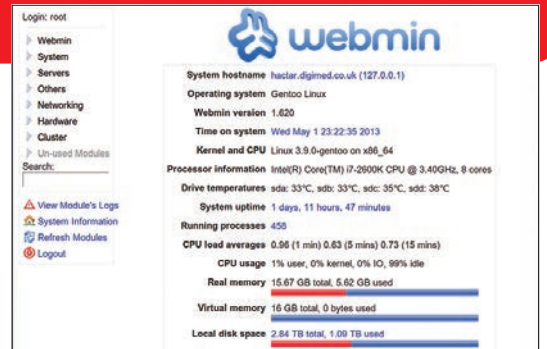
Когда пользователи попытаются записать файл с превышением квоты, они должны получить сообщение об ошибке, но также можно запустить **warnquota** в качестве задания *cron*. Настройки этой программы задаются в **/etc/warnquota.conf**, где нужно изменить хотя бы почтовую программу и отправителя, и она отправляет письмо каждому пользователю, который превысил свой мягкий предел. Она запускается в формате **warnquota /path**,

и в **/etc/cron.d** нужно поместить нечто вроде следующего:

```
0 */3 * * * warnquota /home
```

Эта команда будет проверять квоты каждые три часа и отправлять письма с оповещениями.

➤ Обычно квоты задаются командной строкой, но те, кто не хочет выпускать из руки мыш, могут обратиться к **Webmin**.



один раз. Если у вас ядро индивидуальной сборки, убедитесь, что в нем включена поддержка квот. В ядрах обычных дистрибутивов она уже будет включена. Теперь можно позадавать квоты.

Для создания и изменения квот используются две команды — **setquota** и **edquota**. Для первой параметры указываются в командной строке, а вторая загружает текстовое представление текущих настроек (или настроек по умолчанию, если никаких настроек не задано) в текстовый редактор по умолчанию (или *vi*). Квота пользователя задается командой

```
setquota -u arthur 100000000 110000000 0 0 /home
```

Первый аргумент — имя пользователя. Следующие два параметра — мягкие и жесткие лимиты дискового пространства, указанные в блоках по 1 кбит; еще два — то же самое для индексных дескрипторов. Последний аргумент — файловая система, к которой применяются ограничения, и ее можно указать в виде либо файла устройства (например, **/dev/sda5**), либо точки монтирования. Указание нуля для любого параметра отключает его. В данном случае мы задаем мягкое ограничение в 100 ГБ и жесткое в 110 ГБ, без ограничения на количество файлов. Результаты ее запуска покажет команда

```
quota -u arthur
```

Если добавить параметр **-s**, размер будет показан в обычных единицах, а не в количестве блоков.

```
quota -su arthur
```

```
Disk quotas for user arthur (uid 1001):
```

Filesystem	space	quota	limit	grace	files	quota	limit	grace
/dev/sda5	16014M	97657M	105G		18032	0	0	

Здесь мы видим, что у пользователя **arthur** занято около 16 МБ пространства с мягкой квотой в 97 МБ и жесткой — 105 МБ. Обратите внимание, что квоты задаются в блоках по 1 кбит, а в результатах с параметром **-s** выводятся «двоичные» размеры. Чтобы задать квоту в 100 МБ, придется доставать калькулятор.

Проверка квот

Эта команда сообщает информацию о квотах одного пользователя во всех файловых системах, где у него есть квоты; впрочем, для обычных пользователей это скорее всего только файловая система **home**. Для просмотра квот всех пользователей конкретной файловой системы используется команда **repquota**:

```
repquota /home
```

А для просмотра всех пользователей, у которых есть квоты, для всех файловых систем используется команда

```
repquota -a
```

Мы задавали квоты командой **setquota**, и ею же можно изменить их, указав новые параметры; но для изменения квот есть и другая команда — **edquota**. В большинстве файловых систем сведения о пользовательских квотах хранятся в файле **aquota.user** в корне файловой системы. В XFS они хранятся в метаданных файловой системы, поэтому обычного файла, где их можно было бы посмотреть, нет. Это двоичный файл, но **edquota** преобразует информацию в текстовый формат и загружает в текстовый редактор по умолчанию (или в *vi*, если не задана переменная окружения редактора). При сохранении данных они снова преобразуются обратно в двоичный формат и сохраняются в правильном месте.

```
edquota -f /home -u arthur
```

или

```
edquota -u arthur
```

Первая команда изменяет квоты для **arthur** в **/home**; вторая — для всех файловых систем с включенными квотами. Данные похожи на вывод команды **quota**. Можно изменять только поля со значениями пределов, но не с количеством используемых блоков или файлов. При удалении строки эта квота удаляется, что также можно сделать командой

```
setquota -u arthur 0 0 0 0 /home
```

Это происходит потому, что при установке любой квоты в ноль она отключается.

Несколько пользователей

До сих пор мы занимались квотами для одного пользователя, и не слишком трудно повторить все те же команды для другого; ну, а если их много? Что если вы несколько раз меняли настройки для одного пользователя и хотите, чтобы у остальных были такие же? И у **setquota**, и у **edquota** есть параметр **-p** или **--prototype** для копирования настроек другого пользователя. Установив квоты для **arthur**, можно скопировать их пользователям **bob**, **colin** и **dan** таким образом:

```
for i in bob colin dan; do
```

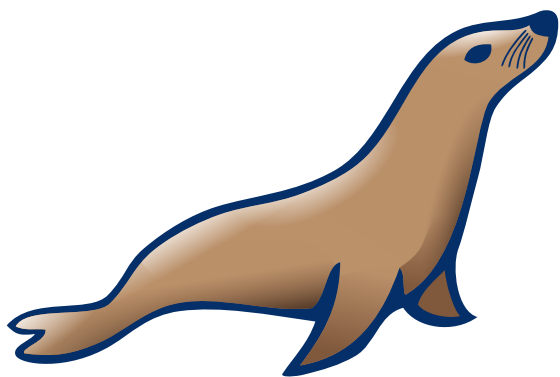
```
setquota -p arthur -u $i /home
```

```
done
```

Если вы предпочитаете **edquota**, с ней можно использовать параметр **-p**, но придется сохранять каждый файл перед переходом к предыдущему, и ею удобнее будет пользоваться, когда у всех пользователей одни и те же базовые настройки с некоторыми вариациями. LXF

MariaDB: База

Джонатан Робертс распределяет базу данных *MySQL* между компьютерами, повышая скорость и безопасность на радость пользователям вашего web-приложения.



MariaDB

» Тюлень Монти — символ *MariaDB*, ответвления *MySQL*, разработанный создателями *MySQL* до его захвата Oracle.



Наш эксперт

Джонатан Робертс сбежал из Башен Linux Format, чтобы поискать счастья в качестве сисадмина.

Web-приложениями ныне пользуются тысячи пользователей, и всем подавая доступ к данным всегда и мигмом, поэтому разработчики и администраторы баз данных SQL вынуждены внедрять в жизнь новые технологии. Для *MySQL* последняя из этих технологий — репликация базы данных. В этом случае одна «главная база данных» передает все изменения в своих данных на несколько «подчиненных баз данных», которые чаще всего находятся на других компьютерах.

В этом руководстве мы познакомим вас с преимуществами репликации баз данных и покажем, как приступить к этому самим.

Расщепление баз данных

Как мы упоминали, говоря о наращивании, подчиненные копии пригодны только для разделения операций чтения, а не записи, хотя можно создать систему репликации «главная копия — главная копия», теоретически поддерживающую запись на несколько компьютеров. Однако это плохая идея. Это способно создать так называемое «расщепление мозга», когда на серверах окажутся разные версии одних и тех же данных, что приводит к повреждению данных и чего следует избегать любой ценой.

Более того, при репликации записи вы не получаете никаких преимуществ. Каждую операцию записи нужно повторить на всех подчиненных копиях, и на одну опе-

рацию записи в главную копию потребуется дополнительная операция записи для каждой подчиненной.

Если вам нужно распределить операции записи для своего приложения, один из самых распространенных подходов — «расщепление» данных. Это означает создание нескольких баз данных, каждая из которых отвечает за свою часть данных, и распределение операций записи между ними.

Базы данных часто разделяются по учетным записям пользователей (например, первый миллион пользователей — в одной базе данных, следующий миллион — в другой), по географии (например, одна база данных обслуживает Великобританию, вто-

рая — Индию) и по многим другим критериям, в зависимости от задач конкретного приложения.

Поддержка расщепления должна быть реализована в вашем приложении, и вам нужны средства, гарантирующие, что запросы отправляются в нужную базу данных. Например, можно создать индексную таблицу в общей базе данных, которая показывает, какой пользователь принадлежит к какой базе данных, чтобы, когда приложению понадобятся его данные, приложение знало, к какой базе данных подключаться. Как и указатель в книге, индексная таблица будет гораздо меньше основной базы данных, и запросы в ней выполняются гораздо быстрее.

Мы будем пользоваться *MariaDB* — заменой *MySQL*, которая набирает популярность во многих дистрибутивах.

Наращивание против распределения

Сначала отметим преимущества, которые дает репликация; и начнем с масштаба. Предположим, вы написали совершенно революционное web-приложение. В первый день у вас 10 пользователей, во второй — 20, затем 40, 80, а потом о вашем приложении рассказывают на *TechCrunch*. На шестой день на вас внезапно наваливаются десятки тысяч пользователей, жаждущих одновременно воспользоваться вашим приложением.

Вы, естественно, писали приложение с оглядкой на «web-масштабирование», с *Node.js* и *Nginx*, так что ваши клиентские серверы чувствуют себя прекрасно. Но сервер баз данных уже пыхтит от натуги, время загрузки страницы взлетает ввысь, и пользователи начинают уходить, не дожидаясь, пока вы все исправите.

У этой жизненно важной проблемы есть два решения: наращивание ваших серверов или их распределение. Наращивание означает покупку более быстрого и дорогого оборудования, на котором будет работать база данных. Это легко и не требует никаких сложных действий, но все имеет предел. А если «железо» у вас — быстрее некуда, но пользователей все прибывает?

Тут пора задуматься о распределении работ. Это означает не покупку нового более быстрого сервера, а покупку нескольких серверов и распределение нагрузки между ними. Добавьте второй сервер — и вы удвоите число операций чтения, добавьте третий — утроите, и т.д. Если у вашего приложения много операций чтения, то можно воспользоваться именно этой технологией, которую позволяет репликация. Поддерживая несколько серверов синхронизированными с главной копией, можно распределить нагрузку с помощью балансировщика нагрузки или других технологий.

ДАнных НА ВЫНОС

Второе преимущество, которое дает репликация — высокая доступность данных. То есть, в случае аппаратной или другой ошибки репликация дает быстрый способ вернуть приложение в работу — возможно, даже раньше, чем пользователи заметят проблему.

Преимущества

Представим себе следующий сценарий: ваше web-приложение хранит в базе данных информацию о любимых породах кошек пользователей. Оно должно иметь возможность записывать в базу новую информацию и считывать оттуда старую. Если на сервере базы данных произойдет сбой, приложение перестанет работать, и пользователи отправятся куда-нибудь еще — например, перекинутся на фотографии морских свинок.

Конечно, вы человек ответственный и регулярно делаете резервные копии базы данных; но если упал весь сервер, то долгий простой вам все равно обеспечен. Нужно найти новый сервер, установить операционную систему и ПО базы данных, настроить его и затем восстановить базу из резервной копии. Этот последний шаг для больших баз данных может отнять немало времени — вплоть до нескольких часов.

А вот пользуясь репликацией, вы бы перенаправили трафик на подчиненный сервер. На это ушло бы всего несколько минут (или еще меньше, если процесс автоматизирован), и приложение возобновило бы работу почти сразу после сбоя.

Кроме того, наличие подчиненного сервера с последней версией базы данных позволяет делать точные резервные копии, не влияя на производительность и доступность вашего web-приложения, а также устанавливать и тестировать обновления без риска повредить актуальные данные.

Как работает репликация?

Репликация в семействе баз данных MySQL (MySQL от Oracle, MariaDB, Percona и т.д.) выполняется довольно просто. На главном сервере нужно включить Binary Log [двоичный журнал]. Он записывает все изменения в базе данных на главном сервере, т.е. все операции вставки [insert], обновления [update], изменения структуры данных [alter] и т.д., игнорируя только операции выбора [select].

Сервер копии считывает этот двоичный журнал и копирует записи в свой «журнал-ретранслятор [relay-log]», с помощью «нити ввода/вывода». Затем он считывает информацию из ретранслятора и выполняет записанные в нем действия по отношению к своей копии данных (с помощью «нити SQL»). Так данные подчиненного сервера синхронизируются с данными главного.

Важно отметить, что этот подход к репликации не является синхронным. Другими словами, главный сервер не ждет, пока подчиненный закончит обновление данных, и может вносить новые изменения в данные; таким образом, подчиненный сервер может немного «отставать» от главного. Если подчиненный сервер используется для операций чтения, пользователи могут увидеть не совсем актуальные данные, а при переключении подчиненного компьютера в режим главного можно потерять часть транзакций.

Двоичные журналы и резервные копии

При включении двоичного журнала производительность сервера немного ухудшается: ведь каждая операция по изменению базы данных будет не только выполняться над базой данных, но и регистрироваться в журнале. Несмотря на это, журнал может пригодиться не только для репликации подчиненных компьютеров, и его можно включить на всех серверах (как реплицированных, так и автономных), поскольку с его помощью можно реализовать выборочное резервное копирование.

Полную резервную копию базы данных обычно не делают чаще одного раза в день; ну, а вдруг база данных упадет через 23 часа после последнего резервного копирования? Это почти целый день обновлений, который будет потерян, если вы не включили двоичный журнал.

Задайте скрипту резервного копирования координаты двоичных журналов (файл главного журнала и текущее смещение

в этом файле) при создании резервной копии. Это можно сделать командами **show binary logs** и **show master status**.

При падении сервера первым делом нужно восстановить полную резервную копию. После этого можно применить содержимое двоичных журналов к восстановленной резервной копии, и все запросы, выполненные с момента последнего копирования до момента падения, повторятся снова. По завершении этого процесса у вас появится полноценная восстановленная копия базы данных.

Для восстановления данных с помощью записей в двоичных журналах понадобится утилита *mysqlbinlog*:

```
mysqlbinlog --start-position <masterlog-pos> <master-log-file> | mysql -uroot -p
```

Если вы хотите сперва только узнать, какие запросы будут выполняться, перенаправьте вывод команды не в MySQL, а в файл, и просмотрите его содержимое.

Избежать этого нельзя, но, тщательно поработав над структурой web-приложения, влияние этой проблемы можно снизить.

Теперь, разобравшись, что такое репликация и почему она бывает очень полезной, и в общих чертах представляя, как она работает, посмотрим, как настроить репликацию между двумя серверами *MariaDB*. В этой части учебника вам нужен доступ к двум компьютерам, находящимся в одной и той же локальной подсети. В их настройке вам поможет пошаговое руководство, приведенное ниже.

Начинаем репликацию

Настроив компьютеры, начинайте репликацию между ними. Она состоит из трех этапов:

- 1 Создание пользователей репликации на каждом компьютере.
- 2 Настройка каждого компьютера для репликации.
- 3 Запуск процесса репликации на подчиненном компьютере.

На первом этапе нужно создать на каждом компьютере пользователя MySQL, у которого есть права, требуемые для репликации. Как и при добавлении любого другого пользователя, для этого нужно зайти в консоль MySQL и воспользоваться командой **grant**:

```
mysql -uroot -p
grant replication slave, replication client on *.* to
test@'192.168.69.%' identified by 'password';
```

Команда **grant** назначает права доступа пользователям MySQL, но если вы пытаетесь назначить права пользователю, которого не существует, она также создает его.

»

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

```

jon@linux-kv5z.site:~
File Edit View Search Terminal Tabs Help

jon@linux-kv5z.site:~
jon@linux-kv5z.site:~
# Don't listen on a TCP/IP port at all. This can be a security enhancement,
# if all processes that need to connect to mysqld run on the same host.
# All interaction with mysqld must be made via Unix sockets or named pipes.
# Note that using this option without enabling named pipes on Windows
# (via the "enable-named-pipe" option) will render mysqld useless!
#
#skip-networking

# Replication Master Server (default)
# binary logging is required for replication
log-bin=mysql-bin

# binary logging format - mixed recommended
binlog_format=mixed

# required unique id between 1 and 2^32 - 1
# defaults to 1 if master-host is not set
# but will not function as a master if omitted
server-id      = 1

# Replication Slave (comment out master section to use this)

```

► При репликации очень важно задать верные настройки в файле `my.cnf`. К счастью, при работе с *MariaDB* в *openSUSE* вы увидите много комментариев себе в помощь.

В данном случае мы назначили права 'replication slave' и 'replication client' для всех баз данных и всех таблиц (*.*) пользователю 'test', который идентифицируется паролем 'password'. Причем данному пользователю мы разрешили входить в систему только с компьютеров локальной сети — `repl@192.168.69.%`: у него есть права доступа ко всем данным в базе (через двоичные журналы), так что это разумная мера предосторожности.

Перенастройка сервера MySQL

Затем нужно изменить кое-какие настройки сервера *MySQL*, чтобы включить репликацию. На главном сервере (у нас это `repl-demo.test`) обеспечьте задание в `/etc/my.cnf` следующих параметров:

```

log_bin = mysql-bin
server_id = 2

```

На подчиненном сервере задайте такие параметры:

```

log_bin = mysql-bin
log_slave_updates = 1
server_id = 3
read_only = 1

```

Первый параметр, `log_bin`, запускает двоичный журнал, который, как мы пояснили ранее, записывает все изменения в данных сервера, чтобы подчиненный сервер копировал их. Второй параметр задает уникальный идентификатор сервера. Все компьютеры в цепочке репликации должны иметь разные `server_id`; по умолчанию он равен 1, и лучше явно задать что-то другое, чтобы случайно не запустить два сервера с одинаковыми идентификаторами.

Для подчиненного сервера есть два дополнительных параметра: `log_slave_updates` велит подчиненному компьютеру добавлять события репликации в его собственный двоичный журнал. Это означает, что при необходимости подчиненный сервер может выступать в качестве главного сервера для другого подчиненного сервера. Такие цепочки репликации удобны в некоторых сценариях, а включение обновления по умолчанию может упростить переключение подчиненного сервера в режим главного в случае сбоя.

Наконец, мы устанавливаем опцию `read_only` на подчиненном сервере. В вашей конфигурации это может быть необязательно, но это хороший способ помешать неправильно настроенному приложению записать данные на подчиненный сервер, приведя к несоответствию данных на главном и подчиненном серверах.

Изменив файл `/etc/my.cnf` на обоих серверах, перезапустите сервис *MySQL* с помощью любой подходящей утилиты в своем дистрибутиве, например, *Systemctl*.

С главного на...

Ну вот, мы, наконец, готовы запустить репликацию на подчиненном сервере. На главном сервере запустите команду `show master status \G` и запишите имя файла и смещение внутри файла. Затем на подчиненном сервере запустите

```

change master to
master_host = '192.168.69.1',
master_user = 'test',
master_password = 'password',
master_log_file = 'mysql-bin.00004',
master_log_pos = 245;

```

Не забудьте указать записанные имя файла и смещение в параметрах `master_log_file` и `master_log_pos`. Затем запустите команды `start slave` и `show slave status \G`, чтобы посмотреть, работает ли репликация. Если все хорошо, вы увидите параметры `slave_sql_running = yes` и `slave_io_running = yes`.

Работу репликации можно легко проверить, выполнив пару команд на главном сервере и посмотрев, реплицируются ли их результаты на подчиненном. Например, создайте новую базу данных на главном сервере:

```

create database lxf;

```

Потом выполните команду `show databases`; на подчиненном сервере, и там должна появиться созданная база данных. Легко!

Репликация готовой базы данных

В примере выше мы начали репликацию с двух пустых баз данных. Если нужно реплицировать существующую базу с данными, понадобится ряд дополнительных шагов.

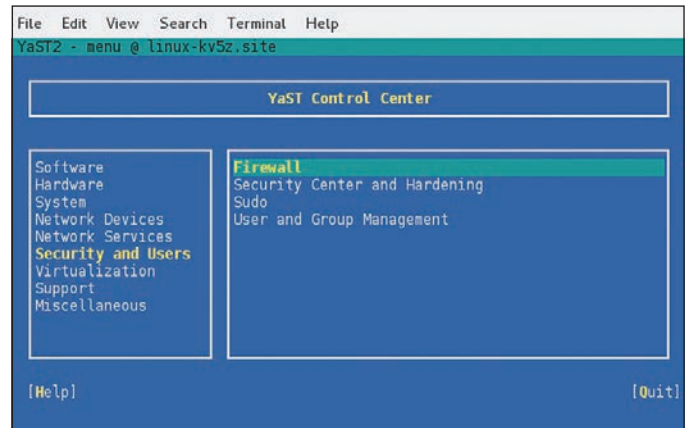
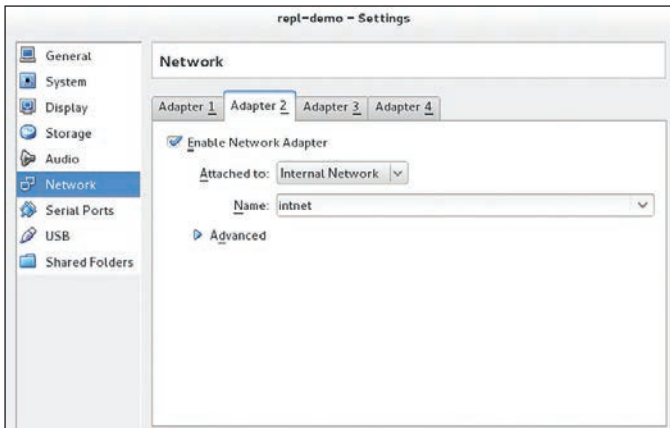
Во-первых, позаботьтесь, чтобы в конфигурационном файле `my.cnf` главного сервера были настройки, приведенные выше. Если их нет, добавьте их и перезапустите сервер. Затем приостановите запись новых данных в таблицы на главном сервере. Это можно сделать разными способами; проще всего остановить процесс *MySQL* с помощью *Systemctl* или другой похожей команды, а также выполнив команду `flush tables with read lock`; в консоли *MySQL*.

После этого спишите из результатов команды `show master status \G` имя и смещение лог-файла на главный сервер, как мы описывали выше, затем создайте копию базы данных главного сервера. После создания копии можно перезапустить *MySQL* или снять блокировку таблиц.

Самое последнее, что нужно сделать — поместить эту копию данных в каталог данных нового подчиненного сервера, перезапустить процесс сервера *MySQL* и выполнить команду `change master to` как ранее. Тогда новый подчиненный сервер будет готов реплицировать любые изменения главного.

Сложность здесь состоит в получении копии — это может отнять много времени, в течение которого приложение будет простаивать. Лучшее решение — запустить базу данных на LVM или на файловой системе вроде `btrfs` или `zfs`. Затем можно будет сделать снимок каталога с данными, что в большинстве случаев занимает меньше секунды, и возобновить работу сервера базы данных и приложения. И вы сможете скопировать данные из снимка без дальнейшего прерывания работы. А зачем останавливать сервер при создании копии? Дело в том, что подчиненный сервер должен начать репликацию точно с момента последнего обновления базы данных, представленного лог-файлом и смещением. При каждом новом обновлении смещение меняется; и, не заблокировав таблицы, вы не получите точных координат копии в лог-файле.

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

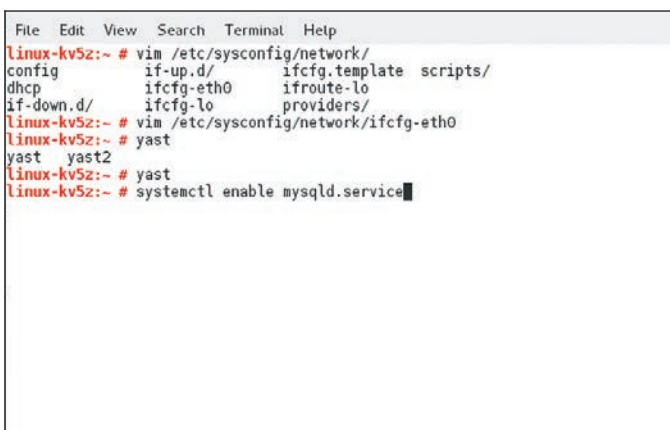


1 Установка VM в VirtualBox

Установите виртуальную машину с выбранным дистрибутивом в *VirtualBox*. Выберите Settings [Настройки] и настройте два сетевых интерфейса: 1) NAT; 2) внутренняя сеть. Настройте интерфейс eth0 (первый из двух) для работы с DHCP. Так у вас появится доступ к Интернет.

2 Настройка доступа в Интернет

Второй интерфейс (eth1) должен иметь IP-адрес **192.168.69.1/24**; оба интерфейса требуется запускать при загрузке системы. Задайте имя хоста **repl-demo.test** (необязательно). Также убедитесь, что брандмауэр разрешает доступ к порту 3306 локальной сети.

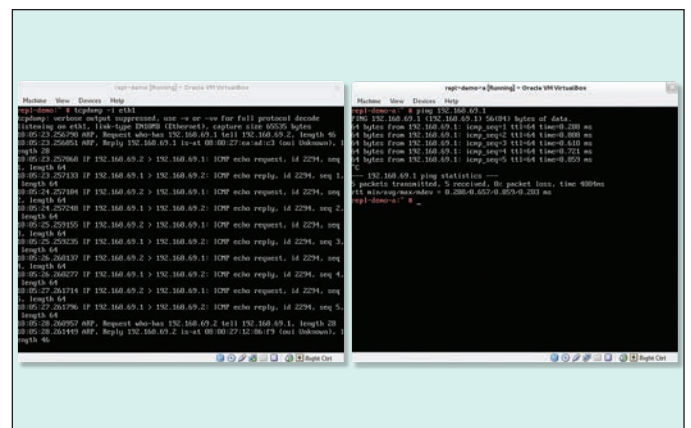
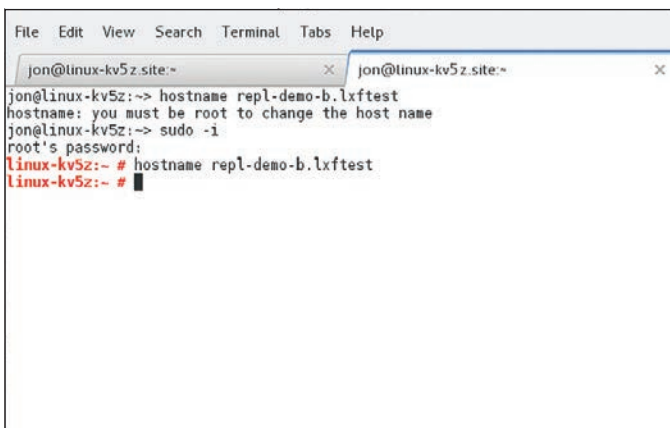


3 Установка MariaDB

С помощью менеджера пакетов дистрибутива установите *MariaDB* и *Mariadb-client* (или эквивалентные пакеты). Настройте запуск *MariaDB* при загрузке системы с помощью *Chkconfig*, *Systemctl* или аналогичной утилиты.

4 Клонирование VM

Выключите виртуальную машину, затем правой кнопкой щелкните на ней в *VirtualBox* и клонируйте ее, назвав новую виртуальную машину **repl-demo-a**. Обязательно выберите опцию **reset MAC addresses option** [сбросить MAC-адреса].



5 Перезагрузка и перезапуск сети

Теперь требуется загрузить виртуальную машину и задать для этой машины новое имя хоста, изменив адрес интерфейса eth1 на **192.168.69.2/24**. После чего перезапустите сеть.

6 Перезагрузка и проверка

Чтобы проверить, что все работает правильно, снова загрузите первую виртуальную машину и убедитесь, что компьютеры могут связаться друг с другом, попинговав с каждого из них IP-адрес другого. **LXF**

CUDA: ДАННЫЕ

Бен Эверард показывает, как предоставить процессору заслуженный отпуск, заставив перемалывать числа... видеокарту.



Наш эксперт

Бен Эверард оставил работу IT-консультанта и на два года отправился в Танзанию устанавливать в местных школах системы на базе Ubuntu. Теперь его знания находят применение в бурлящем котле открытий — Башнях Linux Format.

Компьютеры ускоряются. Современные многоядерные процессоры умеют жонглировать байтами с феноменальной быстротой. Но кое-что растет еще быстрее скорости наших компьютеров, и это — желание перелопатить еще больше данных. Пока процессоры пытаются раскидать накапливающиеся на очереди горы цифр, люди ищут новые способы их обработки.

К решению этой задачи все чаще привлекается графический процессор — GPU (также известный как видеокарта). У многих из нас в компьютерах есть эти мощные вычислительные модули, но мы пользуемся ими лишь изредка, когда играем. Пора выпустить на волю их первозданную вычислительную мощь и найти ей лучшее применение.

Видеокарты и дополнительные ядра в многоядерном процессоре — не одно и то же. На видеокартах нельзя просто запускать обычные потоки-нити [thread] и программы, поскольку они устроены иначе, чем обычный процессор CPU. CPU приходится иметь дело с большими объемами данных самых разных типов, поэтому в нем небольшое количество довольно крупных и сложных обрабатываемых ядер. В отличие от него, графические процессоры получают на обработку гораздо более простые данные, и в них, наоборот, много ядер помельче. В карте Nvidia GTX 680, которой мы пользовались для целей данного урока, 1536 ядер, а в процессоре Intel i5 3570K на том же компьютере — всего 4.

Понятное дело, это означает, что графические процессоры лучше работают с простыми операциями на больших объемах данных (для компьютера, «простые» значит «арифметические»). Самая популярная область применения таких операций — взлом паролей.

Прежде чем двигаться дальше, поясним различие между CUDA и OpenCL. Обе платформы решают примерно одинаковые задачи (заставляют графический процессор выполнять вычисления, не связанные с графикой), но применяют разные подходы. Самое большое отличие в том, что они работают на разных видеокартах. CUDA запускается только на картах Nvidia, а OpenCL в большей степени независима от аппаратуры, хотя в основном

используется на AMD/ATI. Однако CUDA получила гораздо большую поддержку в промышленности, поэтому воспользуемся ею. Те же базовые технологии работают и в OpenCL, так что, пойдя по этому пути, вы все равно сможете воспользоваться идеями, о которых мы здесь рассказываем.

Вернемся к взлому паролей. Пароли хранятся в виде результата свертки. Свертка — однонаправленная математическая функция, которая применяется к паролю. Она называется однонаправленной, поскольку необратима. То есть, имея пароль, легко вычислить свертку, но если в наличии только свертка, вычислить пароль (почти) невозможно. Это отличается от обычного шифрования, так как секретного ключа для расшифровки нет. Не то что злоумышленник, а и законнейший пользователь не сможет воссоздать по свертке исходный пароль. Кстати, именно поэтому созданный ранее пароль нельзя «восстановить» (независимо от схемы аутентификации), а можно только сбросить. Если какой-либо сайт может напомнить вам пароль, это означает, что на нем пароли хранятся ненадежно.

GPU любят свертки

Обычно при вводе пароля компьютер вычисляет свертку от введенного пароля и сравнивает ее с сохраненной сверткой. Следовательно, взлом паролей — это процесс ввода пароля, получения его свертки и сравнения ее с сохраненной сверткой. Так как возможных паролей очень много, то важна скорость перебора; и, оказывается, графические процессоры довольно хорошо справляются с получением свертки наших паролей.

Со взлома паролей удобно начать, потому что для этой цели есть хорошо настраиваемая, простая в использовании утилита: *hashcat*. Точнее, *OpenCLHashcat Plus*. Она есть на нашем диске и на сайте <http://hashcat.net/openclhashcat-plus/>.

Утилита упакована с помощью 7-zip, поэтому нужно распаковать ее и перейти в созданный каталог:

```
7z x oclhashcat-plus-0.14.7z
cd oclhashcat-plus-0.14
```

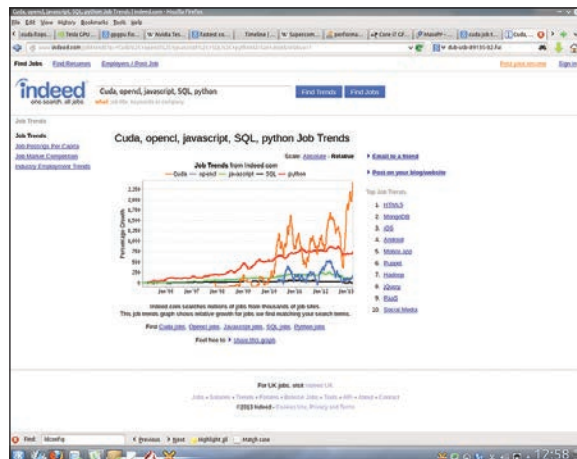
Если 7z у вас не установлен, его легко найти в репозиториях своего дистрибутива. Скомпилированные варианты утилиты есть для 64- и 32-битных компьютеров с видеокартами OpenCL (*oclhashcat*) и CUDA (*cudahashcat*). На нашем компьютере стоит видеокарта Nvidia, но если на вашем — ATI, просто замените **ocl** на последующих примерах.

Для начала протестируем систему и убедимся, что все работает правильно, а также посмотрим, быстро ли работает ваша видеокарта. Откройте скрипт *benchmark.sh* в своем любимом редакторе и измените **export BIN** так, чтобы она указывала на должный исполняемый файл. В нашем случае мы изменили ее на **export BIN="/.cudaHashcat-plus64.bin"**

```
export BIN="/.cudaHashcat-plus64.bin"
Затем запустите скрипт:
./benchmark.sh
```

Этот скрипт проверяет систему с помощью серии недопустимых свертки, чтобы посмотреть, на какой скорости она работает. Для начала нужно убедиться, что тест работает правильно. Если он выдает ошибку, нужно понять, что его остановило. Вероятнее

Согласно indeed.com, CUDA — одна из платформ с наиболее быстро растущим спросом. На графике показан рост ее процентной доли в объявлениях о работе.



ГЛОТАЕТ GPU



всего, вы указали не тот исполняемый файл или у вас нет нужных драйверов (увы, в данном контексте это бинарники), или ваша видеокарта просто не поддерживает данную возможность.

Ну вот, мы заставили программу работать; теперь посмотрим, насколько ускорил процедуру графический процессор. Для этого сравним производительность *cudaHashcat-plus* и обычной *hashcat*. Сначала нужно добыть *hashcat*. Она тоже есть на DVD и на том же сайте. Загрузите и распакуйте ее, как ранее. Теперь у вас имеется два каталога: один для *oclHashcat-plus* и один для *hashcat*. Наши находятся в `~/Downloads` и называются **oclHashcat-plus-0.14** и **hashcat-0.44** соответственно. Если у ваших имена другие (например, у вас более свежие версии) или вы распаковали файлы в другое место, измените следующие команды соответственно.

Проверяем скорость

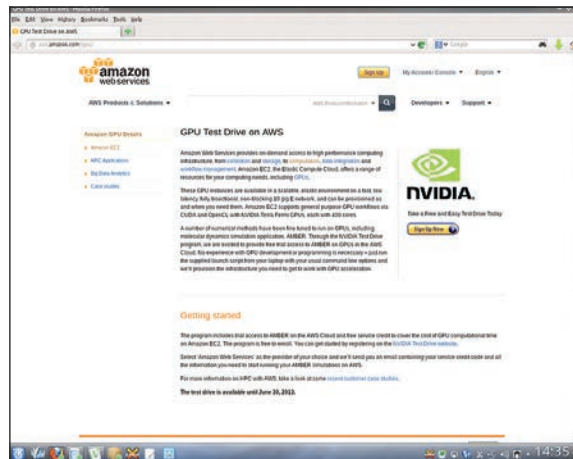
В обеих программах есть несколько примеров сверток. Мы будем запускать один и тот же пример в двух программах и оценивать их производительность. В следующем тесте используется пример сверток из *hashcat*, но список слов — из *oclHashcat-plus*, поэтому будьте внимательны с именами каталогов, если вы распаковали файлы в другой каталог. Сначала запустим тест с *hashcat*:

```
cd ~/Downloads
./hashcat-0.44/hashcat-cli64.bin -m 500 ./hashcat-0.44/examples/A0.M500.hash ./oclHashcat-plus-0.14/example.dict
```

Он должен какое-то время выполняться, а затем обнаружить две свертки. В нашей системе с Intel Core i5 это заняло около 17 секунд, но у вас это время может отличаться. Теперь посмотрим, как тест пройдет на видеокарте. Для этого скопируйте `./oclHashcat-plus-0.14/cudaHashcat-plus64.bin -m 500 ./hashcat-0.44/examples/A0.M500.hash oclHashcat-plus-0.14/example.dict`

У нас на выполнение теста ушло секунды три, причем львиную долю времени отняла передача данных видеокарте. Увеличение скорости почти шестикратное, и это впечатляет, тем более что процессор в компьютере отнюдь не из убогих. Конечно, ваши результаты, скорее всего, получатся другими и будут отчасти определяться графическим процессором. Теперь можно пробовать взломать на графическом процессоре любой пароль!

Конечно, порядочным людям компьютеры нужны вовсе не для взлома паролей. Плохая новость в том, что пока мало какие программы поддерживают обработку в графическом процессоре



► **Полезный совет:** В web-сервисах Amazon (AWS) можно добавить карты Nvidia к экземплярам облаков, чтобы воспользоваться преимуществами программирования с CUDA.

«из коробки». А хорошая новость — существуют привязки, позволяющие применять GPU с большинством распространенных языков программирования. Мы обратим внимание на Python, так как это популярный язык, легко читаемый даже теми, кто с ним не знаком. Для него есть несколько привязок, но мы воспользуемся *PyCuda* (похожая привязка для пользователей OpenCL — *PyOpenCL*). В отличие от *hashcat*, она вполне может оказаться в репозиториях вашего дистрибутива. В Ubuntu и ее производных это пакеты *python-pycuda* и *python-pyopencl*. Если в менеджере пакетов их нет, придется их загрузить и установить. Тут могут возникнуть сложности, но на сайте <http://wiki.tiker.net/PyCuda/Installation/Linux> имеются подробные инструкции.

Приведенный пример работал и в CUDA, и в OpenCL, но теперь мы будем чуть более избирательны и сосредоточимся только на CUDA. Однако, как мы упоминали выше, OpenCL основан на тех же базовых принципах, и при необходимости вы довольно легко сможете перенести туда свои идеи.

Как и прежде, мы проведем ряд математических вычислений на центральном и на графическом процессоре и рассмотрим их производительность. На сей раз мы займемся тригонометрией и рассчитаем несколько прямоугольных треугольников. Не волнуйтесь, если вы забыли школу: мы не будем торопить события. Эта арифметическая операция с плавающей точкой продемонстрирует то, с чем графический процессор хорошо справляется. »

Заказные интегральные схемы

Видеокарты значительно быстрее центральных процессоров, так как они более узко специализированы. На них не лежит груз выполнения всех задач, необходимых пользователю. Но их операции все равно бывают частично лишними. Можно копнуть глубже, и у нас получится устройство для еще более узкого диапазона задач. Такие устройства называют

проблемно-ориентированными (заказными) интегральными схемами (Application Specific Integrated Circuit — ASIC); они предназначены для выполнения одной функции, и делают это очень эффективно.

Одна из областей вычислений, которая переходит с графических процессоров на заказные интегральные схемы ради повышения производительности —

генерация [mining] биткоинов. Поскольку генерировать биткойны все труднее, приходится ускоряться. Для этого некоторые употребляют заказные схемы, так как тут не тянут даже топ-модели графических процессоров. Но коли уж схемы спроектированы под решение конкретной задачи, они и пригодны только для нее.

» **Не хотите пропустить номер?** Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

Начнем с генерации двух случайных чисел. Одно из них будет углом в прямоугольном треугольнике, второе — длиной стороны. На основании этих двух переменных мы вычислим длину двух других сторон треугольника. Конечно, для одного треугольника это слишком быстро и не стоит запуска графического процессора, поэтому увеличим количество треугольников до 100 000. В *PyCuda* это очень просто делает `gpuarray`, позволяя выполнять действия над большим числом точек данных на графическом процессоре. Работает это примерно так же, как массивы `NumPy`, и можно легко сравнить производительность.

Сначала создадим файл `triangles.py` в любимом текстовом редакторе и добавим такие строки, для подключения всех библиотек:

```
import pycuda.autoninit
import pycuda.cumath
import pycuda.gpuarray as gpuarray
import numpy
import time
```

Затем нужно сгенерировать случайные данные. Сгенерируем случайные массивы `NumPy`, а затем заполним ими `gpuarray`.

```
o_cpu = numpy.random.randn(100000).astype(numpy.float32) t_cpu = numpy.random.randn(100000).astype(numpy.float32)
o_gpu = gpuarray.to_gpu(o_cpu)
t_gpu = gpuarray.to_gpu(t_cpu)
```

Теперь выполним вычисления на GPU:

```
ts = time.clock()
h_gpu = ((o_gpu/pycuda.cumath.sin(t_gpu)))
a_gpu = pycuda.cumath.sqrt((h_gpu*h_gpu)-(o_gpu*o_gpu))
print "time taken GPU", time.clock() - ts
```

Как видите, с помощью `gpuarray` большой объем вычислений на графическом процессоре выполняется совершенно тривиально. Второй набор вычислений, который мы выполним на центральном процессоре с помощью `NumPy`, работает точно так же. Его полный код есть на DVD или на www.linuxformat.com/archives. Мы запускаем его командой

```
python triangles.py
```

По завершении нам сообщат, сколько времени потребовалось для выполнения вычислений на центральном и графическом процессорах, затем выведутся массивы результатов, чтобы мы могли

проверить, что результаты тестов одинаковы. После первого запуска у нас получилось следующее:

```
time taken GPU 0.16
time taken CPU 0.004
```

Добавляем управление на C

За что боролись?! На графическом процессоре тест работал дольше! Причем намного. Проблема с запуском таких тестов на графическом процессоре в том, что хотя они выполняются быстро, много времени уходит на пересылку данных туда и обратно и на запуск теста. Даже при 100 000 точках данных и довольно затратном вычислении квадратного корня и синуса для каждой, количества точек все же недостаточно, чтобы выигрыш компенсировал потери из-за пересылки данных. А что произойдет, если данных добавить? Мы изменили две строки, которые генерируют случайные данные, на следующие:

```
o_cpu = numpy.random.randn(1000000).astype(numpy.float32)
t_cpu = numpy.random.randn(1000000).astype(numpy.float32)
```

Появилось еще два нуля, и это должно быть напряженнее для центрального процессора. Мы запустили тест снова, и результатом было

```
time taken GPU 0.16
time taken CPU 0.24
```

Как видите, время выполнения теста на графическом процессоре практически не изменилось (так как большую часть этого времени все равно отняли накладные расходы), а на центральном процессоре стало значительно дольше. Сравнение двух методов обработки с разным числом точек данных показано на рис. 1.

Это довольно впечатляющая скорость для такого простого метода. Но не максимально возможная. Чтобы увеличить скорость, нужен больший контроль над выполнением кода на графическом процессоре. Python там не запустится, поэтому нам нужно нечто уровнем ниже: C. Не бойтесь, это не означает портирования на C всей программы: понадобится переписать на C только куски, предназначенные для выполнения на графическом процессоре.

Мы написали функцию на C, которая будет выполняться на видеокарте с помощью `SourceModule` (импортированном из *PyCuda*):

```
mod = SourceModule("""
__global__ void calculate_them(float *h, float *a, float *o, float *t)
{
    const int i = threadIdx.x + blockDim.x * blockIdx.x;
    h[i] = o[i]/sin(t[i]);
    a[i] = sqrt((h[i]*h[i])-(o[i]*o[i]));
}
""")
```

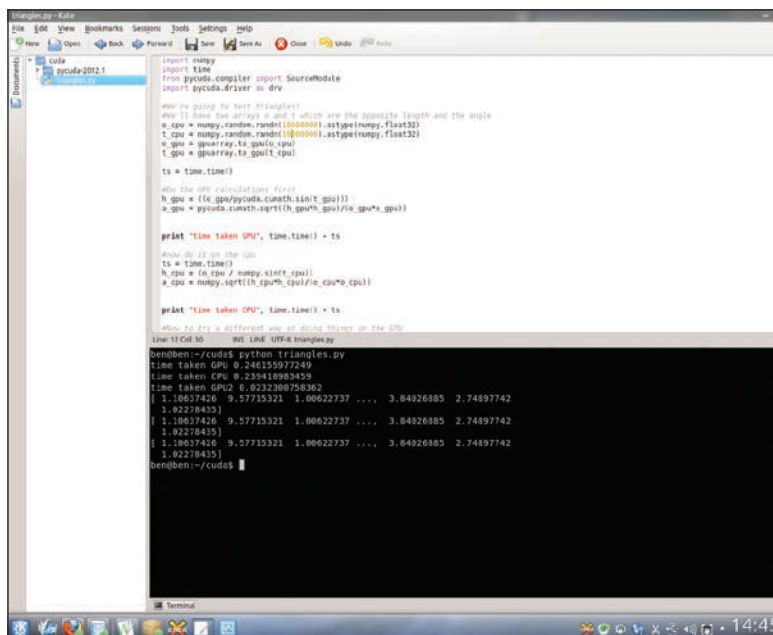
Если вы программировали на C, вам может показаться, что код выглядит неправильно. Но это не стандартный C, а CUDA C. `__global__` — ключевое слово CUDA C, оно говорит системе, что далее следует функция, выполняемая на графическом процессоре, но вызываемая из центрального. Другими словами, если нам нужно вызвать функцию из Python, ей нужно добавить префикс `__global__`. Мы также пользуемся `sqrt()` без подключения `math.h`. Дело в том, что большинство математических функций в CUDA C являются стандартными. Полный список доступных функций см. в руководстве по программированию на CUDA C на сайте Nvidia (<http://bit.ly/14sLQ9C>).

Наверное, самая странная строка здесь — вот эта:

```
const int i = threadIdx.x + blockDim.x * blockIdx.x;
```

Чтобы понять ее, нужно разобраться, как выполняется код. Если бы мы писали этот код для выполнения в одном потоке на центральном процессоре, то могли бы написать цикл, который с каждым проходом увеличивает счетчик на единицу. В каждом

➤ Одно из преимуществ интеграции с Python в том, что CUDA можно внедрить в обычные утилиты.



➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

Производительность

Производительность лучшей видеокарты от Nvidia Tesla K20X — почти 4 терафлопа. Для сравнения — одна эта видеокарта быстрее ASCI Red, компьютера, бывшего самым быстрым в мире до конца 2000 года. Нам всем известен закон Мура [Moore], по которому производительность компьютера будет удваиваться каждые восемнадцать месяцев, но переход менее чем за тринадцать лет от компьютера размером с комнату к видеокарте показывает, как быстро развиваются компьютеры. Приводим результаты наших тестов.

ОБОЗНАЧЕНИЯ

- Время выполнения на GPU
- Время выполнения на CPU
- Время выполнения на специализированном GPU

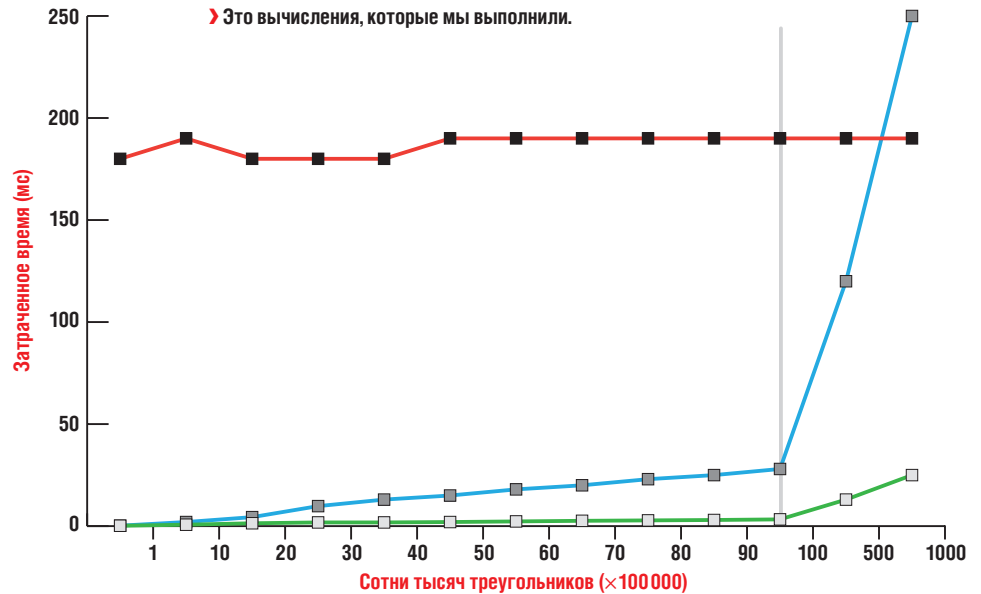


Рис. 1. За счет параллельной структуры графический процессор хорошо работает с большими объемами данных.

цикле вычислялось бы значение *i*-го элемента массива. Здесь же мы делаем нечто похожее, но вместо цикла распределяем обработку между ядрами графического процессора.

Позже мы рассмотрим размеры блоков [block], в которые группируются потоки, и сеток [grid] из них [об этом также см. LXF169], но сейчас важно отметить, что эта строка ведет себя как счетчик в цикле `for`, только для нескольких потоков, выполняющихся на нескольких устройствах. Последние строки должны быть знакомы по другому коду, который мы написали.

Выполнение на графическом процессоре

В той части мы поместили код на графический процессор, но не выполнили его. Для этого нужно еще кое-что добавить.

```
calculate_them = mod.get_function("calculate_them")
h_gpu2 = numpy.zeros_like(h_cpu)
a_gpu2 = numpy.zeros_like(a_cpu)
ts = time.clock()
calculate_them(
drv.Out(h_gpu2), drv.Out(a_gpu2), drv.In(o_cpu), drv.In(t_cpu),
block=(1000,1,1), grid=(10000,1,1))
```

В первой строке создается функция Python `calculate_them` и связывается с функцией графического процессора, которую мы только что написали. В следующих трех строках создаются массивы для выходных данных и запускается таймер, затем мы выполняем наш код.

Первые четыре параметра скрипта — просто массивы, которые у нас есть (обратите внимание, что мы можем повторно использовать массивы центрального процессора с предыдущего этапа, а не копировать их снова), последние же два — блок и сетка. Эти параметры требуют некоторых пояснений. Как мы упоминали ранее, каждый элемент данных вычисляется в отдельном потоке. Потоки объединяются в блоки, а блоки — в сетку. В этом примере мы пользуемся не всеми возможностями CUDA. Например, каждый поток использует только свои собственные данные и не взаимодействует с другими потоками. Однако в блоках есть разделяемая память, которой потоки могут пользоваться.

Кроме доступа к памяти, размер блоков и сеток оказывает большое влияние на производительность. Чтобы вычислить все точки данных, произведение размера блока на размер сетки должно быть равным размеру массива. Кроме того, размер блока ограничен аппаратно. При массивах в десять миллионов элементов тест можно запустить с любым размером блока от 1 до 1024 (на нашем устройстве было 512). Методом проб и ошибок мы определили, что тест выполняется быстрее при большем размере блока и в самом медленном случае примерно вдвое быстрее центрального процессора, а в самом быстром — раз в десять быстрее. Возможно, вы заметили, что параметры блока и сетки поддерживают три измерения, а не одно. Однако обратите внимание, что предел в 1024 потока на блок является общим по всем измерениям, поэтому размер блока может быть 10, 10, 10, но не 100, 100, 100. После нескольких попыток мы сумели получить существенный рост скорости для блока 1000, 1, 1.

Как мы убедились, графический процессор выигрывает при увеличении числа точек данных, но его относительная скорость возрастает не при любом увеличении количества информации. Потребительские графические процессоры 32-битные, и вы, возможно, заметили, что мы работаем с типами данных `float32` (32-битное с плавающей точкой). Если изменить его на `float64`, вы увидите, что графический процессор замедлится примерно вдвое, тогда как производительность центрального процессора упадет не столь значительно.

Графический процессор — не просто еще один процессор. У него особые характеристики, и некоторые задачи он умеет выполнять невероятно быстро — а именно, задачи, похожие на графический рендеринг, с хорошо распараллеливаемыми данными одинарной точности. В других случаях он может отставать от центрального процессора. Но похожие данные фигурируют в большей части научных вычислений, и потенциал для ускорения огромен. Особенно если учесть, что на одном компьютере может быть несколько графических процессоров. Существуют материнские платы, поддерживающие несколько видеокарт, привязанных к одному процессору, и так можно построить свой собственный суперкомпьютер — только не забудьте об охлаждении. LXF

Erlang: ОПЯТЬ

Андрей Ушаков вынужден будет приостановить полет мысли из-за недостатка места. Но уж в следующий раз мы *все* узнаем!



Наш эксперт

Андрей Ушаков активно приближает тот день, когда функциональные языки станут мейнстримом.

Мы продолжаем решение нашей задачи: создание многозадачных версий функций `map` и `reduce`. Но на сей раз мы постараемся завершить начатое в предыдущей статье и не начнем ничего нового.

По традиции, вспомним, на чем мы остановились в прошлой статье. Мы поговорили о том, что явное управление назначением заданий рабочим процессам далеко не всегда эффективно. Мы помним, что раньше мы назначали задание на обработку порции данных тому рабочему процессу, который обработал некоторую порцию исходных данных и послал результаты обработки главному процессу. Предположим, что мы назначаем задания процессам 1, 2 и 3, располагающимся каждый на своем узле (а каждый из узлов — на отдельном компьютере). Предположим, что узел, на котором располагается процесс 3, сильно загружен, что выражается в более долгой обработке порций исходных данных этим процессом. В наиболее вырожденном случае процессы 1 и 2 могут обработать все остальные порции быстрее, чем процесс 3 обработает назначенную ему порцию. В итоге мы не получаем ожидаемого ускорения обработки исходных данных за счет ее распараллеливания, а в вырожденном случае обрабатываем данные даже медленнее, чем при обычной, не параллельной обработке. Конечно, если рабочие процессы выполняются на узлах, расположенных на одинаковых компьютерах с одинаковой загрузкой (как узлов, так и операционной системы в целом), то такая ситуация вряд ли возможна. Но уж если она имеет место, необходимо использовать какой-то другой механизм назначения заданий вместо назначения задания процессу, вернувшему результат обработки некоторой порции данных.

Проблема назначения задания некоторой единице выполнения кода (потоку и/или процессу) не является специфичной для языка Erlang. Во многих средах времени выполнения и операционных

системах вопрос о назначении задания некоторой единице выполнения кода решается при помощи пула потоков. Пул потоков — это набор некоторых уже созданных потоков, которые служат для выполнения того или иного пользовательского задания. Причем когда поток из пула выполнит пользовательское задание, он не прекращает свое существование (как это бывает с потоками, созданными специально для выполнения пользовательских заданий), а «возвращается» обратно в пул потоков. Существование подобного механизма связано с тем, что в таких средах времени выполнения и операционных системах создание новой единицы выполнения кода — достаточно дорогая операция. В среде времени выполнения Erlang ситуация с созданием новых процессов кардинально иная. Создание процессов является настолько легким, что рекомендуется создавать их столько, сколько нужно для решения той или иной задачи. Поэтому создание некоторого пула процессов языка Erlang не имеет особого смысла. С другой стороны, мы можем создать пул узлов, который будет выполнять пользовательские задания на одном из своих узлов (естественно, создавая для этого процесс) в зависимости от того или иного условия (например, от загрузки узлов). Очевидно, создание такого пула узлов имеет смысл; и такой пул узлов в стандартной библиотеке языка Erlang есть.

Разобравшись с возможным решением проблемы назначения очередного задания на обработку, мы перешли к реализации общей функциональности, на базе которой мы в дальнейшем реализуем очередные многозадачные версии функций `map` и `reduce`. Эту общую функциональность мы реализовали в модуле `parallel_pool_helper`. Из этого модуля мы экспортируем две функции: точку входа в общую функциональность — функцию `parallel_pool_helper:pool_core/5` и тело рабочего процесса — функцию `parallel_pool_helper:pool_worker/4`. При реализации этой общей функциональности мы приняли важное решение, договорившись, что конкретизировать механизм назначения заданий мы будем в функциях, использующих нашу общую функциональность. Для этого в функцию `parallel_pool_helper:pool_core/5` в качестве одного из параметров мы передаем интерфейс (интерфейсную функцию) к механизму назначения заданий. При таком подходе наша общая функциональность не будет зависеть от конкретного механизма назначения заданий, специфицировать который мы будем в функциях, использующих нашу общую функциональность. Чтобы упростить интерфейс к механизму назначения заданий, мы решили экспортировать функцию `parallel_pool_helper:pool_worker/4` из модуля `parallel_pool_helper`, хотя делать это не обязательно.

И, наконец, мы посмотрели, что же представляет собой пул узлов из стандартной библиотеки языка Erlang. Пул узлов определен в модуле `pool`. Этот модуль может как самостоятельно создавать узлы на основе имен хостов, определенных в файле `.hosts.erlang`, так и использовать заранее созданные узлы. Для начала работы с пулом узлов необходимо вызвать одну из функций `pool:start/1` или `pool:start/2`; при этом пул узлов создает узлы на хостах, определенных в файле `.hosts.erlang`, если есть такая возможность. Чтобы использовать заранее созданные узлы,

Исходники библиотеки Erlang

На данном уроке мы заглянем в исходные коды модуля `pool`. Однако мы не затронули вопрос, а где вообще лежат исходные коды библиотек языка Erlang (и в частности, модуля `pool`). Сейчас мы этот вопрос разберем. Корневая директория среды выполнения Erlang (та, в которую мы устанавливаем среду выполнения Erlang) определяется переменной среды окружения `$OTPROOT`; также она может быть определена через вызов функции `code:root_dir/0`. Директория `$OTPROOT/lib` содержит все библиотеки, идущие вместе со средой выполнения. Согласно принципам OTP, все библиотеки принадлежат некоторому приложению; поэтому директория `$OTPROOT/lib` содержит поддиректории, соответствующие таким приложениям.

Каждая из директорий приложений всегда содержит следующие поддиректории: `doc`, `ebin`, `src`. Директория `doc` содержит документацию на приложение (на модули, которые данное приложение содержит). Директория `ebin` содержит все модули в откомпилированном виде (в виде файлов `*.beam`). И, наконец, директория `src` содержит исходные коды всех модулей. Следует также учесть, что директории приложений помимо имени приложения содержат также его версию. Так, например, на Ubuntu `$OTPROOT` будет `/usr/lib/erlang`, модуль `pool` принадлежит приложению `stdlib`, поэтому полный путь до исходного кода модуля `pool` будет следующий `/usr/lib/erlang/lib/stdlib-1.18.1/src/pool.erl`.

практикуемся

необходимо каждый из таких узлов добавить в пул при помощи функции `pool:attach/1`. При этом временем жизни такого переданного узла будет управлять пул узлов. Для выполнения некоторого задания (вычисления функции, определенной в некотором модуле, с заданными аргументами) используется одна из функций `pool:pspawn/3` или `pool:pspawn_link/3`. Обе эти функции создают процесс на одном из узлов пула, где ожидается наименьшая загрузка, и возвращают идентификатор созданного процесса, что позволяет дополнительно с ним взаимодействовать. Их отличие только в том, что первая функция просто создает процесс, а вторая, помимо процесса, создает и связь между созданным процессом и вызывающим процессом. И, наконец, чтобы закончить работу с пулом узлов, необходимо вызвать функцию `pool:stop/0`; при этом все узлы пула будут уничтожены.

А теперь мы можем пойти дальше и реализовать очередные многозадачные версии функций `map` и `reduce`. Эти функции будут использовать общую функциональность, определенную нами в прошлый раз; причем в качестве механизма назначения заданий мы будем использовать пул узлов, определенный в модуле `pool`. При этом нам необходимо учесть следующий факт: в зависимости от операционной системы и окружения пул узлов либо сможет создать узлы на хостах, определенных в файле `.hosts.erlang`, либо не сможет. Кроме того, в каких-то ситуациях пользователи наших функций `map` и `reduce` могут захотеть использовать созданные ими узлы в пуле узлов. Поэтому наши очередные версии функций `map` и `reduce` должны принимать в качестве одного из параметров список узлов, который будет передан пулу узлов. Соответственно, в начале выполнения функций `map` и `reduce` мы должны инициализировать (при помощи одной из функций `pool:start/1` или `pool:start/2`) пул узлов, а в конце выполнения этих функций завершить его работу (при помощи функции `pool:stop/0`). Если же в силу тех или иных причин вам необходимо использовать уже инициализированный пул узлов и нет нужды его останавливать, то наша реализация функций `map` и `reduce` вам не подойдет. На самом деле это не является проблемой, но об этом мы поговорим, когда рассмотрим реализацию функций `map` и `reduce`.

Прежде чем приступить к долгожданной реализации функций `map` и `reduce`, давайте реализуем пару функций, упрощающих нам работу с пулом узлов. Это будут функции `pool_helper:start_pool/2` и `pool_helper:stop_pool/0`, экспортируемые из модуля `pool_helper`:

```
start_pool(Prefix, PoolNodes) ->
    pool:start(Prefix),
    lists:foreach(fun(Node) -> pool:attach(Node) end, PoolNodes).
stop_pool() ->
    pool:stop().
```

Функция `pool_helper:start_pool/2` инициализирует пул узлов, после чего добавляет узлы из списка `PoolNodes` в пул узлов. Функция `pool_helper:stop_pool/0` завершает работу пула узлов. Видно, что функция `pool_helper:stop_pool/0` всего лишь вызывает функцию `pool:stop/0`. Может показаться, что введение функции `pool_helper:stop_pool/0` излишне и ее стоит удалить. На самом деле для создания этой функции есть две причины. Во-первых, для симметрии — чтобы у функции инициализации пула узлов

`pool_helper:start_pool/2` была функция-антипод для завершения работы узлов; во-вторых, для возможных будущих изменений, когда при завершении работы пула узлов нам необходимо будет выполнять еще какие-либо действия.

Пришла долгожданная пора реализовать очередные версии функций `map` и `reduce`. Начнем мы по традиции с многозадачной версии функции `map`, основанной на использовании пула узлов. В этой функции мы инициализируем пул узлов (при помощи функции `pool_helper:start_pool/2`), вычисляем результат операции отображения исходных данных, после чего завершаем работу пула узлов (при помощи функции `pool_helper:stop_pool/0`) и возвращаем результат. В нашем случае это будет функция `parallel_map:pool_pmap/5`, определенная в модуле `parallel_map` и экспортируемая из него:

```
pool_pmap(Fun, Source, PortionSize, WorkerCount, PoolNodes) ->
    pool_helper:start_pool(pmap, PoolNodes),
    Result = pool_pmap_impl(Fun, Source, PortionSize,
        WorkerCount),
    pool_helper:stop_pool(),
    Result.
```

Результат операции отображения исходных данных вычисляется (естественно, многозадачным образом) в функции `parallel_map:pool_pmap_impl/4`:

```
pool_pmap_impl(_Fun, [], _PortionSize, _WorkerCount) -> [];
pool_pmap_impl(Fun, Source, PortionSize, _WorkerCount)
when length(Source) =< PortionSize -> lists:map(Fun, Source);
pool_pmap_impl(Fun, Source, PortionSize, WorkerCount) ->
    WorkerFun = fun(Portion) -> lists:map(Fun, Portion) end,
    AssignFun = fun(Portion, Index, Master) -> pool:pspawn_
        link(parallel_pool_helper, pool_worker, [WorkerFun, Portion,
            Index, Master]) end,
    parallel_pool_helper:pool_core(fun lists:append/1, Source,
        PortionSize, AssignFun, WorkerCount).
```

Эта функция содержит три варианта и обрабатывает, помимо общего случая, еще пару граничных случаев. Первый вариант функции `parallel_map:pool_pmap_impl/4` обрабатывает случай, когда исходные данные отсутствуют (т.е. исходные данные — пустой список). Второй вариант этой функции обрабатывает случай, когда размер исходных данных не больше размера порции данных; в этом случае нет смысла обрабатывать исходные данные многозадачным способом. И, наконец, третий вариант функции `parallel_map:pool_pmap_impl/4` обрабатывает общий случай, когда мы используем нашу многозадачную реализацию из модуля `parallel_pool_helper`. В этом варианте функции `parallel_map:pool_pmap_impl/4` мы создаем интерфейс (интерфейсную функцию `AssignFun`) к механизму назначения заданий на базе пула узлов (в нашем случае — к функции `pool:pspawn/3`), после чего используем общую функциональность, реализованную в модуле `parallel_pool_helper`. Если сравнить функции `parallel_map:pool_pmap/5` и `parallel_map:pool_pmap_impl/4`, то видно, что функция `parallel_map:pool_pmap_impl/4` вычисляет результат операции отображения уже с использованием готового пула узлов и не управляет его жизнью, в отличие от функции

»

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

Создать условия пулу

При инициализации (которая происходит при помощи вызова одной из функций `pool:start/1` или `pool:start/2`) пул узлов может создавать узлы на хостах, определенных в файле `.hosts.erlang`. Для создания узлов на удаленных хостах используется утилита `rsh`. Можно использовать альтернативу данной утилите; для этого необходимо среду выполнения Erlang запускать с ключом `-rsh` и именем альтернативы, отделенным пробелом. Естественно, что когда мы используем

темную сторону силы (одну из операционных систем семейства Microsoft Windows), то создавать узлы на удаленных хостах мы не можем. Тем не менее, можно создавать узлы на локальном хосте. Однако тут существует небольшая хитрость. Если при задании локального хоста (в файле `.hosts.erlang`) мы используем имя `localhost`, то узел создан не будет. Чтобы узел на локальном хосте был создан, необходимо использовать имя этого хоста, например, `stdstring`.

`parallel_map:pool_pmap/5`. Таким образом, мы экспортируем из модуля `parallel_map` обе функции `parallel_map:pool_pmap/5` и `parallel_map:pool_pmap_impl/4`. Первую функцию мы используем тогда, когда нам не нужно управлять временем жизни пула узлов и нас устраивает сценарий, что пул узлов будет создан в начале ее выполнения и завершен в конце ее выполнения. Вторую же функцию мы используем тогда, когда у нас есть некоторый пул узлов и мы желаем управлять его временем жизни (в этом случае, в отличие от предыдущего, у нас получается долгоживущий пул узлов).

После реализации многозадачной версии функции `map` перейдем к рассмотрению многозадачной версии функции `reduce`, основанной на использовании пула узлов. В этой функции мы инициализируем пул узлов (при помощи функции `pool_helper:start_pool/2`), вычисляем результат операции свертки исходных данных, после чего завершаем работу пула узлов (при помощи функции `pool_helper:stop_pool/0`) и возвращаем результат. В нашем случае это будет функция `parallel_reduce:pool_reduce/6`, определенная в модуле `parallel_reduce` и экспортируемая из него:

```
pool_reduce(Fun, Source, {InitValue, PortionInitValue},
PortionSize, WorkerCount, Nodes) ->
pool_helper:start_pool(pmap, Nodes),
Result = pool_reduce_impl(Fun, Source, {InitValue,
PortionInitValue}, PortionSize, WorkerCount),
pool_helper:stop_pool(),
Result.
```

Результат операции свертки исходных данных вычисляется (естественно, многозадачным образом) в функции `parallel_reduce:pool_reduce_impl/5`:

```
pool_reduce_impl(_Fun, [], {InitValue, _PortionInitValue}, _
PortionSize, _WorkerCount) ->
InitValue;
pool_reduce_impl(Fun, Source, {InitValue, _PortionInitValue},
PortionSize, _WorkerCount)
when length(Source) <= PortionSize -> lists:foldl(Fun, InitValue,
Source);
pool_reduce_impl(Fun, Source, {InitValue, PortionInitValue},
PortionSize, WorkerCount) ->
WorkerFun = fun(Portion) -> lists:foldl(Fun, PortionInitValue,
Portion) end,
AssignFun = fun(Portion, Index, Master) -> pool:pspawn_
link(parallel_pool_helper, pool_worker, [WorkerFun, Portion,
Index, Master]) end,
FinalAggFun = fun(List) -> lists:foldl(Fun, InitValue, List) end,
```

```
parallel_pool_helper:pool_core(FinalAggFun, Source,
PortionSize, AssignFun, WorkerCount).
```

Эта функция содержит три варианта и обрабатывает, помимо общего случая, еще пару граничных случаев. Первый вариант функции `parallel_reduce:pool_reduce_impl/5` обрабатывает случай, когда исходные данные отсутствуют (когда исходные данные — пустой список). Второй вариант этой функции обрабатывает случай, когда размер исходных данных не больше размера порции данных; тогда нет смысла многозадачным способом обрабатывать исходные данные. И, наконец, третий вариант функции `parallel_reduce:pool_reduce_impl/5` обрабатывает общий случай, когда мы используем нашу многозадачную реализацию из модуля `parallel_pool_helper`. В этом варианте функции `parallel_reduce:pool_reduce_impl/5` мы создаем интерфейс (интерфейсную функцию `AssignFun`) к механизму назначения заданий на базе пула узлов (в нашем случае — к функции `pool:pspawn/3`), после чего используем общую функциональность, реализованную в модуле `parallel_pool_helper`. Про отношение между функциями `parallel_reduce:pool_reduce/6` и `parallel_reduce:pool_reduce_impl/5` можно сказать то же, что и про отношение между функциями `parallel_map:pool_pmap/5` и `parallel_map:pool_pmap_impl/4`. Первая функция позволяет не задумываться об управлении жизнью пула узлов и берет ответственность за это на себя (за создание пула узлов, добавление необходимых узлов к пулу узлов и завершение его работы). Вторая функция перекладывает ответственность за управление жизнью пула узлов на вызывающую сторону и полагается во время своей работы на то, что пул узлов создан и нужным образом проинициализирован. Естественно, что мы экспортируем из модуля `parallel_reduce` обе эти функции.

Давайте проверим, что наши реализации многозадачных версий функций `map` и `reduce` на базе пула узлов работают. Прежде чем начать проверку, давайте слегка модифицируем функцию `parallel_pool_helper:pool_worker/4` следующим образом: добавим в начало тела этой функции вывод имени узла на консоль, на которой функция в данный момент выполняется. Выглядеть наша модификация будет следующим образом:

```
pool_worker(Fun, Portion, Index, Master) ->
io:format("~p~n", [node()]),
Result = Fun(Portion),
Master ! #task_result{index = Index, result = Result}.
```

При помощи данной модификации мы сможем убедиться, что у нас действительно создаются и выполняются процессы на узлах из пула узлов.

Теперь можно перейти непосредственно к проверке наших реализаций функций `map` и `reduce`. Первым делом давайте создадим три узла со следующими короткими именами: `node1`, `node2`, `node3`. Полные имена созданных узлов будут такими (с учетом, что имя компьютера автора — `stdstring`): `node1@stdstring`, `node2@stdstring`, `node3@stdstring`. Узлы с именами `node2` и `node3` мы будем использовать в качестве дополнительных узлов для пула узлов; узел с именем `node1` будет у нас основным — на нем мы будем производить проверку наших реализаций функций `map` и `reduce`. Как всегда, начнем проверку с проверки реализации функции `map`. Вызовом `parallel_map:pool_pmap(fun(X) -> X+0.1 end, [], 2, 3, ['node1@stdstring', 'node2@stdstring', 'node3@stdstring'])` мы проверяем работу функции `parallel_map:pool_pmap/5` в ситуации, когда список исходных данных пуст. Другими словами, мы проверяем работу первого варианта функции `parallel_map:pool_pmap_impl/4`. Этот вызов, как и ожидается, вернет нам пустой список. Очевидно, что при этом рабочие процессы на узлах из пула узлов создаваться не будут; мы это можем увидеть по отсутствию в выводе на узле `node1` имен узлов из пула

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

узлов. Вызов `parallel_map:pool_pmap(fun(X) -> X+0.1 end, [5, 13], 2, 3, ['node1@stdstring', 'node2@stdstring', 'node3@stdstring'])` проверяет работу функции `parallel_map:pool_pmap/5` в ситуации, когда размер список исходных данных не больше размера порции данных. Другими словами, мы проверяем работу второго варианта функции `parallel_map:pool_pmap_impl/4`. Этот вызов вернет нам список элементов [5.1, 13.1], что совпадает с ожидаемым результатом. Точно так же, как и в прошлый раз, рабочие процессы на узлах из пула узлов создаваться не должны; мы это можем увидеть по отсутствию в выводе на узле `node1` имен узлов из пула узлов. И, наконец, вызов `parallel_map:pool_pmap(fun(X) -> X+0.1 end, [1, 3, 4, 7, 8, 11, 2, 19], 2, 3, ['node1@stdstring', 'node2@stdstring', 'node3@stdstring'])` проверяет работу функции `parallel_map:pool_pmap/5` в ситуации, когда мы список исходных данных обрабатываем с использованием многозадачности. Другими словами, мы проверяем работу третьего варианта функции `parallel_map:pool_pmap_impl/4`. Этот вызов вернет нам список элементов [1.1, 3.1, 4.1, 7.1, 8.1, 11.1, 2.1, 19.1], что совпадает с ожидаемым результатом. Исходный список содержит 8 элементов, а размер порции данных — 2 элемента, поэтому исходный список при обработке будет разбит на 4 порции. Так как рабочих процессов может быть не больше 3, а у нас 4 порции исходных данных, то при обработке исходных данных должны быть задействованы все узлы (и один из узлов дважды). И действительно, вывод на узле `node1` будет содержать имена всех трех узлов, причем имя одного из узлов встретится дважды.

Следующий шаг, который следовало бы сделать — это проверить работу реализации функции `reduce` (в нашем случае это будет функции `parallel_reduce:pool_reduce/6` и `parallel_reduce:pool_reduce_impl/5`). Однако давайте эту проверку оставим читателям, кому интересно ее провести (или поверьте на слово автору, что эти функции проверены и работают), и несколько углубимся в то, как работает пул узлов (реализованный в модуле `pool`). Для этого придется рассмотреть исходный код модуля `pool`, а также документацию к этому модулю.

В документации утверждается, что функции `pool:pspawn/3` и `pool:spawn_link/3` создают процесс на узле (из пула) с минимальной ожидаемой загрузкой узла; также, такой узел с минимальной загрузкой возвращает функция `pool:get_node/0`. Загрузка узла вычисляется при помощи функции `statistics(run_queue)`, которая возвращает количество процессов, готовых к выполнению. Однако если мы посмотрим на исходный код модуля `pool`, то мы найдем там следующую информацию, которая, что естественно, специфична для данной реализации пула узлов (а реализация может измениться в любой момент). Пул узлов хранит каждый узел вместе со значением загрузки узла в списке пар (кортежей), причем этот список отсортирован по значению загрузки узла по возрастанию. Это означает, что голова этого списка всегда содержит узел (точнее, пару узел–загрузка) с минимальной загрузкой. Значение загрузки каждого узла периодически пересчитывается (в данной реализации, каждые 2 секунды), после чего список пар перестраивается в соответствии с пересчитанными значениями загрузки узлов. При обращении к одной из функций `pool:pspawn/3`, `pool:spawn_link/3` или `pool:get_node/0` пара узел — значение загрузки этого узла из головы переносится в хвост списка пар, при этом значение загрузки узла увеличивается на единицу.

Для нас это означает следующее: если у нас пул узлов содержит `N` узлов с разной загрузкой, и мы захотим достаточно быстро выполнить при помощи этого пула `N` заданий, то задания будут распределены между всеми узлами. Давайте это продемонстрируем. Для этого мы создадим модуль `load_sample` с тремя экспортируемыми функциями `load_sample:create_load/1`, `load_sample:ping/0` и `load_sample:pong/0`:

```
create_load(0) -> true;
create_load(Count) ->
```

Как зовут тебя, узел?

Узел — это именованный экземпляр среды выполнения Erlang. Для создания узла при запуске среды выполнения Erlang необходимо задать его имя, используя один из ключей `-sname` или `-name`. Ключ `-sname` задает короткое имя узла; ключ

`-name` задает длинное имя узла. При работе с узлами следует помнить следующее правило: необходимо всегда создавать узлы с одним типом имен. Узлы с разными типами имен не смогут взаимодействовать друг с другом.

```
PingPID = spawn(load_sample, ping, []),
PongPID = spawn(load_sample, pong, []),
PingPID ! {ping, PongPID},
create_load(Count-1).
ping() ->
receive
  {ping, PID} -> PID ! {pong, self()}, ping()
end.
pong() ->
receive
  {pong, PID} -> PID ! {ping, self()}, pong()
end.
```

Видно, что для создания загрузки мы создаем `N` процессов, отправляющих сообщения друг другу. Очевидно, что при создании этих `N` пар процессов вызов `statistics(run_queue)` вернет нам значение, очень близкое к `N`.

А теперь посмотрим, как работает пул узлов при их неравномерной загрузке. Как и раньше, создадим 3 узла: `node1`, `node2` и `node3`. На узле `node3` мы создадим нагрузку вызовом `load_sample:create_load(10000)`. То, что нагрузка создалась, можно проверить при помощи вызова функции `statistics(run_queue)` на узле `node3`. После этого на узле `node1` создаем пул узлов вызовом `pool_helper:start_pool(test, ['node1@stdstring', 'node2@stdstring', 'node3@stdstring'])`. Сейчас у нас есть пул узлов, причем один из узлов нагружен сильнее, чем два других. На узле `node1` выполним следующее выражение: `lists:foreach(fun(_ -> io:format(«~p~n», [pool:get_node()]) end, lists:seq(1, 9))`. Это выражение выводит на экран имя наименее загруженного узла 9 раз. Так как после обращения к функции `pool:get_node/0` головной элемент переносится в конец списка узлов, а наш вызов происходит практически моментально, то мы видим, что все наши 3 узла выводятся на экран одинаковое количество раз. А сейчас на узле `node1` давайте выполним следующее выражение: `lists:foreach(fun(_ -> io:format(«~p~n», [pool:get_node()]), timer:sleep(4000) end, lists:seq(1, 9))`. Это выражение отличается от предыдущего только тем, что после каждого вызова `pool:get_node/0` мы ждем достаточно большое время (4 секунды), чтобы произошел пересчет статистики по загрузке узлов на пуле узлов. Результат выполнения этого выражения будет отличаться от предыдущего: на экране мы увидим только имена узлов `node1` и `node2`. Из данной демонстрации можно сделать следующий вывод: пул узлов из модуля `pool` не очень хорошо подходит в ситуации, когда нам необходимо в достаточно короткий промежуток времени создать много заданий на обработку при неравномерной загрузке узлов пула.

На нашем уроке мы закончили с очередной многозадачной реализацией функций `map` и `reduce` (в этот раз на основе пула узлов) и проверили их работоспособность. Также мы разобрались с ограничениями реализации пула узлов из библиотеки языка Erlang (модуль `pool`). Автор в прошлый раз обещал, что мы подведем итоги решения задачи о реализации многозадачных версий функций `map` и `reduce`. К сожалению, из-за ограниченности места журнала автор сделать этого не смог (хотя желание, чтобы весь журнал целиком посвятили одному ему, у него есть). Обещанные итоги мы подведем в следующий раз. [LXF](#)

ЧАСТЬ 1

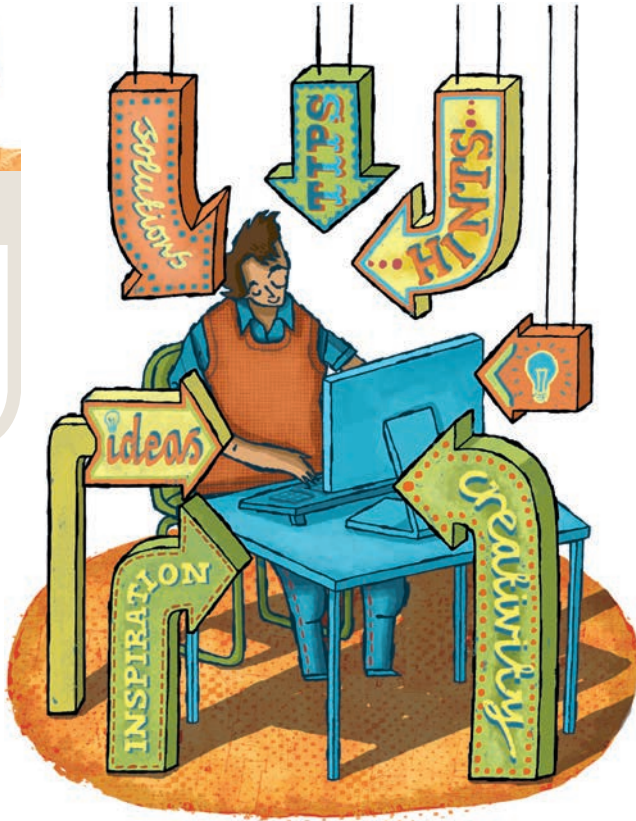
Видео: Средства

Петр Семилетов не без здоровой критики оценивает возможности подручных свободных программ для изготовления видеопродукции.



Наш эксперт

Петр Семилетов
Разработчик звукового редактора ЕКО и текстового редактора ТЕА, музыкант, звукорежиссер и краевед.



Программы для нелинейного видеомонтажа востребованы многими — от коммерческой киноиндустрии до независимых любительских студий. Программное обеспечение этой категории — сложное, а в проприетарном случае еще и весьма дорогое. Чем может ответить свободное ПО и ПО с открытым кодом, и есть ли альтернативы продуктам от Sony, Adobe, Avid? Сегодня мы познакомимся с состоянием ПО для нелинейного монтажа под Linux, а далее перейдем к практикуму работы с одним из лучших его представителей — *Kdenlive*.

Введение

Сначала — немного терминологии. Так сложилось, что в русском языке пока отсутствуют не то чтобы удобные сокращения, но даже устоявшиеся названия для двух категорий программного обеспечения, в англоязычном мире именуемых DAW и NLE. DAW — это сокращение от “Digital Audio Workstation”, то есть программа либо аппаратно-программный комплекс для записи и сведения звука. Это такие программы (привожу примеры не только для Linux, но и других платформ), как *Ardour*, *QTractor*, *Reaper*, *Nuendo*.

A NLE — это “Non-Linear Editor”. Так называют класс программ для видеомонтажа, причем не простых, а с поддержкой нескольких параллельных дорожек, наложением различных эффектов и так далее. NLE следует отличать от линейных редакторов

видео, таких как *Avidemux* либо *VirtualDub*. В линейных редакторах у вас всего одна дорожка. Такие программы используются в основном для удаления из видео фрагментов, конвертирования в другие форматы, изменения размеров и тому подобных функций. NLE же предназначены именно для монтажа — у вас есть монтажный стол, так называемый «таймлайн [timeline]», на нем множество дорожек; на дорожках вы свободно располагаете фрагменты видео, звука, можете динамически менять параметры эффектов (это называется автоматизацией и присутствует не во всех NLE).

Как правило, NLE в той или иной мере обеспечивает функциональность DAW.

K NLE вплотную примыкает класс программ, предназначенных для композитинга. Композитинг (или совмещение, но монтажеры так не говорят) — это разного рода эффекты, так или иначе использующие послынную структуру, когда несколько изображений совмещаются в одно. Например, известный эффект «синего экрана», когда синий фон подменяется любым другим фоном, включая произвольное видео — это эффект композитинга.

Пример «чистой» программы для композитинга — *AfterEffects* от Adobe. Некоторые NLE оснащены функциями композитинга, причем коммерческие — в той мере, чтобы не конкурировать с «соседней» отдельной программой композитинга от того же производителя, а некоммерческие — как придется.

В мире коммерческого ПО рынок NLE жестко поделен между несколькими компаниями, предлагающими либо программные, либо аппаратно-программные решения. Конкуренция не так уж много, ибо и область специфическая, и само ПО — одно из сложнейших, которое можно себе представить. Выпускаются новые версии одних и тех же программ. Новшества зачастую надуманные, не служащие откликом на запросы пользователей, а напротив — формируется предложение без наличия спроса, поскольку все нужные функции уже воплощены, но выпускать новые версии надо: коммерческое ПО не может позволить себе остановить производство. Иначе вылетит в трубу.

Основное отличие свободного ПО и ПО с открытым кодом от коммерческого проприетарного в том, что в первых двух, как правило, спрос рождает предложение, а не наоборот. Когда наоборот, это встречает сопротивление и порицание. (Потому и ругают Gnome 3.) Естественное же развитие свободных программ — это постепенное появление в программе недостающих функций и исправление ошибок. Программы, созданные таким образом, не раздуваются в разы, подобно коммерческим аналогам, и обладают взвешенным набором функций — функций, необходимых для работы, а не придуманных отделом маркетинга.

Программы NLE в Linux

Какие NLE предложит нам Linux? Есть ли в них «взвешенный набор функций» и достаточно ли их возможностей для работы над монтажом столь же продуктивной, как в коммерческих NLE?

Если сравнивать с флагманами коммерческих NLE, то ответ, увы, «нет». Отчасти потому, что основной рынок NLE находится вне мира свободного ПО, где, как я отметил, программы

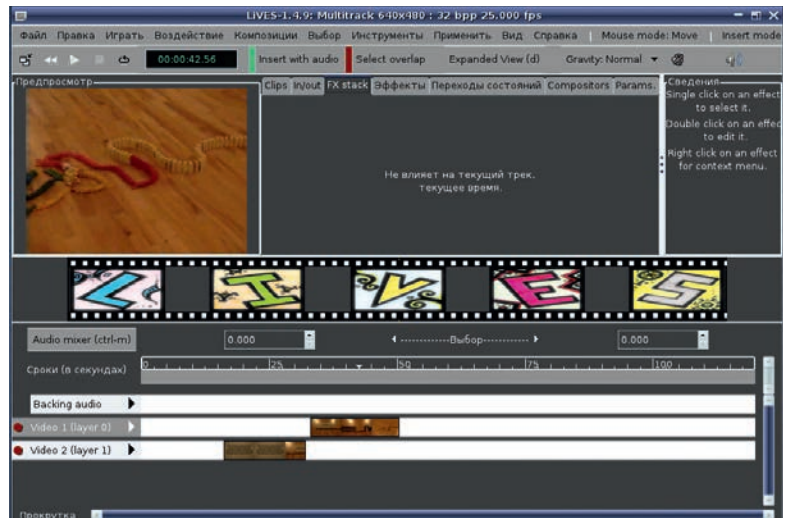
МОНТАЖ

развиваются в ответ на спрос. Нет рынка — нет спроса на различные функции. Чем шире рынок — тем больший диапазон функций будет востребован. А пока рынок узкий, это сказывается на возможностях программ.

Давайте же посмотрим, какие программы доступны под Linux для нелинейного видеомонтажа. Ниже я выражу мнения, которые неоднократно высказывал в общении с читателями и просто в разговорах. Эти мнения основаны на сравнении моей потребности в продуктивном средстве монтажа и возможностями, предоставляемыми программой. При определенной сноровке монтировать можно в чем угодно. У пионеров кинематографа не было самой простейшей NLE, но посмотрите на монтаж в фильмах отечественных пионеров документального кино Дзиги Вертова и Михаила Кауфмана. Однако, если вам нужно делать монтаж, сосредоточившись на нем, а не на борьбе со спецификой монтажной программы, то философия о пригодности «любых средств» отходит на второй план. Итак...

LIVES

LIVES (lives.sourceforge.net) — весьма живой и, я бы сказал, идейный проект Гэбриела Финча [Gabriel Finch] и его соратников. Идейный потому, что в весьма обширном руководстве пользователя много сказано о движении свободного ПО и некоторых смежных явлениях, например о проекте fyne.org. Написан на C с использованием инструментария *GTK+2*, но требует множества других программ, библиотек и интерпретаторов языков программирования — Python, Perl. На одной чаше весов — множество интересных функций, на другой — скорость работы и интерфейс. Например, чтобы изменить границы фрагмента видео на монтажном столе, надо переключаться в отдельное окно и задавать границы там, вместо того, чтобы просто сдвинуть нужную границу мышью прямо в таймлайне.

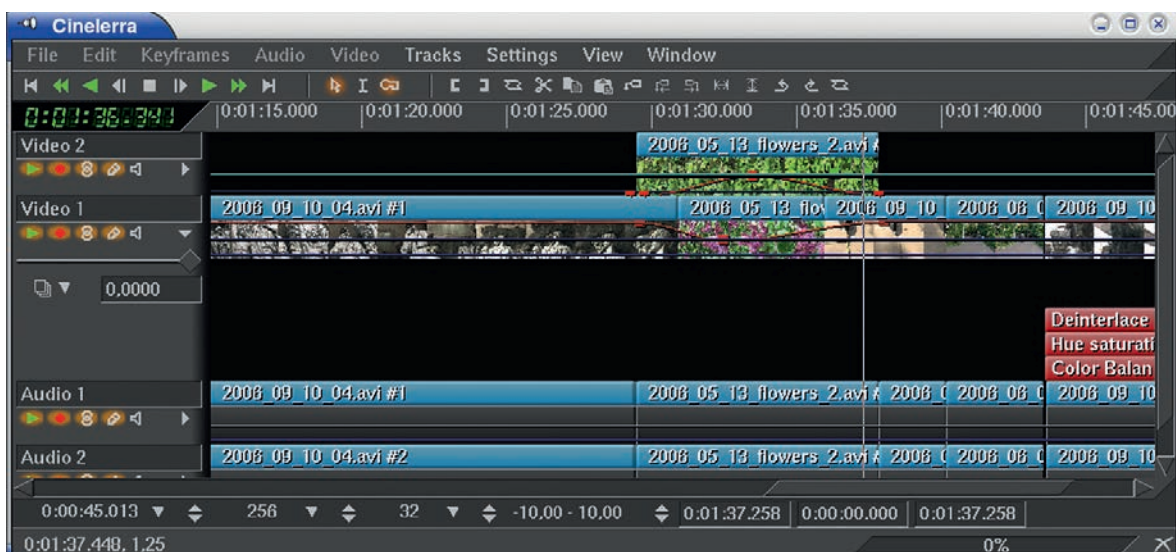


Cinelerra

Существует две версии *Cinelerra* — оригинальная от разработчика Джека Кроссфайра [Jack Crossfire] (он же Адам Уильямс [Adam Williams], он же Heroine Warrior/Heroine Virtual Ltd.) и так называемая «версия сообщества», ответвление [fork] — *Cinelerra CV* (lumiera.org). Сайт Кроссфайра сохранился только как устаревшее зеркало на SourceForge (sourceforge.net/projects/heroines/files/). Версия от Джека шагнула в нумерации дальше, чем *CV*. Существует взаимосвязь между форком и оригиналом, в сторону первого, ибо *CV*-версия вбирает чужие наработки охотнее, нежели это делает со своим кодом Джек Кроссфайр.

Cinelerra выросла из другого проекта того же Кроссфайра, *Broadcast 2000*. Я работал с *Broadcast 2000*, а потом перешел

» **Lives в режиме монтажного стола.**



» **Таймлайн в Cinelerra.**

» **Не хотите пропустить номер?** Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

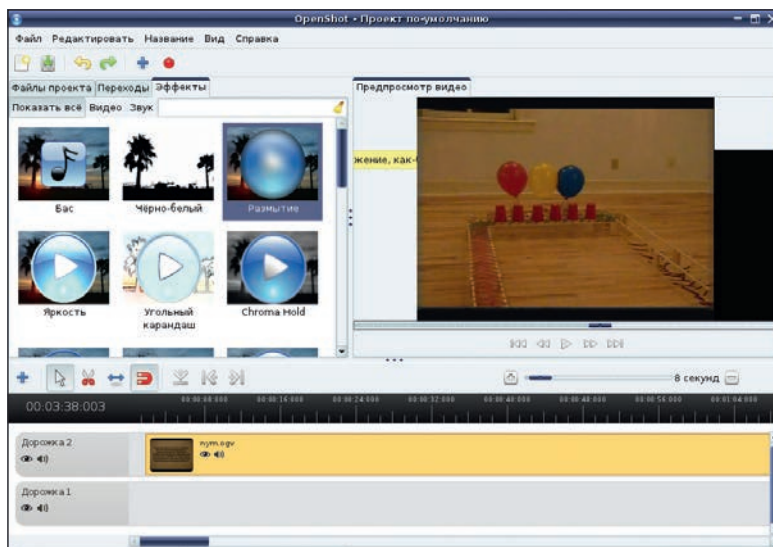
на первые версии *Cinelerra* и застал ее расцвет. В сообществе пользователей *Cinelerra* до сих пор в ходу написанная мною вспомогательная утилита *SevenGnomes*, служащая для удобной подготовки анимационных последовательностей из набора статичных изображений и для многих других вещей.

Ядро *Cinelerra* стоит на уровне коммерческих NLE, а в чем-то и превосходит их. *Cinelerra* сочетает в себе функции NLE и средства для композитинга, причем весьма развитые. Например, поддерживается работа с маской произвольной формы, то есть без всяких синих экранов вы можете отметить в кадре область, а все изображение вне области станет прозрачным, и за ним отобразится лежащая на соседней дорожке видеоподложка. В *Cinelerra* реализована интересная концепция камеры и проектора, позволяющая управлять масштабированием и панорамой. Параметры эффектов *Cinelerra*, а также камера и проектор управляются динамически при помощи механизма автоматизации, то есть распределенного по времени изменения параметров. Для многих функций, входящих в состав *Cinelerra*, в мире коммерческого ПО пришлось бы купить несколько отдельных программ-тяжеловесов. Кроме прочего, *Cinelerra* обладает широкими возможностями обработки звука, в том числе и его записи, что отличает *Cinelerra* от большинства других NLE под Linux, которые умеют записывать видео и звук с камеры и импортировать аудиовизуальные файлы, но не способны записать отдельно звуковую дорожку с микрофона или линейного входа.

К сожалению, с течением времени сборка *Cinelerra* стала требовать слишком много усилий. Сейчас во многих дистрибутивах либо вообще не делают «официальную» сборку *Cinelerra*, либо вынуждены применять множество патчей и прибегать к ухищрениям — то версия компилятора не подходит, то приходится определенные функции *Cinelerra* вообще отключить, потому что нужные для них библиотеки нельзя откомпилировать, и так далее. Кажется, что нормально, без шаманства, *Cinelerra* собирается только на компьютерах ее разработчиков.

К интерфейсу *Cinelerra* можно привыкнуть, но нельзя пользоваться программой, которая компилируется через пень-колоду (если вообще компилируется). Во многом тому виной был целый арсенал сторонних библиотек, таскаемых в исходнике *Cinelerra*.

» Так выглядит OpenShot.



«Поговорим о том, что в Linux можно использовать на деле.»

Все в одном — хорошо, но код имеет обыкновение устаревать, и если его никто не поддерживает, он перестает компилироваться. Версия сообщества — *Cinelerra CV* — была попыткой привести *Cinelerra* в лучшую (в плане простоты сборки, а значит, переносимости) форму. Итогом этой попытки стала идея радикально переписать часть кода — что получило название *Cinelerra3*, после чего разработчики поняли, что проще начать писать код с нуля, чем переделывать текущий.

Так родился еще один проект — *Lumiera* (lumiera.org). Команда *Lumiera* создала огромный сайт, мощную инфраструктуру по коллективной работе над кодом, активно освещала свою деятельность на различных мероприятиях, посвященных FOSS, но — на последнем таком мероприятии ребята из *Lumiera* рассказывали не о своей программе, а о том, как здорово использовать *Git*. По сути, провели мастер-класс по *Git*! Существует некая сборка *Lumiera* под Debian (пользователи вне команды *Lumiera* потерпели неудачу в компиляции), с прототипом графического интерфейса, один-два снимка экрана этого прототипа, идет медленная деятель-

ность в *Git*-хранилище, судя по всему, заключающаяся в поддержке уже написанных разрозненных частей, а не в создании нового кода, непосредственно относящегося к видеомонтажу. Исправляют какие-то тесты... Я не знаю, когда такими темпами нам будет показано что-нибудь работоспособное. Еще пара лет такой пассивной поддержки зачатков кода *Lumiera*, и оставшиеся разработчики совсем потеряют к нему интерес, если только постоянная адаптация этого кода к новым версиям внешних библиотек — не самоцель.

Видите, как все печально в области NLE под Linux? Но есть еще несколько монтажных программ.

PiTiVi

Например, *PiTiVi* (www.pitivi.org). Говорим *PiTiVi* — подразумеваем Gnome. А раз Гном, то первым делом на сайте нам сообщают, что программой легко пользоваться и ее интерфейс построен по принципам, заложенным в знаменитом руководстве Gnome Human Interface Guidelines. Это сообщение занимает львиную долю пространства главной странички сайта, а вот сведения о функциях *PiTiVi* помещены внизу, по ссылке. Очень зря, ибо при всей простоте в *PiTiVi* есть функции, выделяющие его на фоне других, более сложных и навороченных NLE.

Дело в удобной реализованной в *PiTiVi* механизме «оглашающих» — кривых, при помощи которых вы можете визуально управлять автоматизацией, в случае *PiTiVi* — прозрачности фрагментов видео и громкости звуковых фрагментов. Если бы автоматизация подлежала еще и эффектам, *PiTiVi* вышло бы на другой уровень, удовлетворив запросы и профессионалов. Увы... Автоматизацию эффектов планировали сделать в рамках Google Summer of Code 2011 — с тех пор ничего не изменилось. Если бы разработчики меньше надеялись на Google Summer of Code и писали код сами...

PiTiVi не обладает, в отличие от *Cinelerra*, собственным движком монтажа: вместо этого он использует Gnonlin, а в качестве библиотеки мультимедийной поддержки — *GStreamer* (Gnonlin также и часть *GStreamer*). *PiTiVi* — объединяющий эти технологии графический интерфейс, написанный на Python с использованием *GTK*.

Обзорная часть подходит к завершению. О *Lightworks* я мною писать не хочу, да и не о чем особо писать. Да, три года назад

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

EditShare, компания-разработчик *Lightworks*, собиралась открыть код своего продукта «с двадцатилетней историей» и выпустить версию для Linux. С тех пор много воды утекло, открытого кода нет, а двоичную бета-версию предлагают скачать после регистрации на сайте и принятия лицензии, которая кроме прочего обявляет вас, как участника бета-тестирования, общаться с сотрудниками компании EditShare. Да не хочу я с ними общаться, тем более что для скачивания предлагают Deb-пакет, протестированный на Ubuntu и его клонах, а у меня RPM-дистрибутив. Разработчики также пишут, что «Fedora is not a supported distribution» — а как насчет десятков других дистрибутивов? Очень странно писать код, который работает только в конкретном дистрибутиве Linux.

Текущая функциональность бета-версии не вызывает у меня воодушевления. Некогда я пробовал версию *Lightworks* для Windows. Двадцатилетняя история развития программы, с одной стороны, хорошо; с другой — концепция работы в *Lightworks* осталась той, конца 1980-х — начала 1990-х годов.

Даже если код откроют под приемлемой лицензией, то не повторит ли *Lightworks* судьбу *Xara Xtreme*? Многие помнят, как исходники этого векторно-растрового редактора были открыты, завертелась какая-то деятельность, а потом все затихло. Люди рисуют в *Inkscape*, а следы исходников на сайте *Xara Xtreme for Linux* (www.xaraxtreme.org) я найти не могу. Кажется, кто-то их видел, скачивал и пытался в них разобраться. Также на сайте висит ссылка на коммерческую версию *Xara* под Windows. Кстати, внимание многих линуксоидов было привлечено и к *Lightworks* подобным образом — на фоне обещаний Linux-версии предлагалось попробовать Windows-версию. Но исходный код *Xara Xtreme* так и был открыт...

Давайте лучше поговорим о том, что можно в Linux использовать на деле. Это два NLE на монтажном движке MLT (*Media Lovin' Toolkit*) — *OpenShot* и *Kdenlive*.

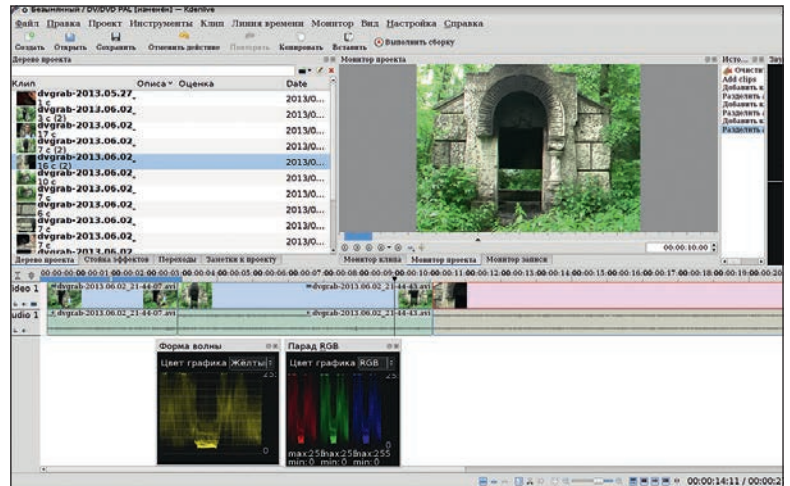
OpenShot

Об *OpenShot* (www.openshotvideo.com) сейчас писать сложно, ибо недавно главный его разработчик, Джонатан Томас [Jonathan Thomas], провел на Kickstarter'e кампанию по сбору средств на *OpenShot* нового поколения. Отказ от MLT в пользу некоего собственного «революционного движка», работа над которым, оказывается, идет уже два года. Кроме того — выбор для интерфейса инструментария Qt вместо прежнего GTK. Обо всем этом Томас говорит в будущем времени и, судя по всему, показать ничего рабочего пока не может, поскольку в новостях о грядущих изменениях предлагается посмотреть пластилиновый мультфильм с рекламой возможностей Qt5, созданный разработчиками Qt.

Текущая версия *OpenShot* — это редактор для простого монтажа, наложения эффектов и нехитрого композитинга. Очень базовая поддержка автоматизации — всего по двум ключевым кадрам (в начале и конце фрагмента видео) — свойств самого фрагмента, но не эффектов. Что будет дальше с этим «старым» *OpenShot*, сказать сложно; скорее всего, какое-то время будут выходить версии с исправлениями ошибок. Я понимаю разработчика *OpenShot* в его намерении уйти от использования движка MLT: дело в том, что весьма популярен еще один видеоредактор на этом движке — *Kdenlive*, который по функциональности превосходит *OpenShot*. О *Kdenlive* я и буду говорить подробно.

Kdenlive

Kdenlive (www.kdenlive.org) — один из немногих проектов, нацеленных на сложные монтажные работы. Грубо говоря, есть два уровня монтажа — бытовой (склейка нарезки из домашнего видео) и профессиональный (в равной мере относящийся к и любительскому независимому кино). На профессиональный монтаж ориентирована *Cinella*. На такой же уровень нацелен и *Kdenlive*, стремительно эволюционируя.



В *Kdenlive* можно выполнить монтаж такой же сложности, как в коммерческих NLE-гигантах — впрочем, затратив на монтаж больше времени. И дело тормозится больше особенностями (вначале я написал «недоработками») интерфейса, нежели технологически. Об этом мы еще поговорим в следующей части статьи.

Как явствует из названия, *Kdenlive* привязан к рабочей среде KDE — хотя, конечно, никто не мешает вам запускать его из другого окружения. *Kdenlive* написан на C++ и использует ряд сторонних движков, в частности, MLT для монтажа (часть разработчиков *Kdenlive* — одновременно и разработчики MLT), *FFmpeg* для поддержки различных мультимедийных форматов — и от сборки *FFmpeg* будет зависеть, допустим, сможете ли вы отрендерить свое видео в MPEG4/XVid или нет. Как и *OpenShot*, *Kdenlive* оснащен встроенными эффектами, и *Kdenlive* — очень активно развивающийся проект, вокруг которого сплотилось и не менее активное сообщество пользователей: идет постоянный обмен мнениями, внесение предложений, нахождение и исправление ошибок. Механизмы KDE позволяют также обмениваться по сети различными профилями настроек, шаблонами, эффектами переходов — не выходя из интерфейса программы.

Двоичная сборка *Kdenlive* включена в большинство дистрибутивов Linux. А для компиляции вручную на сайте MLT даже предлагается особый скрипт (<http://www.mltframework.org/bin/view/MLT/BuildScripts>), который выполняет титанический труд — скачивает *lame*, *libvpx*, *libx264*, *FFmpeg*, *Frei0r*, MLT и *Kdenlive*, а затем пытается их компилировать и установить в локальный каталог, т.е. вы получаете полноценный набор всех нужных кодеков, эффектов и *Kdenlive* — но без общесистемной установки, что весьма удобно, если вы не хотите трогать системный *FFmpeg* и тому подобное.

Что до традиционной сборки «ручками», то мой опыт в этом был отрицательным. Удалив прежнюю версию *Kdenlive* и некоторые сопутствующие ей пакеты, я скачал нужные исходники и все последовательно успешно откомпилировал и установил. Все зависимости были соблюдены и удовлетворены. Но когда я попытался запустить собранный таким образом *Kdenlive*, часть его модулей не работала, а сама программа слишком часто вылетала. Недавно я поставил себе новую, третью версию своего рабочего дистрибутива Mageia, в ней новая версия *Kdenlive* уже в сборке, работает хорошо.

Летом нынешнего года в исходнике *Kdenlive* происходят значительные изменения по вливанию в код изменений от Тилла Зито [Till Theato], поэтому до осени ждать стабильной новой версии не следует. Кроме того, *Kdenlive* научится работать с библиотекой *Movie*, просчитывающей видеоэффекты при помощи GPU.

В нашем же учебнике, в следующей его части, мы поговорим о том, что умеет текущая версия *Kdenlive*, и внимем в ее практическое использование. **LXF**

➤ Главное окно *Kdenlive*.

Blender: Внедрим

Андрей Прахов рассуждает о практике и теории создания моделей для игр.



Наш эксперт

Андрей Прахов
Участник нескольких игровых проектов, представитель СМИ, автор первой русской книги по Blender: «Blender. 3D-моделирование и анимация. Руководство для начинающих».



Как-то раз в автобусе мне довелось присутствовать при разговоре тинейджеров, которые бурно обсуждали вопросы создания игр. А так как эта тема мне близка и интересна, то я невольно прислушался к весьма содержательному диалогу. Ребята сыпали терминами и проявляли недюжинные знания в области игростроя. Разговор шел об оптимальном выборе программного инструментария для создания трехмерного игрового персонажа со всеми положенными для него атрибутами. Юные вундеркинды перебрали, наверное, все популярные приложения в этой области, включая узкоспециализированные плагины для них. Но когда возник вопрос о законности использования пиратского софта, на «плаву» остался лишь *GMAX* (бесплатная, урезанная версия *3Ds MAX*). «А почему не *Blender*? — Да говорят, он сложный. — А он вообще годится для этого?»... Приехали.

Действительно, какие плюсы имеет *Blender* по сравнению с популярными проприетарными пакетами? Оставим в стороне набившую оскомину бесплатность и обратимся к его возможностям именно для создания игрового 3D. Под этим понимается не только трехмерная модель, но и текстуры, а также анимация.

Работа над моделью разбивается на несколько этапов:

1 Создание базового объекта и его высокополигональной копии
Существуют разные подходы к реализации этого шага. Например, кто-то предпочитает на основе низкополигонального объекта разрабатывать высокополигональный, а кто-то — наоборот. В любом случае *Blender* предлагает удобные инструменты для реализации этого этапа.

2 Создание текстур
В современной игре модель обертывается несколькими видами текстур: это основная, которая несет в себе цветовую окраску модели (*Diffuse map*); карта выдавливания

для имитации мелких деталей (*bump/normal/parallax map*); карта освещенности (*Lightmap map*); карта высот (*Displacement map*). Конечно, многое зависит от технического задания, и некоторые виды текстур бывают не нужны, а иногда какие-то специально объединяют в одну. Допустим, основная (*Diffuse*) плюс освещение (*Ambient occlusion*). В итоге такая модель прекрасно выглядит даже без участия источника света, что положительно сказывается на маломощных платформах — допустим, мобильных. Бывает необходимость в дополнительной окраске вершин объекта. К примеру, можно написать программный код в игре, который будет учитывать цвет вершин модели и деформировать ее (обычно используют черно-белый градиент). Так вот, все перечисленные карты прекрасно создаются в *Blender*, без привлечения дополнительного ПО. А если в наличие имеется графический планшет, то и основная окраска модели может осуществляться на профессиональном уровне. Особенно удобно рисовать непосредственно на имеющемся трехмерном объекте, а не раскрашивать текстуру в графическом редакторе.

3 Создание анимации
Традиционно в играх используют три вида анимации: объектная, скелетная и вершинная. Первая выполняется исключительно с помощью программного кода игрового движка, и для ее реализации трехмерные редакторы не нужны. А вот остальные подготавливаются только в них. Скелетная анимация — это использование вспомогательных объектов для движения частей модели. Фактически художник создает каркас, который впоследствии и «оживляется», а уже к нему прикрепляется сама модель. В итоге трехмерный редактор должен иметь инструменты не только для создания и анимирования скелета, но и позволять качественно настраивать деформацию объекта в местах сгибов. Вершинная анимация (также ее называют морфинг) обычно используется для создания «мелкого» движения. Например, с ее помощью можно передавать эмоции на лице персонажа. Фактически в трехмерном редакторе создаются ключевые положения элементов объекта [*morph target*], а уже в движке программно осуществляются переходы между ними. Разумеется, в *Blender* имеются инструменты и для этих задач.

4 Финальная подготовка и экспорт
Немаловажный этап, когда готовая модель с анимацией должна быть сохранена в формате, который подходит для игрового движка, а их великое множество! В *Blender* имеется несколько десятков экспортеров для разных случаев жизни — популярный Autodesk FBX, Wavefront (OBJ), 3D Studio (3DS) и так далее. Однако некоторые движки работают исключительно только с собственным форматом и предлагают соответствующие плагины для популярных трехмерных пакетов. В этом списке *Blender* всегда присутствует.

Вот и получается, что *Blender* вполне подходит для реализации каждого этапа работы над игровой моделью. И это не так сложно, как кажется! Давайте рассмотрим практическую сторону.

Основа основ

Дом начинается с кирпича; то же самое можно сказать о трехмерной модели, ведь в основании ее лежит примитив — простой геометрический объект. Таковых в программе несколько десятков, равно как и способов создания моделей. Все, конечно,

В ИГРОСТРОЙ

Базовый элемент

Blender предлагает большое количество примитивов для создания моделей. Все они расположены в главном меню программы Add и сгруппированы по типу: Mesh, Curve, Surface, Metaball. Это сделано не зря, ведь каждый блок объектов имеет свои характеристики, способы редактирования и предназначения. В целом их можно условно разделить на две большие группы по способу функционирования: статичные (Mesh) и математические (Curve, Surface, Metaball). Так, Mesh — это массив элементов (вершины, ребра, плоскости) с заданными координатами, а остальные — вычисляемые математические функции. Вот и получается, что Mesh-примитивы позволяют спокойно манипулировать буквально каждой вершиной, зато математические удобно использовать для создания объектов с плавными изгибами.

зависит от поставленной задачи; так, например, объект обтекаемой формы (ложка, подводная лодка и даже персонаж) удобно создавать при помощи кривых NURBS, а оружие, растения, инвентарь — с помощью Mesh, но нужно учитывать общую специфику игровой модели: в итоге она должна быть низкополигональной, т.е. содержать минимально необходимое количество элементов. Моделлеру приходится балансировать между качеством и количеством, но многое «исправляется» с помощью визуальных технологий Normal map.

Задача этого этапа — создание базовой модели, которая впоследствии будет текстурирована и анимирована. Причем даже не одной, а целых двух, если, конечно, требуется карта Normal map. В этом случае помимо основной низкополигональной модели необходимо создать высокополигональную, которая будет иметь множество мелких деталей.

Обратите внимание на рисунки, где изображена модель червяка. Рис. 1, а демонстрирует готовую низкополигональную модель, состоящую примерно из 260 вершин, а рядом расположена

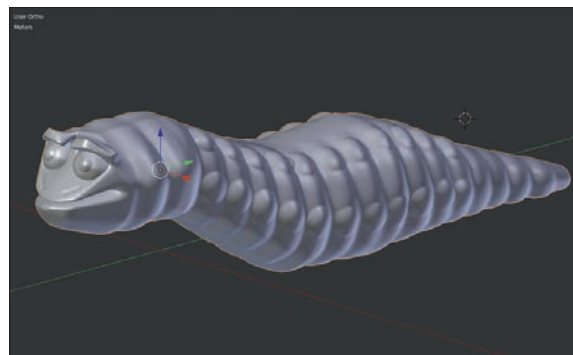
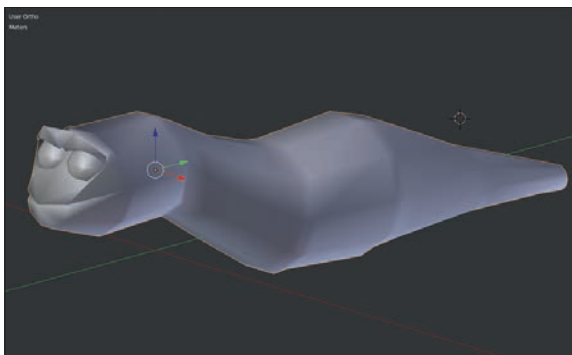
ее высокополигональная копия, но уже из 270000 вершин. Такое количество элементов позволяет создавать более детализированный объект. В дальнейшем разница между объектами и используется для просчета карты Normal map.

Возникает вопрос: какой же примитив выбрать для своей модели? Традиционно для создания игровых моделей используются примитивы группы Mesh, и это не случайно. Во-первых, моделлер может контролировать положение каждой вершины, равно как их количество, что является основным фактором в выборе «кирпичика» для низкополигонального объекта. Во-вторых, *Blender* предлагает специальные механизмы, упрощающие разработку. К таким относятся симметричное моделирование, изменение детализации «на лету», простота привязки к костям скелета и многое другое.

Группа Mesh содержит 8 простых объектов, от плоскости до сферы. Здесь все зависит от основной формы будущего объекта и предпочтений художника. Так, червяк (см. рис. 1) был создан на основе примитива Circle [Окружность], и это не случайно, ведь он цилиндрической формы. Но можно было бы использовать другие примитивы: Tube [Туба], Cone [Конус] и даже Cube [Куб]. Инструменты редактирования Mesh настолько удобны и многочисленны, что не составляет труда изменить базовую форму объекта, изначально, казалось бы, не соответствующего требованиям.

Несмотря на сложность формы конечного объекта, принцип моделирования прост — манипулирование и наращивание элементов примитива. Рассмотрим основные этапы этого процесса применительно к вышеуказанной модели червяка:

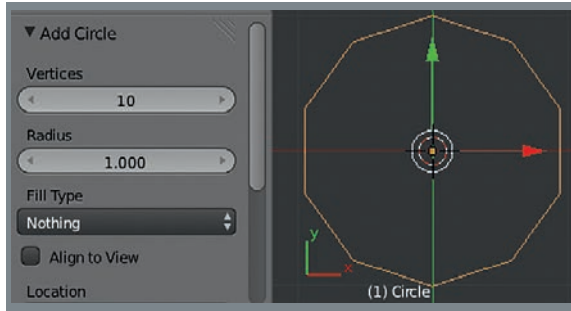
1 Создание примитива Circle с минимально необходимым количеством вершин и заливкой Здесь нужно учитывать тот момент, что *Blender* позволяет редактировать количество элементов примитива сразу после его добавления в сцену (рис. 2). Все необходимые настройки находятся в левой панели Tool Shelf. Увеличивая или уменьшая параметр Vertices [Вершины], можно визуально добиться нужного баланса. Запомните простое правило: всегда проще добавить элементы объекту, чем безболезненно удалить их. »



» Рис. 1. а) Низкополигональная модель червяка; б) ее высокополигональный эквивалент.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

➤ Рис. 2. Панель настроек примитива на этапе его создания.



2 **Операция Extrude [Выдавливание]** Ее используют для дублирования выделения без разрыва связей с исходными элементами. Так, для рассматриваемой модели было выполнено выдавливание всех вершин примитива и перемещение новых по одной из осей (рис. 3, а).

3 **Создание дополнительных элементов** В *Blender* имеется несколько инструментов для разбиения структуры объекта на более мелкие составляющие — это использование модификаторов, инструментов Subdivide [Подразделение] и Loop Cut and Slide [Вырезать и Сдвинуть]. Модификаторы рассмотрим чуть позже, а пока обратите внимание на рис. 3 (б, в). В первом случае это результат работы Subdivide. Как видите, инструмент выполнил разбивку каждого выделенного полигона. В итоге — 360 вершин и чрезвычайно узкая сетка между сегментами. А вот второй вариант просто добавил дополнительные ребра, и это всего 60 вершин. Таким образом, Loop Cut and Slide идеально подходит для поставленной задачи.

4 **Придание формы объекту** Червяк — очень простая модель. По сути, нужная форма достигается путем масштабирования сегментов. Только предварительно необходимо добавить достаточное количество ребер.

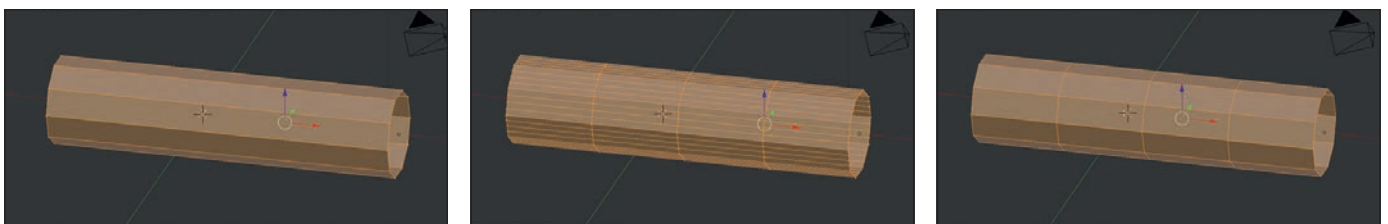
Рассмотренный пример моделирования очень прост и содержит базовые манипуляции. Но, согласитесь, и готовая модель — не образец сложности. Другое дело...

*Точка, точка, запятая,
Минус, рожица кривая,
Палка, палка, огуречик —
Вот и вышел человечек.*

Обычная детская считалочка, знакомая всем, но в ней кроется особенность строения человека, да и почти любого живого организма — симметрия. И правда, зачем моделировать две руки, если можно создать одну и зазеркалить ее. Именно этот принцип положен в основу симметричного моделирования.

Для успешной работы с симметрией нужно учитывать ряд требований. Самое главное — правильная подготовка примитива. В качестве последнего можно использовать любой из группы Mesh, но обычно предпочитают Cube [Куб]. В режиме редактирования к кубу применяют Subdivide или Loop Cut and Slide. Это делается для точного деления его пополам. Затем удаляется одна из половинок объекта и выбирается модификатор Mirror [Зеркало].

➤ Рис. 3. а) Объект после Extrude; б) результат двойного применения Subdivide; в) результат работы Loop Cut and Slide.



Интерфейс Blender

Опытные пользователи, работая с программой, традиционно отдают предпочтение «горячим» клавишам. И этот путь является сильной стороной *Blender*. Концепция интерфейса такова, что максимальная эффективность проявляется только при использовании данного способа. В то же время *Blender* предлагает другие варианты доступа к нужным функциям — это привычное всем оконное меню; панель Tool Shelf, содержащая доступные в текущем режиме функции (расположена слева); плавающее «умное» меню. Для использования последнего достаточно нажать клавишу пробела, затем ввести несколько букв в появившемся окне — и программа предложит доступные варианты команд.

На рис. 5 показан объект, у которого удалена одна половина и добавлен модификатор. При этом манипулирование вершинами исходной сетки в точности отражается на другой стороне. Такой способ редактирования несомненно облегчает жизнь моделлеру.

Модификатор Mirror — очень простой инструмент, но неаккуратное его использование может здорово усложнить работу на следующем этапе — высокополигональном моделировании. Дело в том, что его желательно применить (окончательно закрепить результат его работы) до использования инструментов скульптурной лепки. Если этого не сделать, в готовой модели могут появиться артефакты — «ложные ребра», неправильное сглаживание, разрывы сетки и т. д.

Особо следует опасаться удвоения элементов в оси зеркалирования объекта. Лучшим средством от этой напасти служит удаление всех ненужных центральных вершин модели или использование опции Merge (см. рис. 5), которая объединяет близко находящиеся вершины.

Перед окончательным применением модификатора и переходом к следующей части работы нужно убедиться в корректном построении сетки. Алгоритм перехода простой:

- 1** Применяем модификатор.
- 2** Выполняем операцию Remove Doubles [Удаление дублей]. Программа попытается отыскать некорректные вершины и удалить их, но особо рассчитывать на интеллект этой функции не стоит.
- 3** Добавляем сглаживание объекта (функция Smooth в панели Tool Shelf). Необходимо тщательно осмотреть модель со всех сторон. Обычно эффект сглаживания явно демонстрирует подозрительные места. Они могут выглядеть как некорректные тени, складки или разрывы.
- 4** Добавляем модификатор Multiresolution [Мультиразрешение], увеличиваем разрешение сетки с его помощью и проверяем ее на наличие артефактов.
- 5** Если на любом из этапов проверки есть подозрительные места, нужно отменить все сделанное и исправить ошибки.

Чуть выше прозвучало название еще одного модификатора — Multiresolution. Это обязательный инструмент, используемый для создания высокополигональной модели.

➤ **Пропустили номер?** Узнайте на с. 108, как получить его прямо сейчас.

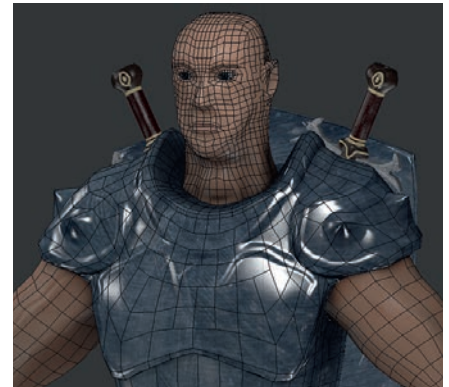
Эталон красоты или практичность?

В принципе, существует некий эталон «идеальной» модели. Она должна быть цельной и состоять из геометрически правильных элементов. Отлично получится, если средний размер полигонов будет одинаковым, без резких перепадов. Вот только насколько это необходимо — ведь при таких требованиях моделирование заметно усложняется?

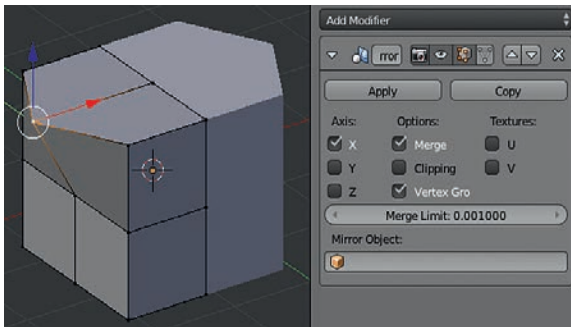
Посмотрите на рис. 4, где изображена модель воина. При ее создании как раз и учитывались эталонные требования. Достаточно плотная сетка в 6000 вершин позволила создать вполне детальную модель. К тому же тело воина, включая мелкие детали, состоит из одного объекта. Однако не всегда имеется возможность работать с большим количеством вершин. Чаще всего требуется уложиться в технический минимум, и вот тут уже не до красоты.

Представьте, к примеру, что нужно сделать лезвие для модели меча. Кажется, что может быть проще — вытянуть треугольник и добавить еще парочку вершин для объема. Но по сравнению с полигонами рукояти элемент лезвия будет резко бросаться в глаза. Конечно, пара дополнительных ребер исправит этот недостаток и добавит столь приятные профессионалам аккуратные полигоны, но и лишние вершины также появятся.

Когда-то, на заре трехмерных игр, подобное требование было не лишено смысла, но сейчас даже маломощные мобильные системы прекрасно справляются с любым построением сетки. Часто создатели мобильного контента используют различного размера полигоны и следят только за количеством вершин. Так что красота или практичность, а может быть, баланс обеих, остается на совести моделлера.

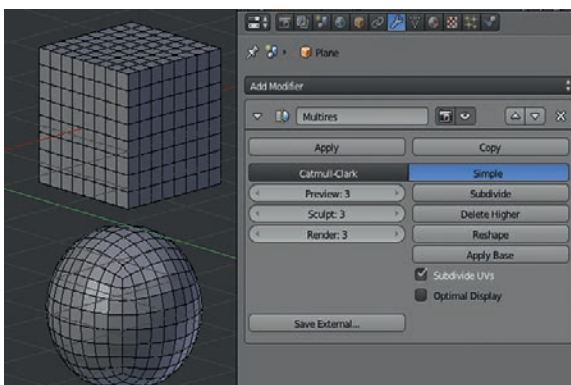


➤ Рис. 4. Эта модель может служить образцом «идеальной» сетки.



➤ Рис. 5. Деформированный куб и настройки модификатора Mirror.

Как вы, наверное, помните, такая модель необходима для генерации некоторых игровых текстур, например Normal map. Тщательная проработка этого этапа — залог качественной работы. И дело даже не в том, чтобы вылепить на объекте все мельчайшие детали. В первую очередь обе модели должны дополнять друг друга и не иметь резких различий базовой формы. Например, если художник создал низкополигональную модель телевизора, а в ее высокополигональной копии добавил антенну, то программа не сможет создать корректную текстуру Normal map. Другое дело, когда базовый объект уже имеет грубые заготовки для последующей более тонкой обработки. Сравните рисунки 1, а и 1, б, и вы увидите между ними много общего. В связи с этим создание высокополигональной модели для генерации Normal map (да и других карт) имеет ряд специфических приемов.



➤ Рис. 6. Как вы думаете, какая начальная форма была у сферы?

Если вы задумали сделать копию базового объекта для использования его в скульптурной лепке, то стоит отказаться от этой мысли, по одной простой причине: в дальнейшем придется создавать развертку текстурных координат, и это лучше делать только для одного объекта.

Алгоритм действий простой:

- 1 К низкополигональной модели добавляется модификатор Multiresolution, и тем самым объект становится высокополигональным.
- 2 Создается развертка текстурных координат (это мы рассмотрим в следующей статье).
- 3 Дублируется объект, и в одной из копий удаляется модификатор Multiresolution.

После таких нехитрых манипуляций останутся две модели — низкополигональная и высокополигональная, но уже с одной общей текстурной разверткой.

Теперь посмотрите на рис. 6, где изображены два разных объекта. На самом деле изначальная форма у обоих — куб. Получившаяся сфера — это результат работы модификатора Multiresolution.

По сути, данный инструмент выполняет ту же работу, что и функция Subdivide. Вот только делается это несравненно более удобно. Моделлер может выбирать подходящее для себя разрешение объекта (увеличивать или уменьшать количество полигонов), причем без потери ранее выполненных манипуляций. Но самое главное, результат разбивки зависит от выбранной математической функции. Всего их две (см. рис. 6): Catmull-Clark (стоит по умолчанию) и Simple. В первом случае вместе с разбивкой выполняется сглаживание объекта (сфера на рис. 6), во втором — только меняется разрешение и сохраняется изначальная форма объекта (куб на рис. 6).

Итак, для создания игровой высокополигональной модели нужно обязательно использовать функцию Simple модификатора Multiresolution. А вот сглаживание острых углов или ненужных складок будет выполняться уже инструментами скульптурной лепки. **1x3**

Анонс

На следующем уроке мы поговорим о приемах работы в режиме скульптуры; также раскроются некоторые секреты генерации игровых текстур.

ОТВЕТЫ

Есть вопрос по открытому ПО? Пишите нам по адресу answers@linuxformat.ru, и мы найдем ответ.

В этом месяце мы ответим на вопросы про...

- 1 Создание образа диска с разделами
- 2 Установку драйверов на ноутбук
- 3 Изменение размеров разделов
- 4 Системную информацию
- 5 Переход с HDD на SSD
- 6 Проверку зависимостей

1 Образ диска с разделами

В журнале за март 2013 года была статья Дэвида Хейварда «Виртуализируйте свою машину» [Учебники, LXF168], и в ней он показывал, как создавать образ жесткого диска. В своем примере он включал в образ весь диск, поэтому указывал `if=/dev/sda`. Статья мне очень понравилась, и я захотел такое попробовать. Но у меня на диске две операционные системы. И я хотел бы создать образ с разделами `sda1`, `sda2` и `sda3`.

Как это сделать? Я обнаружил, что в команде `dd` нельзя указывать больше одного значения `if=`. Подумал о подцеплении файла с `if`, но не уверен, что это сработает. Буду очень благодарен за любую подсказку.

Боб [Bob]

Для *VirtualBox* нужен виртуальный диск, а образы `sda1`, `sda2` и `sda3` — это три образа файловых систем. Подцепление не поможет, потому что тогда у Вас получится большая склейка из файловых систем без таблицы разделов, которая говорила бы компьютеру, где начинается и заканчивается каждый раздел.

Самый простой вариант — последовать совету статьи и сделать образ всего диска. На нем будет таблица разделов, из которой ПО виртуальной машины узнает, где что находится, а также загрузчик (при простом копировании всех разделов его тоже не хватало бы). На нем также будет другая операционная система. Если это тоже Linux, Вы по-прежнему сможете загружать ее. Если Windows — то, скорее всего, нет, потому что по Врожденной Неуклюжести (думаю, так это называется) Windows определит виртуальную машину как другое «железо» и потребует аутентификации. В Windows также может не хватать драйверов для виртуальных устройств, хотя в большинстве программ для виртуализации устройства эмулируются на самом базовом уровне.

Основная проблема с созданием образа всего диска — место, которое он занимает. С `dd` Вы получите побайтовую копию всего диска, поэтому

файл образа всегда точно такого же размера, как и весь диск. Зато образы VDI занимают ровно столько места, сколько им нужно. Когда Вы добавляете данные, файл образа увеличивается, чтобы вместить их. Когда Вы данные удаляете, размер файла образа не меняется, потому что восстановление размера диска замедлит виртуальную машину; к тому же Вы, весьма вероятно, скоро снова воспользуетесь этим пространством. В случае с разделом Windows такого не произойдет, поэтому после удаления данных можно сжать образ.

Виртуальный диск можно сжать с помощью *VBoxManage*, но тот удаляет только области, состоящие из нулей. Зайдите в Linux на виртуальной системе, откройте терминал и заполните раздел Windows нулями, скомандовав

```
dd if=/dev/zero of=/dev/sdaN bs=1M
```

Здесь `sdaN` — раздел с Windows. Затем можно удалить раздел Windows, выключить виртуальную машину и запустить следующую команду в терминале в основной системе:

```
VBoxManage modifyhd --compact /path/to/diskimage.vdi
```

Она удалит все пространство, заполненное нулями, и у Вас получится образ виртуального диска, содержащий только разделы с Linux исходного диска.

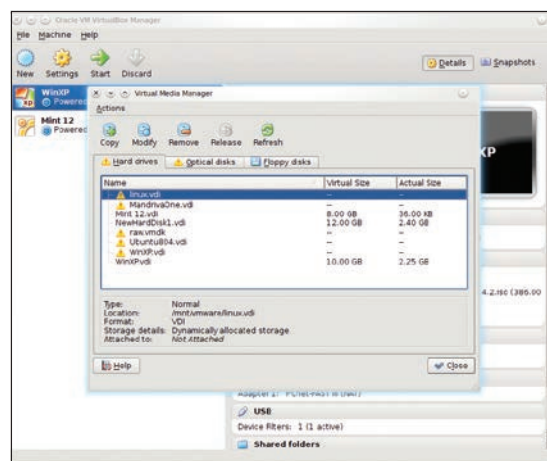
2 Откуда дровишки

Я новичок в Linux и совсем недавно набрел на ваш журнал. Отличные статьи, кстати! Я установил BackTrack 5 на свой ноутбук Dell Studio 17 и хотел установить драйверы устройств, но не могу понять, как это сделать. Не могли бы вы подсказать наилучший способ?

Адам Винчевски [Adam Winczewski]

Если коротко — такого способа нет. Linux сильно отличается от Windows во многих аспектах. Два самых очевидных — получение драйверов и установка программ. В ядре Linux есть огромное количество готовых драйверов, и подавляющее большинство устройств просто работает без дополнительного ПО. Вы немного усложнили дело, не сообщив нам, для какого устройства нужны драйверы, но драйверы для проводной и беспроводной сети в ядре есть, как и драйвера для web-камеры. Выполните следующую команду в терминале:

```
sudo ifconfig -a
```



Образ диска в *VirtualBox* — это диск целиком, а не просто склейка одиночных разделов.

Она должна вывести `eth0` для проводной сети и `wlan0` для беспроводной. Если какого-то из этих интерфейсов нет, то автоопределение не сработало, и Вам нужно загрузить драйвера вручную. Модули, которые Вам нужны — `tg3` для порта Ethernet и `iwlwifi` для беспроводной сети. Добавьте необходимые модули в конец файла `/etc/modules`, по одному на строку. В этом файле задаются модули, загружаемые при запуске компьютера. Чтобы включить их, не нужно перезагружать программу, просто выполните команду:

```
sudo modprobe tg3
```

или

```
sudo modprobe iwlwifi
```

Для драйвера `iwlwifi` также необходим код прошивки, но по умолчанию он уже установлен в BackTrack. Web-камера использует стандартный протокол UVC (класс устройств USB-видео). Не копитесь на название — протокол используется также и для встроенных web-камер. Если драйвер `uvcvideo` не загружен, загрузите его принудительно, как мы делали ранее. Убедитесь, что файл `/dev/video0` существует. Так программа работает с web-камерой.

Установка драйверов — только часть решения. Еще Вам понадобится программа для работы с устройством. Для сети это *Network Manager*, он установлен по умолчанию. Для web-камер программ много, но популярный вариант — *Cheese*. Для установки программ в BackTrack или любом другом дистрибутиве Linux используется менеджер пакетов дистрибутива. В BackTrack это *KPackageKit* или *Synaptic*, в зависимости от того, пользуетесь Вы KDE или Gnome. Сами программы находятся в репозиториях, поддерживаемых дистрибутивами, то есть все программы

проверяются на совместимость, безопасность и отсутствие вредоносного кода (в отличие от программ в Windows, которые загружаются со случайных сайтов). Наберите название программы (или ее тип) в строке поиска менеджера пакетов, выберите нужные результаты и нажмите Install [Установить]. На этом все! А менеджер пакетов сообщит Вам и об обновлениях. Это и вправду так просто — добро пожаловать в Linux!

3 VM и LVM

В После недавних проблем с двойной загрузкой openSUSE и Windows 7 — проблем, вызванных, похоже, мной — я решил запустить Windows на виртуальной машине. Windows у меня на компьютере HP была предустановлена, поэтому DVD с Windows у меня нет. Можно ли установить Windows на виртуальную машину без DVD?

Windows установлена на первых два первичных раздела, а на третьем первичном разделе — раздел восстановления Windows (Windows Recovery). Я никогда не понимал, зачем Windows два раздела. Четвертый первичный раздел — openSUSE. На всякий случай я создал диски восстановления.

Раз уж мне подавай луну с неба... а можно ли это сделать, не трогая openSUSE, восстановить место, использованное тремя разделами Windows, и одновременно переключиться на LVM?

Грег Джонс [Greg Jones]

О С копированием установленной системы Windows на виртуальную машину есть две проблемы: юридическая и практическая. Юридическая в том, что Ваша лицензия скорее всего не позволяет этого. Лицензии Windows OEM обычно предназначены только для одного набора устройств. Следовательно, они содержат мало драйверов и не будут работать на других устройствах. Даже несмотря на то, что Вы запускаете Windows на том же компьютере, она работает внутри виртуальной машины, в которой используются другие устройства.

При этом также понадобится снова аутентифицировать Windows, если Вы сможете это сделать. Самый безопасный вариант, если Вам нужна Windows — начать снова с обычным компакт-диск Windows.

Луна с неба — дело хорошее, но последний вопрос — это уже чересчур! После удаления трех разделов Windows можно переразбить диск для LVM, но перенести существующую установку будет непросто, потому что у Вас все на одном разделе. При загрузке через LVM используется отдельный раздел /boot для хранения ядра и кода, необходимого для инициализации логических томов. С LVM можно только снова установить openSUSE на место, где раньше были три раздела Windows, оставив старую систему нетронутой.

Потом Вы сможете загрузиться в нее, смонтировать файловую систему прежней SUSE и скопировать в нее свой домашний каталог и все настройки из /etc и данные из /var или /www, которые Вам могут понадобиться. Также можно получить список пакетов, установленных в старой системе, загрузившись в нее и выполнив команду

```
rpm --query --all
```

Перенаправьте ее вывод на файл или на принтер, чтобы получить копию — так Вы сможете убедиться, что в новой системе установлено все необходимое. Покончив со старой системой, можно удалить содержащий ее раздел и увеличить размер физического тома, используемого LVM, до размера остальной части диска. Безопаснее, хотя и не обязательно, делать это с Live CD. Затем нужно сообщить LVM об изменении размера, запустив команду:

```
sudo pvresize /dev/sdaN
```

Теперь команда **pvdisplay** должна выводить новый размер, а **vgdisplay** — показывать много свободного места в группе томов. Изменение размера группы томов не влияет ни на один из логических томов, и это пространство можно выделить любому из существующих логических томов или создавать новые по мере необходимости.

Терминалы и суперпользователи

Мы часто предлагаем в качестве решения проблемы ввести те или иные команды в терминале. Хотя обычно то же самое можно сделать с помощью графических утилит дистрибутива, такие решения будут слишком конкретными (будут зависеть от дистрибутива). Команды в терминале более гибкие и — самое главное — ими можно пользоваться во всех дистрибутивах. Команды настройки системы часто нужно выполнять от имени суперпользователя, называемого также root. Существует два основных способа это делать, в зависимости от используемого дистрибутива. Во многих дистрибутивах, особенно в Ubuntu и его производных, перед командой можно написать **sudo** — при этом будет запрошен пароль пользователя, и ему будут предоставлены привилегии root только на время выполнения команды. В других дистрибутивах применяется команда **su**, для использования которой требуется ввести пароль root и которая предоставляет полный доступ root до того момента, пока вы не наберете **logout**. Если в вашем дистрибутиве используется **su**, запустите ее один раз и выполняйте любые заданные команды без предшествующей **sudo**.

4 Информация о системе

В У меня есть сервер Ubuntu 12.04, к которому я подключаюсь через SSH по локальной сети. Я почти уверен, что раньше при входе в систему сервер отображал сообщение о загрузке системы, пользователях, которые вошли в систему, IP-адресе и планируемых обновлениях и др.

Кажется, эта информация предоставлялась **Landscape**. Я установил его и вижу это сообщение, если запускаю **landscape-sysinfo** вручную, но как



Коротко про...

/proc

В ы когда-нибудь просматривали все свои файлы с целью понять, что занимает столько места на диске? Если да — возможно, вам попался каталог **proc** в корне файловой системы, и вы задавались вопросом, можно ли обойтись без тысяч файлов, которые кажутся пустыми, но занимают более 1 ГБ на диске.

Ответ — нет. Удалить их нельзя. Хорошая новость — они не занимают места, потому что /proc — это виртуальная файловая система. Это не набор настоящих файлов, а отражение со-

стояния системы, каким ее видит ядро, представленное в виде файловой системы, чтобы с ней было удобнее работать.

/proc состоит из огромного количества «файлов» (на моем компьютере сейчас таких 7109), каждый из которых содержит порцию информации. Например, наберите **cat /proc/version**, и вы узнаете больше о своем дистрибутиве и ядре, или **cat /proc/cpuinfo** — и вы получите подробную информацию о процессоре. Такие каталоги, как **ide**, **scsi**, **bus/usb**, **bus/pci** и **bluetooth**, содержат необходимые сведения о подключенных устройствах.

Возможно, вы также заметили каталоги с именами-числами. Каждый каталог соответствует выполняемому процессу, а его имя —

идентификатор процесса. В каждом из них вы найдете информацию о процессе, об использовании им памяти, об аргументах командной строки для его запуска и т.д.

Основная часть содержимого /proc доступна только для чтения, но есть несколько файлов, в которые разрешается запись. Например, файл **/proc/sys/vm/swappiness** определяет, насколько интенсивно система будет использовать раздел подкачки. Большие значения означают, что система прибегает к разделу более часто. Изменить это можно командой **echo**:

```
echo 30 >/proc/sys/vm/swappiness
```

Подробную документацию по /proc можно найти в файле **/usr/src/linux/Documentation/filesystems/proc.txt**.

сделать так, чтобы оно показывалось автоматически после входа в систему через SSH?

Purple Penguin [Пурпурный Пингвин]

Когда Вы входите в оболочку, она загружает профиль — набор команд, которые устанавливают командную строку, задают несколько общих алиасов и делают кое-что еще. Профиль — это просто набор команд, которые выполняются оболочкой перед тем, как Вы начинаете ей пользоваться, и в него можно добавить почти все что угодно, в том числе и команду `landscape-sysinfo`.

Расположение профиля зависит от оболочки, которой Вы пользуетесь. `Bash` загружает системный профиль из `/etc/profile`, а затем профиль пользователя из первого найденного из трех файлов `~/bash_profile`, `~/bash_login` или `~/profile` — это относится к оболочке входа в систему. Оболочка не для входа в систему, которую Вы получаете при запуске `Bash` в командной строке или при переключении пользователей с помощью `su` (без параметра `-l`), вместо них загружает файл `~/bashrc`.

В каждом случае сперва загружается глобальный профиль, а значит, то, что в пользовательском профиле, имеет преимущество. Команды выполняются по порядку, и перед запуском следующей должна завершиться предыдущая, поэтому не добавляйте сюда команд, которые требуют времени, иначе оболочка будет запускаться с задержкой.

В Вашем конкретном случае лучше всего, пожалуй, добавить `landscape-sysinfo` в `/etc/profile`, так как тогда информация будет доступна всем пользователям.

5 С жесткого на твердый

Вя мигрировал с Windows на Linux, и теперь на моем ноутбуке Ubuntu 12.04 LTS, как я и хотел. Я хотел поменять жесткий диск на твердотельный, но не хочу ничего переустанавливать! Можно ли сделать копию всей системы на флэшку или как-то легко перенести ее на новый твердотельный диск?

Джон Бертон [Jon Burton]

Копирование данных с жесткого диска на твердотельный — не то же самое, что с одного жесткого диска на другой, так как твердотельные диски придирчивы к разбивке на разделы. Если выравнивание не идеальное, это скажется на производительности, и со временем она станет хуже, чем у жесткого диска. Самый безопасный вариант — разбить диск вручную, скопировать данные с помощью флэшки и переустановить загрузчик. Все это нужно делать, загрузившись с Live CD — диск с Ubuntu прекрасно подойдет. Загрузитесь с него, подключите USB-диск, откройте терминал и выполните команду:

```
fdisk -l /dev/sda
```

Она покажет, какие разделы есть на жестком диске. Если Вы используете разбивку Ubuntu по умолчанию, состоящую из раздела подкачки и единственного раздела для всего остального, то Вам нужно скопировать только один раздел. Подключите USB-устройство и посмотрите, куда оно смонтировалось. Назовем его `/media/usbstick`. Затем можно смонтировать и создать резервную копию корневого раздела, следующими командами:

```
sudo -i
mkdir -p /mnt/root
mount /dev/sda1 /mnt/root
tar czf /media/usbstick/root.tar.gz /mnt/root
```

Первая строка переключит Вас в `root`, что позволяет не добавлять перед каждой командой `sudo`. Затем монтируется корневая файловая система (предполагается, что она на `/dev/sda1`), и на флэшку делается ее резервная копия. Просто скопировать файлы нельзя, потому что при этом могут не сохраниться атрибуты файлов Linux. Посмотрите, не отформатирована ли флэшка в FAT, так как в FAT максимальный размер файла 4 Гб, а этого может быть недостаточно. Флэшку нужно отформатировать в NTFS, а лучше в ext2. Стоит проверить резервную копию командой:

```
tar tvf /media/usbstick/root.tar.gz
```

Если ошибок нет, выключите ноутбук и замените диск. Снова загрузитесь с компакт-диска и разбейте диск точно так же, как ранее. Запустите

Partition Editor и создайте точно такие же разделы, какие у Вас были до этого. В окне Create Partition [Создать раздел] убедитесь, что Align To: [Выравнивание] установлено в MiB [Мебибайт], затем создайте корневой раздел и раздел подкачки. Чтобы работал спящий режим, раздел подкачки должен быть как минимум равным объему доступной оперативной памяти. В качестве типа корневого раздела выберите ext4. Примените изменения и перезагрузитесь, чтобы переключиться на новую таблицу разделов, затем восстановите файлы командой, противоположной той, которую запускали ранее:

```
sudo -i
mkdir -p /mnt/root
mount /dev/sda1 /mnt/root
tar xf /media/usbstick/root.tar.gz -C /mnt/root
```

Разделы в Ubuntu обозначаются через UUID. Для просмотра UUID новой файловой системы выполните команду

```
sudo blkid /dev/sda1
```

Затем откройте файл `/mnt/root/etc/fstab` и замените старый UUID новым. Сделайте то же самое для раздела подкачки. Наконец, установите загрузчик на твердотельный диск командой:

```
grub-install --boot-directory=/mnt/root/boot --root-directory=/mnt/root /dev/sda
```

Вот и все; перезагрузите систему и наслаждайтесь скоростью твердотельного диска.

6 Сполоснуть и повторить

Большинство программ я устанавливаю из менеджера пакетов, но иногда программы, которая мне нужна, нет в репозиториях, или для нее нет RPM (у меня Fedora), и приходится устанавливать ее из исходников. Я распаковываю архив и запускаю команду `.configure`. Она проверяет всякие зависимости и завершается, когда чего-то не хватает. Обычно мне везет, и после установки одной библиотеки проблема решается. Однако недавно я пробовал установить программу и при каждом перезапуске `configure` продвигался чуть дальше, но потом все равно останавливался. Интересно, можно ли велеть скрипту `configure`, чтобы он не завершался при нахождении одной отсутствующей зависимости? Таким образом у меня появился бы список зависимостей, которые я мог бы установить сразу с помощью менеджера пакетов. Уверен: тогда и я, и многие другие сэкономили бы массу времени.

Эндрю Корт [Andrew Court]

Проверить, есть ли в Вашей системе требуемые зависимости, можно; этим как раз и занимается скрипт `configure`. Для этого он и нужен; но он делает свою работу по-простому, проверяя зависимости по одной и завершаясь при первой же ошибке. Этак, может, и медленно, зато надежно. Как только обнаруживается отсутствующая зависимость, не делается никаких предположений о том, можно ли дальше работать в этой среде. Даже если бы скрипт мог продолжить работу, результаты были бы ненадежными.

Но не все потеряно. Во многих пакетах список зависимостей приводится в файле `Readme` или `Install`, и это быстрее, чем снова и снова запускать



Landscaper или любую другую команду, которую нужно запускать при входе в систему, можно добавить в свой профиль.

Помогите нам
помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, поскольку проблема описана в них недостаточно полно. Чтобы дать вам наилучший ответ, нам нужно знать как можно больше.

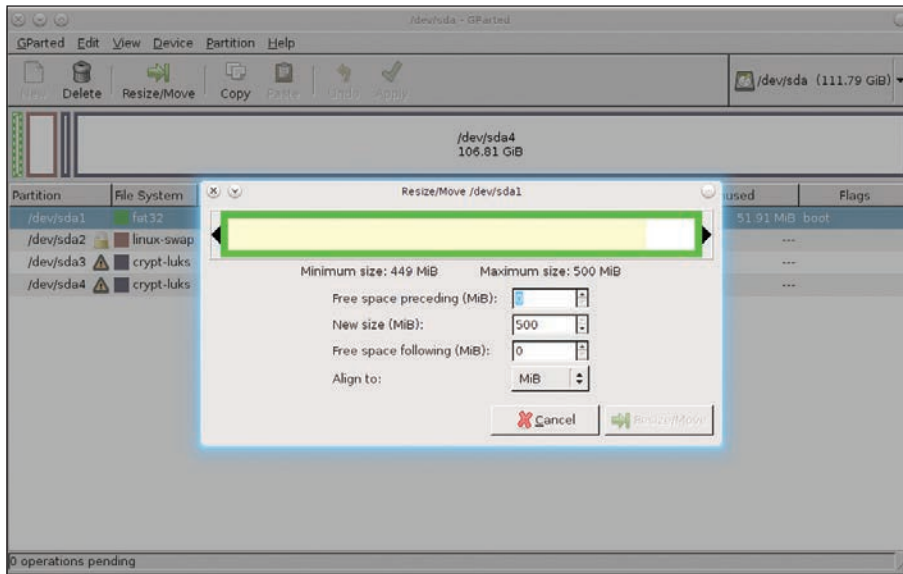
Если у вас появляется сообщение об ошибке, приведите его точный текст и опишите конкретные условия, когда оно появляется. При возникновении проблемы с устройствами перечислите нам все установленные устройства.

Если Linux уже запущен, можете применить для этого отличную программу *Hardinfo* (<http://hardinfo.berlios.de/>) — она сохранит подробную информацию об устройствах и о состоянии системы в HTML-файле, который вы сможете приложить к своему письму.

Не уступающий в удобстве альтернативный вариант — *lshw* (<http://ezix.org/project/wiki/HardwareLiSter>). Одна из указанных программ непременно должна быть включена в ваш дистрибутив (а иногда и обе).

Если вы не хотите или не можете их установить, выполните следующие команды в терминале от имени root и приложите файл **system.txt** к письму. Это здорово поможет диагностике.

```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```



➤ При разбиении твердотельного диска на разделы необходимо использовать определенное выравнивание. *Gparted* делает это за вас.

./configure. Если этой информации нет в архиве, попробуйте зайти на сайт проекта. Если ее нет и там, отправьте сообщение об ошибке. Если скрипт падает из-за определенной зависимости, можно набрать **./configure -help** и получить список аргументов для дополнительных возможностей. Затем можно запустить скрипт снова, отключив эту зависимость, чтобы посмотреть, появится ли следующая; но это все же ненадежно.

Скрипт **./configure** работает медленно. Он выполняет проверки одну за другой, и если он па-

дает, пройдя 90 % пути, придется, установив зависимость, запускать все с нуля. Другой вариант — воспользоваться аргументом **--configcache:**

```
./configure --config-cache
```

В этом случае результаты работы скрипта кэшируются в файле **config-cache**, который при следующем запуске считывается. Также можно немного ускорить работу, особенно на медленном терминале или при SSH-подключении, указав параметр **--silent**, который отключает вывод всех сообщений **Checking for foo... [Проверка ...]. [txt]**



Часто задаваемые вопросы

Мобильный широкополосный доступ

➤ **Что такое этот мобильный широкополосный доступ, о котором все болтают? Нечто вроде ADSL для мобильных телефонов?**

Нечто вроде. Он использует мобильную сеть 3G, но не ADSL. Технология называется HSPA (High Speed Packet Access — высокоскоростной пакетный доступ) и предназначена для мобильных устройств. Она используется в смартфонах, которым нужен постоянный (или по крайней мере частый) доступ в Интернет.

➤ **Будет ли у меня такая же скорость, как при широкополосном подключении?**

Нет. Это не широкополосное подключение в исходном смысле слова (как, кстати, и некоторые из более медленных проводных подключений), но его скорость достаточна для

решения важных задач, таких как чтение почты и просмотр видео на YouTube. Анонсируемая скорость у разных компаний разная, но она всегда серьезно зависит от уровня сигнала и сопутствующих факторов.

➤ **Что нужно, чтобы ею воспользоваться?**

Вам нужен широкополосный модем и договор с мобильным оператором. В большинстве договоров модем уже входит, но можно воспользоваться и собственным. Все операторы используют по сути одинаковые модемы с USB-подключением. В такой модем вставляется SIM-карта, которая идентифицирует вас.

➤ **Сколько это стоит?**

Как и для большинства телекоммуникационных услуг, стоимость

зависит от поставщика услуги, продолжительности договора и объема пересылаемых данных. В Великобритании за £15 можно купить 3 ГБ трафика в месяц.

➤ **Каков срок действия договора?**

Он варьируется от одного до 18 месяцев, либо можно платить за услуги сразу. При договоре на больший срок предоставляется бесплатный модем.

➤ **Хорошо ли широкополосный доступ работает в Linux?**

Сейчас очень хорошо. Все компании, похоже, используют модемы одного производителя — Huawei, а драйверы для него есть в ядре Linux, поэтому он должен работать сразу. Вам нужно настроить модемное соединение в *KPPP* или *Gnome-*

PPP, выбрав в качестве модема **/dev/ttyUSB0**. Так как аутентификация выполняется с помощью SIM-карты, имя пользователя и пароль могут быть любыми (если, конечно, программа требует вводить что-то в эти поля).

➤ **А дистрибутив имеет значение?**

Стандартные настройки PPP должны работать во всех дистрибутивах. Дистрибутивы, в которых есть *Network Manager*, имеют преимущество, потому что теперь он поддерживает 3G-модемы. В таком дистрибутиве при подключении модема может появиться окно, где нужно выбрать провайдера. Затем все будет настроено автоматически. После этого вы сможете включать и выключать модем в меню *Network Manager*.



LXF HotPicks



Майк Сондерс
Изажив самые недосягаемые и укромные уголки Интернета, Майк точно знает, где таятся главные сокровища открытого кода.

RText » TMSU » Cosmosmash » NcdU » Typer » Ascii Design » ZedLog
» Tower Toppler » Converseen » Crayonizer » QMentat

Редактор для программистов

RText

Версия 2.0.7 Сайт www.fifesoft.com/rtext

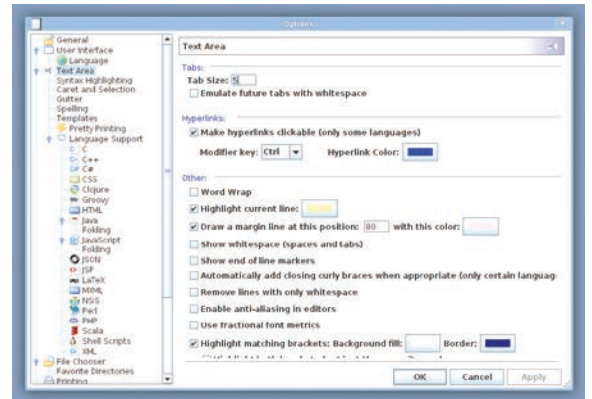
Многие программисты всю свою жизнь ищут идеальный текстовый редактор. Одни оседают в *Vim*, другие — в *Emacs*, а третьи выкапывают какой-нибудь безвестный инструмент из репозитория своего дистрибутива. Так или иначе, ценнейшей функцией редактора является его способность работать на разных платформах: в конце концов, сейчас вы кропаете код в Linux, но в будущем, возможно, вам достанется работа на машине с Windows, или Mac,

а то и вообще с каким-нибудь экзотическим Solaris или Atheos.

Убойная черта *RText* в том, что он написан на Java, и по своему виду, вкусу и запаху одинаков на всех платформах, на какой бы он ни работал. А тестировали его на многих платформах — даже на NetWare. Если ваш опыт обращения с Java ограничен web-апплетами, вы, вероятно, ожидаете, что *RText* будет невероятно раздутым и медлительным монстром, который разлетается на куски каждые 20 секунд. Все наоборот!

RText на удивление шустрый и поразительно надежен.

Его даже не нужно устанавливать; просто



» Почти любой аспект редактора (а не только цветовую схему) можно настроить в точности как душа просит.

распакуйте файл `.tar.gz` и запустите в полученной директории `java -jar RText.jar`. По умолчанию *RText* аккуратно разделена на несколько панелей и информационных панелей, согласно описанию.

По части функций в нем есть практически все, что полагается редактору для программиста: панель редактирования с выделением синтаксиса (на 26 языков), свертыванием кода, сопоставлением скобок; система макросов для автоматизации рутинных задач; закладки для быстрого перехода к часто запрашиваемым фрагментам исходного кода; браузер кода для определения местонахождения функций и переменных; и многое другое. Вы можете по ходу создавать привязанные к клавишам «инструменты» (например, привязать `Ctrl+B` к команде `make` в директории кода), добавлять к редактору IDE-подобные элементы, а настраиваемых параметров только в диалоговом окне Options хватило бы на отдельное руководство.

И хотя по функциональности *RText* вряд ли задавит конкурентов, но его лучшей чертой, как уже говорилось, заключается в том, что он одинаково работает на разных платформах. Если вам придется много заниматься программированием на Linux и Windows, вы можете создать идеальную раскладку в Linux, а затем скопировать ее на компьютер с Windows (в Linux настройки хранятся в `~/.rtext/`), и вам никогда не придется дважды задумываться о мелких расхождениях в пользовательском интерфейсе или привязках к клавишам.

«RText написан на Java и одинаков на всех платформах.»

Исследуем интерфейс RText

Панель вкладок

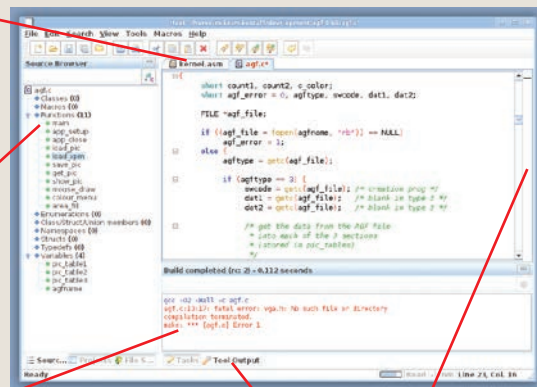
Редактируйте разные файлы чохом — щелкните средней кнопкой по вкладке, как по значку, чтобы ее закрыть.

Браузер

Здесь можно быстро переключаться между определениями переменных и функций в панели редактирования.

Tools output

Все определенные пользователем команды, которые созданы в меню Tools, будут отображать результаты здесь.



Отображение во вкладках

Щелкните по View > Docked Windows > Console, и вы сможете добавить к этим вкладкам терминал.

Ссылки

Щелкните по переменной или имени функции, и строка в правой части полосы прокрутки образует вам другие ссылки на нее.

Организатор файлов

TMSU

Версия 0.2.0 Сайт www.tmsu.org

Нужно иерархия директорий и вправду лучший способ хранения файлов? Вот вы, допустим, размещаете свои деловые файлы в `~/business/`, фотографии — в `~/photos/`, а персональные — в `~/personal/`. Ну, а куда девать такую умильную фотографию своего котика? Это ведь персональный файл, однако он также заслуживает места в папке `~/photos/`, верно?

Куда более элегантное решение — присвоение тэгов: это система, при которой вы навешиваете на файлы один или более тэгов (пометок). *TMSU* — как раз такая система, и разработчик весьма любезно создал прекомпилированный бинарник, требующий для работы только *Fuse* и *SQLite*; если эти зависимости установлены, можно просто скопировать бинарник в `/usr/local/bin` — и вперед.

Лучше всего объяснить, как работает *TMSU*, на примере. Вот есть у нас три файла: `cat.jpg`, `contract.doc` и `letter.doc`. Пусть нам требуется присвоить тэг

“personal” файлам `cat.jpg` и `letter.doc`, а тэг “business” — файлу `contract.doc`; тогда скомандуем

```
tmsu tag cat.jpg personal
```

```
tmsu tag letter.doc personal
```

```
tmsu tag contract.doc business
```

Теперь, введя `tmsu tags`, мы сможем увидеть прикрепленные тэги. Но! Как мы уже говорили, нам ведь нужно прикрепить к картинке с котом тэг “photo”, поэтому введем еще `tmsu tag cat.jpg photo`. Обратите внимание, что файлы не меняются, а все метаданные тэгов хранятся в базе данных в `~/tmsu/`. Это еще не великое счастье; но нашу новую систему на базе тэгов можно распространить на всю установку Linux, командой

```
mkdir mp && tmsu mount mp
```

```

terminal: mikelmike-K52F: ~/tmp
mike@mike-K52F:~/tmp$ ls
cat.jpg  contract.doc  letter.doc
mike@mike-K52F:~/tmp$ tmsu tag cat.jpg personal
tmsu: New tag 'personal'.
mike@mike-K52F:~/tmp$ tmsu tag letter.doc personal
mike@mike-K52F:~/tmp$ tmsu tag contract.doc business
tmsu: New tag 'business'.
mike@mike-K52F:~/tmp$ tmsu tag cat.jpg photo
tmsu: New tag 'photo'.
mike@mike-K52F:~/tmp$ tmsu tags +
cat.jpg: personal photo
contract.doc: business
letter.doc: personal
mike@mike-K52F:~/tmp$ mkdir mp && tmsu mount mp
mike@mike-K52F:~/tmp$ ls mp/tags
business  personal  photo
mike@mike-K52F:~/tmp$ ls mp/tags/personal
cat.1.jpg  letter.2.doc  photo
mike@mike-K52F:~/tmp$ ls mp/tags/photo
cat.1.jpg
mike@mike-K52F:~/tmp$ ls mp/tags/photo/personal/
cat.1.jpg
mike@mike-K52F:~/tmp$
  
```

➤ Результат команд из нашего примера — обратите внимание, что файлы могут появляться более чем в одной директории, на основании их тэгов.

Здесь *TMSU* создает виртуальную файловую систему (с помощью *Fuse*) в директории `mp`. Войдите в нее, затем в поддиректорию `tags`, и вы поймете, как это работает: налицо директории для всех созданных тэгов (`business`, `personal` и `photo`). Файл `cat.jpg`, как и следовало ожидать, виден и в папке `personal`, и в папке `photo` — и это один и тот же файл, просто он доступен в обоих местах через символические ссылки. Теперь вы можете получить доступ к файлам в этих папках из любой программы Linux; их поиск отныне основан на тэгах, а не на иерархической системе. Снабдив все файлы множественно тэгов, вы убедитесь: *TMSU* — намного более эффективный способ организации файлов.

«TMSU — намного более эффективный способ организации файлов.»

Пакетный конвертер изображений

Converseen

Версия 0.6.2 Сайт <http://converseen.sourceforge.net>

ImageMagick — одна из этих удивительно мощных, давно существующих программ Unix, о которых все слышали; но лишь немногие умеют пользоваться всеми ее возможностями. Этот инструмент обработки изображений появился в 1990 году, и теперь поддерживает более 200 форматов файлов. *ImageMagick* производит множество трансформаций и применяет всевозможные эффекты к изображениям — размытие, резкость, обрезку, изменение размера, анимацию и многое другое, и умопомрачительно богата функциями.

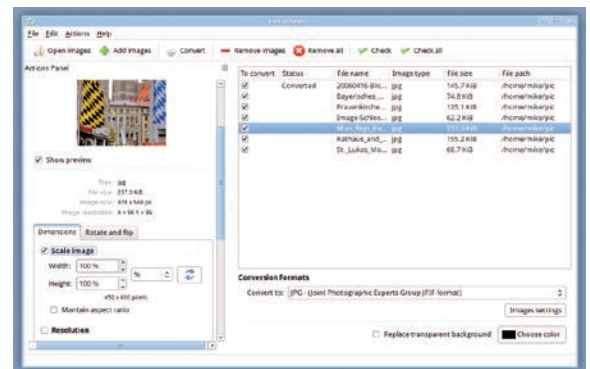
Но *ImageMagick* в основном управляется опциями командной строки, а их не так-то легко запомнить. Если надо срочно выполнить некое действие, например, изменить размер или конвертировать пакет изображений, придется вам лезть в справочник и разбираться с программой. *Converseen* решает эту проблему, размещая поверх *ImageMagick* GUI на базе Qt;

чтобы скомпилировать его из исходника, понадобятся библиотеки *Magick++* и заголовки разработки.

Когда *Converseen* запустится, щелкните по кнопке Open Images [Открыть изображения] вверху слева и выберите партию файлов с изображениями (несколько файлов выбираются через Shift+щелчок). Файлы появятся в панели списка справа, и вы можете отметить те, которые хотите конвертировать. Под этой панелью вы увидите выпадающий список, где выбирается формат файлов для конверсии; он впечатляюще длинный, что демонстрирует вышеупомянутую мощь *ImageMagick*.

Мощь Qt

Все подготовив, щелкните по кнопке Convert вверху, и ваши изображения переписутся заново в выбранном формате. Но это только первая функция: прокрутив предпросмотр изображения и спустившись ниже в левой стороне, вы увидите,



➤ Многие форматы изображений имеют дополнительные опции, доступные по кнопке Image Settings.

что с изображениями можно проделать намного больше. Например, уменьшить их масштаб, поддерживая при необходимости коэффициент пропорциональности, или повернуть их, или зеркально отразить. Полезная функция — добавление суффикса или префикса к имени файла конвертированных изображений, чтобы ни в коем случае не затоптать оригинал.

Converseen являет собой классический пример модульности, которую мы так ценим в Unix/Linux: вместо того, чтобы за все отдаваться самому, он делает надстройку над мощными функциями других программ, добавляя поверх нее GUI.

Анализатор использования диска

Ncdu

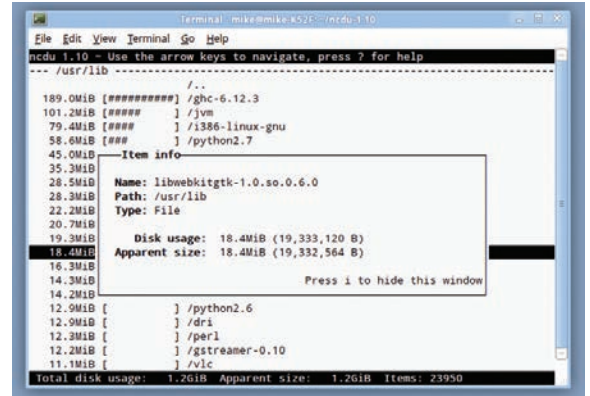
Версия 1.10 Сайт <http://dev.yorhel.nl/ncdu>

С вашим сервером творятся всякие странности... он не спешит отвечать, сервисы случайным образом схлопываются, а данные теряются. Ну, коли мы говорим о славном сервере Linux, вряд ли он страдает от вирусов. Нет. Куда более разумное объяснение — то, что у него просто иссякло место на диске. А для систем Linux (в частности, для серверов с массой работающих процессов) это всегда имеет непредсказуемые последствия, поскольку каждая программа обрабатывает сценарий «некуда сохранять» по-своему. Нередко — кончиной.

Поэтому все приличные сисадмины стараются отслеживать использование диска; но стандартный инструмент *du*, хотя и обладает достаточными возможностями, не особо впечатляет. *Ncdu* предоставляет намного более привлекательный (и очень информативный) эквивалент, созданный на базе библиотеки текстовых виджетов *Ncurses*. Это все, что вам нужно для его компиляции из исходника,

но у разработчика есть версия, статически ссылающаяся на крошечную библиотеку C на сайте проекта, если вдруг она вам понадобится.

Наберите *ncdu* из командной строки, и вам покажут список директорий и файлов, по которому вы сможете перемещаться клавишами курсора. Для каждого пункта в списке вы увидите размер и псевдографическое изображение того, какое место на диске он занимает, через знаки диеза. Клавишей G можно изменить формат отображения, чтобы заодно показывалось процентное соотношение. Само по себе это уже исключительно ценная функция, поскольку вы сразу можете увидеть, какие пункты занимают больше всего места, и вам не придется привешивать



Вас, возможно, удивит, какие пожиратели пространства прячутся на вашем диске.

всякие опции сортировки и форматирования в конце команды *du*.

Для перехода в директорию, выделите ее и нажмите Enter; в директорию выше этажом вы сможете вернуться по клавише <. Нажав на клавишу I, вы увидите небольшую информационную панель по данному пункту, а клавиши N и S сортируют список соответственно по имени и размеру. Есть еще клавиши, с которыми можно побаловаться — нажмите на ? для вывода окна справки и просмотра прочих опций.

Мизерный вес *Ncdu* (голый бинарик — 59K), минимум зависимостей и текстовый режим интерфейса облегчают его установку на любой сервер и работу с SSH. Без него мы больше на серверы ни ногой...

«Все приличные сисадмины отслеживают использование диска.»

Программа обучения набору текста

Typer

Версия 0.4.0 Сайт <http://code.google.com/p/juanmabcprj/wiki/Typer>

То, что вы — матерый линуксоид, еще не значит, что вы здорово набираете тексты. Например, данный автор довольно криворук в этом смысле и не чувствует себя комфортно ни с одной клавиатурой. Видимо, пора браться за программу обучения набору текста...

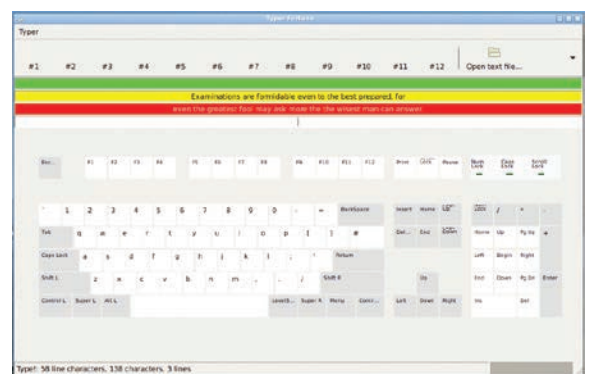
Изобретательно названный *Typer* уже несколько лет томится где-то посреди разработки, но в последнем релизе разработчик обновил его под самые современные библиотеки Gnome. На сайте проекта есть RPM для Fedora 18 (и совместимых дистрибутивов), и если у вас установлены заголовки разработки Gnome, компиляция из исходника не должна вызвать проблем.

Typer полностью отображает клавиатуру ПК, что сперва кажется не особо удобным, но зато позволяет видеть, какие спецсимволы можно ввести, удерживая правую клавишу Alt [это касается клавиатур с неамериканской раскладкой, — прим. ред.]. Сверху располагаются кнопки от 1 до 12,

которые создают крайне сложные задачи по набору текста. Нажмите на кнопку, и вам предложат ввести текст в желтой строке; зеленая строка показывает вам уже набранное, а красная — то, что еще предстоит. Внизу строка состояния показывает, сколько символов вы набрали правильно, и там есть таймер, так что вы сможете состязаться сами с собой.

Тест текста

Пронумерованные тесты несколько странные: они производят случайные комбинации букв, наводящие на мысли о нагромождениях согласных в гренландском языке. Куда приятнее тест, предлагаемый *Fortune*, небольшой программой, установленной на многих системах Linux и Unix, которая выдает цитаты, мысли и прочие малоформатные проявления мудрости. Прогулка по базе данных *Fortune* всегда вызывает здоровый хохот, а ее применение в практических целях — отличная идея. Или же



Выбор цветовой схемы светового фона для строк текста отчасти безвкусен, но это — единственное, к чему мы можем придраться.

загрузите в качестве практического материала любой текстовый файл.

Это не самая многофункциональная программа по обучению набору текста, но она делает весьма достойную работу. Правда, озадачивает система подсказки: нажав на Ctrl+N, вы окажетесь на странице Wikipedia, посвященной набору текста. Умно, нечего сказать...

Программа раскраски командной строки

Crayonizer

Версия 0.2 Сайт <http://sites.google.com/site/columnscode/home/crayonizer>

Будем честны: командная строка довольно уныла и некрасива. Мы любим ее за мощь и способность связывать операции, за секунды выполняющая задачи, на которые в файловом менеджере ушли бы часы. Но зачем нам мириться с ее монохромной безликостью? Некоторые эмуляторы терминала пытаются оживить ее с помощью полупрозрачного фона, сквозь который видны обои рабочего стола, но это не спасает ситуацию.

Crayonizer — умный инструмент, который добавляет цвета в выводимый на экран результат работы многих команд. И это не только делает терминал красочнее, но и помогает сделать результат более читаемым — примерно так же, как выделение синтаксиса в текстовых редакторах. Его метод работы довольно сложен: нужно поместить его в директорию, идущую первой в вашем `$PATH` (например, `/usr/prebin/`), а затем сделать символическую ссылку на основе команд, которые вы хотите использовать.

Например, *Crayonizer* может раскрасить результат работы `ifconfig`. Для этого нужно создать символическую ссылку на бинарник `crayonizer`, под названием `ifconfig`. Теперь при вводе `ifconfig` в окне оболочки вместо запуска обычной версии сначала запустится версия по символической ссылке в `/usr/prebin/`, и она будет расцветивать результаты программы-оригинала (`/sbin/ifconfig`). Если вам это показалось слишком сложным, взгляните в файл `README`, там все объясняется.

Crayonizer способен украсить немало команд, в том числе `make`, `top`, `ps`, `netstat` и `dmesg`. Немного потрудившись, можно добавить поддержку и других программ, отредактировав файл `crayonizer.conf`; следовать синтаксису довольно легко —

```

Terminal: mike@mike-K52F: ~/Crayonizer-0.2
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
Interrupt:47

Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr:::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:112 errors:0 dropped:0 overruns:0 frame:0
TX packets:112 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:8640 (8.6 KB) TX bytes:8640 (8.6 KB)

Link encap:Ethernet Hwaddr:e0:a5:e0:a5:fc
inet addr:192.168.2.2 Bcast:192.168.2.255 Mask:255.255.0
inet6 addr::e2b9::e2b9:a5ff:fe1d:e2b9/64 Scope:Link
UP BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:5015 errors:0 dropped:0 overruns:0 frame:0
TX packets:3387 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4188808 (4.1 MB) TX bytes:623857 (623.8 KB)

```

► Вот улучшенная версия результатов `ifconfig`.

просто скопируйте то, что введено для одной команды, и возьмите за основу. *Crayonizer* не имеет зависимостей сверх обычных для любой системы Linux, и это всего лишь одиночный бинарник — его можно накопировать на несколько машин.

Пользовавшись цветной версией команд, будет тяжело вернуться к их монохромным эквивалентам.

«Crayonizer добавляет цвета в результат многих команд.»

Клавиатурный шпион (Кейлоггер)

ZedLog

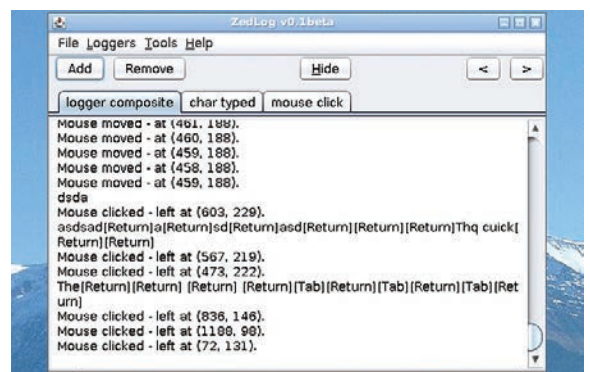
Версия 0.1b Сайт <http://bit.ly/11nbNFm>

Наверное, мы взяли за дело не с того конца, когда писали материалы `LXF170` «Взлом Сети» и `LXF154` «Изучите взлом». Да, мы показали, как взламывать пароли и использовать уязвимости — но не со злым умыслом, а как раз наоборот. Мы объясняли, как хакеры делают свою работу, какими инструментами пользуются, и как от них защититься. Для должной реализации политики безопасности надо точно знать, что именно вам угрожает.

Такая же история с клавиатурными шпионами [keylogger]. Их применяют и с гнусными намерениями, но есть ситуации, когда они помогут пресечь противозаконные действия. Скажем, вы уверены, что некто использует ваш компьютер для сомнительного свойства онлайн-приключений, а историю посещений перед вашим приходом стирает. С помощью кейлоггера вы, по крайней мере, сможете получить и предъявить доказательства злодеяния.

В базовой форме кейлоггер представляет собой небольшую программку, которая сидит в фоновом режиме, записывает нажатия на клавиши и сохраняет их в файле для дальнейшего просмотра. *ZedLog* — программа на Java, которая делает именно это, да еще имеет парочку тузов в рукаве; чтобы запустить ее, распакуйте файл `.zip` и наберите `java -jar zedlog.jar`. В появившемся окне щелкните Add слева вверху, чтобы запустить новый процесс записи. Можно выбрать, записывать ли отдельные нажатия на клавиши, движения мыши или щелчки.

Когда запись начнется, вы увидите ее результаты во вкладках — если у вас запущено несколько процессов записи,



► *ZedLog* пока что на очень ранней стадии разработки, но основные функции в нем уже присутствуют.

то их объединенный результат будет отображаться во вкладке Composite.

В меню File выбирается файл, в который будет вестись запись всех действий. Клавиатурный шпион бессмыслен, если ваш подозреваемый знает о его присутствии, поэтому нажмите на кнопку Hide, чтобы стереть все следы *ZedLog* с экрана. Работать он не перестанет, однако вернуть его на божий свет не так-то просто — это делается только через встроенную подсказку (F1), которая на данный момент, похоже, не работает. Но мы влезли в `.jar` и откопали соответствующие документы: чтобы окно *ZedLog* появилось снова, щелкните средней кнопкой точно в левом верхнем углу экрана.

«Для должной политики надо точно знать, что именно вам угрожает.»

HotGames Развлекательные приложения

Игра-платформер

Tower Toppler

Версия 1.6 Сайт <http://toppler.sourceforge.net>

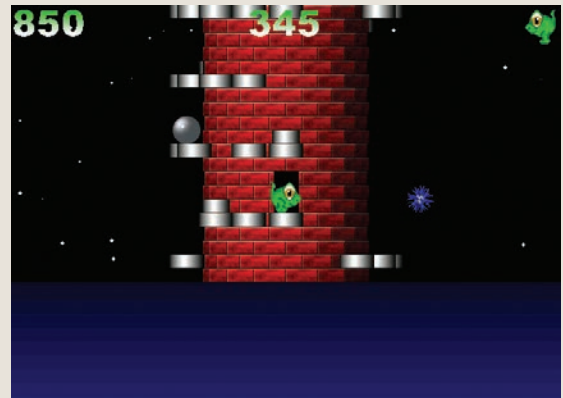
Мы не намекаем, что дизайнеры Tower Toppler переели грибочков, но право, странно, как же они дошли до такой концепции. «Итак, зеленая лягушка выпрыгивает из подводной лодки на цоколь башни, и ей нужно залезть на башню, поджигая мячики от пинг-понга, перескакивая через шипы и проходя сквозь двери. Когда она доберется до вершины, башня рухнет, и лягушка отправится рыбачить на своей подводной лодке, пока не достигнет другой башни...»

Кхе, ну-ну. Но если отставить в сторону безумный сюжет, *Tower Toppler* — это действительно хороший платформер, впервые появившийся в 1987 году под названием *Nebulous*. Тогда он был портирован на ряд платформ, и на сегодняшний день у нас есть версии с открытым кодом, C++/SDL, благодаря Андреасу

Рёверу [Andreas Roever] и Паси Каллинену [Pasi Kallinen]. Компиляция из исходника не представляет ни малейшей сложности (см. файл **README**), а запускается игра посредством `./toppler`.

Как уже упоминалось, ваша цель — достичь верхней двери в каждой башне. Говоря об управлении — вы передвигаете лягушку стрелками, нажимая на пробел для прыжка (в движении) или броска мячом (когда стоите). Что, в общем, обычно для платформера; однако уникальным *Tower Toppler* делает фокусирование всегда на лягушке. Ваш персонаж нипочем не сместится с центра экрана, а башня

«Ваш персонаж нипочем не сместится с центра экрана.»



» Даже не будь все так сложно, ограничение времени уже само по себе напрягает...

вращается вокруг вас, и отсутствует риск наткнуться на правую или левую грань.

Это любопытным образом влияет на игровой процесс: можно, например, врезаться в препятствие с одной стороны, и мимо пройти нельзя. Однако подойдите к нему с обратной стороны, и эта миссия окажется вполне выполнимой. Сюда входит 60 уровней; некоторые из них невероятно сложны, и есть даже редактор уровней, если, конечно, вы уже перебрали все представленные.

Стрелялка

Cosmosmash

Версия 1.4.7 Сайт <http://perso.b2b2c.ca/sarrazip/dev/cosmosmash.html>

На обилие насилия в нынешних видеоиграх часто жалуются, однако мы и не упоминим времена, когда это было по-другому. Практически каждая космическая игра включает необходимость палить в кого-то из лазера — или этот кто-то пальнет из лазера в вас. Именно так было с *Astromash*, релизом 1981 года для Intellivision (позднее портированным на Atari 2600). *Cosmosmash* — клон *Astromash* с открытым кодом, написанный на C++ и использующий SDL.

Для компиляции *Cosmosmash* необходимо сначала скомпилировать и установить игровую библиотеку *Flatzebra*. Затем отработайте обычные процедуры `./configure`, `make` и `make install` (от имени root), после чего просто введите `cosmosmash`, она и запустится. Вы сразу же увидите, что графика — тотальный стиль ретро; хотя спрайты обсчитываются в намного более высоком

разрешении, чем в версиях старой школы, дизайн крайне строг, а цветовая гамма отличается скупостью.

Игровой процесс характеризуют быстрая и ярость — прямо-таки пальцы на клавиатуре заплетаются. Вы управляете космическим кораблем, который должны провести по низу экрана, управляя стрелками, и корабль автоматически выдает лазерные залпы. Ваша задача — палить во все, что встречается на пути; если низ экрана заденет метеор — вы потеряете баллы, а если этого низа достигнет белый «волчок», корабль взорвется, и ваша жизнь завершена.

Жизнь в 1980-х

Пока все стандартно для стрелялки 1980-х, но здесь есть один очень крутой трюк, добавляющий изюминку игровому процессу: гиперпространство. Нажатие на стрелку вниз телепортирует ваш



» С точки зрения графики — ничего выдающегося, но игровой процесс отличается неплохим балансом между умениями и везением.

корабль в случайно выбранное место внизу экрана. Это может спасти вам шкуру, если вы попадете в ловушку, а при везении вы займете идеальную позицию для обезвреживания волчка. Но можно, наоборот, угодить в куда более затруднительное положение, так что все это — не более чем рулетка, хотя и забавная.

Программа создания произведений ASCII

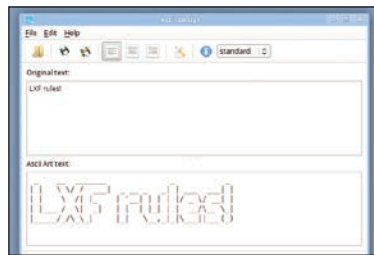
Ascii Design

Версия 1.0.1 Сайт <http://ascii-design.sourceforge.net>

Давным-давно, еще до изобретения растровой графики и прочих хитроумных технологий, миром правило искусство ASCII. Это был способ создания изображений из текстовых символов — то есть букв, цифр, знаков препинания и т.п. Особо тонких деталей этап не прорисуешь, но все же оно стало отдельной формой искусства, и даже сегодня вы периодически можете видеть логотип в стиле ASCII в текстовых файлах или подписях электронной почты.

И вот как создается подобный шедевр. Сначала устанавливаем *Figlet* (он включен в репозитории многих дистрибутивов, или скачайте его с www.figlet.org). Затем скомпилируем *Ascii Design*, распаковав его файл `.tar.bz2` и следуя инструкциям в `INSTALL.TXT` в получившейся директории. При первом запуске *Ascii Design* попросит указать местоположение бинарника *Figlet* и шрифтов — наиболее вероятны `/usr/bin/figlet` и `/usr/share/figlet/`, но все зависит от вашей настройки.

Затем в *Ascii Design* вводите текст в верхней панели, наблюдая его эквивалент в виде продукта ASCII внизу. В выпадающем списке вверху справа (отмеченном по умолчанию как «баннер») выбираются разные шрифты; некоторые из них выдают впечатляющие результаты, с учетом, конечно, имеющихся ограничений. Продукт можно скопировать и вставить за свою подпись в электронной почте или на форуме, и это куда дружелюбнее к пропускной способности, чем всякие смехотворные картинки, которыми пользуются многие. Имею я право поворчать?..



► Вот огромный баннер, который можно невзначай подбрасывать в беседы по IRC.

Программа обучения счету в уме

QMentat

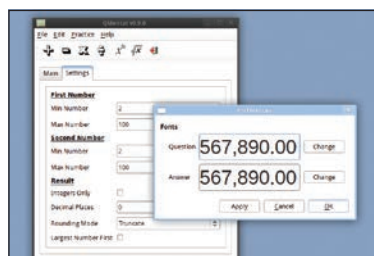
Версия 0.9.0 Сайт <http://realgrep.github.io/QMentat>

Ощущаете некое отставание в математической части большой тройки (чтение, письмо, математика)? Возможно, *QMentat* поможет горю — это «гибкая и мощная программа для тренировки счета в уме»; она ставит перед вами математические задачи и просит их решить. Но прежде чем упражнять свои математические таланты, придется поломать голову над ее компиляцией.

Инструкции по компиляции *QMentat* содержатся в файле `INSTALL`, однако на стадии `qmake` могут появиться сообщения об ошибках, утверждающие, что не найдены два файла `.qm`. Это не особо значимые файлы перевода, отсутствующие в `tar`-архиве, поэтому отредактируйте `qmentat.qrc`, удалив две строки "file" в разделе "translations". Тогда компиляция пройдет гладко.

В процессе работы *QMentat* умеет переключаться между шестью типами вопросов: сложение, вычитание, умножение, деление, возведение в степень и извлечение

корня. Цифры по умолчанию довольно малы и не поставят в тупик нормального взрослого человека, но для каждого типа вопросов есть вкладка Settings, позволяющая усложнить задачи. Для некоторых вопросов, например, про деление, есть дополнительные опции: скажем, генерировать задачи, имеющие только целочисленные решения, или настроить шрифт, или отобразить диаграмму своих результатов. Вот, в общем-то, и все, но это отлично сработанный инструмент, и прекрасный способ стимулировать серое вещество. **LXF**



► Мы думаем, что отличным дополнением к 1.0 был бы таймер, чтобы малость поддать жару.

Также вышли

Новые и обновленные программы, тоже достойные внимания...

► Zim 0.60

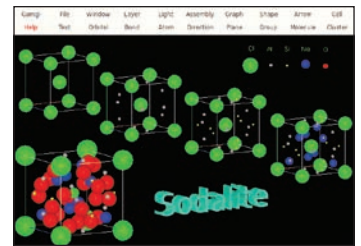
Графический текстовый редактор для ведения настольной wiki. <http://zim-wiki.org>

► XtermControl 3.0

По ходу дела настраивайте в *XTerm* цвета, шрифты, геометрические фигуры и прочее. www.thrysoe.dk/xtermcontrol

► Gamgi 0.16.4

Создавайте, просматривайте и анализируйте атомарные структуры. www.gamgi.org



► Стряпайте молекулы, похожие на гроздь шариков, и прочие забавные штуки.

► XOWA 0.5.0

Настольное приложение для оффлайн-редактирования Википедии. <http://sourceforge.net/projects/xowa>

► mdp 0.2

Генератор и менеджер паролей, применяющий для шифрования *GnuPG*. <http://tamentis.com/projects/mdp>

► vcprompt 1.1

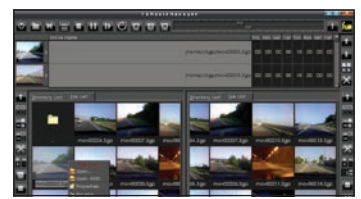
Добавьте в свою оболочку систему управления версиями. <http://freecode.com/projects/vcprompt>

► Sanzang 1.0

Кросс-платформенная система перевода для человеческих языков. <http://rubygems.org/gems/sanzang>

► FXMovieManager 6.3

Файловый менеджер, отображающий миниатюры файлов фильмов. <http://sourceforge.net/projects/fxmoviemanager>



► Приложение позволяет создавать плей-листы и смотреть видео в *MPlayer*.

На диске

Дистрибутивы, приложения, игры, подкасты и всякое-разное...

Лучшее из Интернета, помещенное в 8 ГБ качественного DVD.



Дистрибутивы

Чуть больше года я готовил диски Linux Format DVD, но это мой последний. Нет, я не покидаю *Linux Format*, просто возвращаю управление DVD оплоту LXF Нейлу Ботвику, а сам смогу каждый месяц дарить журналу тонны слов.

Больше всего после разлуки с DVD мне будет не хватать возможности попробовать все дистрибутивы. Я экспериментировал примерно с восемью каждый месяц, чтобы оценить их и выбрать те, который войдут на DVD.

Я пробовал хорошие дистрибутивы, плохие дистрибутивы и дистрибутивы очень странные. Вместе они создают мильную оперу, которой и является Linux. Стычки, разводы, причудливые семейные древа... Все это порождает культуру, которой не может похвастаться ни одна другая ОС.

Возможно, было бы перебором утверждать, что человека характеризует дистрибутив, которым он пользуется, но я думаю, по разнообразным дистрибутивам вы можете сказать кое-что о нашем сообществе. Возможно, самое главное — *насколько* они разнообразны. От дружелюбных к пользователю до сознательно эзотеричных... поражает, что они происходят от одной подборки программ, однако это есть факт. Итак, какой же выбрать мне?

Ben

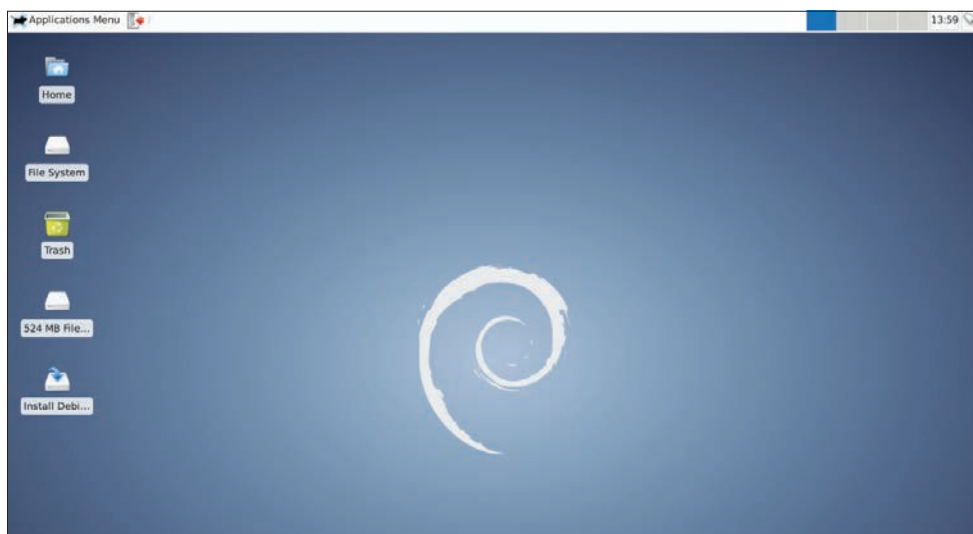
» Важно ВНИМАНИЕ!

Порченные диски

В маловероятном случае какого-то дефекта вашего LXF DVD обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

Дистрибутив Linux

Debian 7



Ни один из крупных дистрибутивов не привлекает к себе столь мало внимания, как Debian. Возможно, это не выглядит комплиментом, но тем не менее — комплимент. Он не на слуху, потому что не суетится, не делает глупостей, ничего не меняет просто по прихоти и не стремится к известности ради известности. Нет, это прочный как скала дистрибутив, просто делающий то, что дистрибутивам и полагается.

Но, к сожалению, в программе установки есть ошибка. Если вы получите отказ на стадии Install Bootloader, вернитесь назад и введите "No" в ответ на вопрос «Установить программу загрузки Grub в master boot record?»; затем следует вручную ввести устройство, где будет устанавливаться Grub (обычно это `/dev/sda`). После этого установщик будет работать нормально.

Дистрибутивы Linux

Mageia

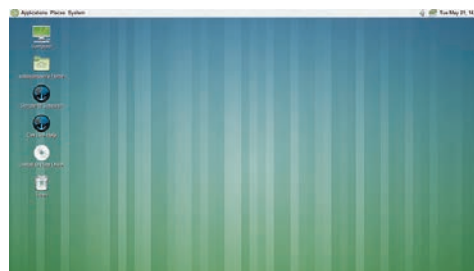
Несколько лет автор был счастливым пользователем Mandrake. До того, как я разобрался во внутреннем устройстве Linux, и до того, как Интернет заполнился статьями How-To обо всем подряд, дистрибутив Mandrake дал мне все, что мне было нужно, и тем способом, как мне было удобно.

Mageia продолжает добрую традицию. Возможно, основным отличием этого дистрибутива является Mageia Control Center, предоставляющий доступ к настройке вашего компьютера из одного центра.

Sabayon

Захоти мы составить рейтинг дистрибутивов в этом месяце, расположив их от разумного до крутого, разумнейшим стал бы Debian, затем Mageia, а Sabayon

признали бы самым крутым. Не то что он неразумен, с этим у него порядок... но другие разумнее. И этот фактор крутизны как раз и выделяет Sabayon.



» Sabayon: одаренное дитя Linux.



Новичок в Linux? Начните отсюда!

- » Что такое Linux? Как его установить?
- » Есть ли в нем эквивалент MS Office?
- » Зачем нужна командная строка?
- » Как устанавливать программы?

Ответы есть в [Index.html](#) на диске.

Из журнала

LXF64

Вернитесь в прошлое с этим полным старым выпуском *Linux Format*.

Ответы

Все ваши проблемы с открытым кодом решаются именно здесь.



LXFHotPicks

Ascii Design

Великое искусство не обязано быть реалистичным. Отвернитесь от пикселей — и назад, к старой школе.

Coverseen

Графическая оболочка для *Image Magick*, упрощающая жизнь и запоминание всех опций.

CosmoSmash

Стрелялка в стиле 1980-х на основе классической игры *AstroSmash*.

Crayonizer

Вы когда-нибудь мечтали, чтобы ваш терминал хоть чуть-чуть радовал глаз? *Crayonizer* его слегка подкрасит.

NCDU

Основанный на *Curses* инструмент использования диска, работающей на удаленном сервере через простое соединение SSH.

QMENTAT

Арифметическая игра для возбуждения серых клеточек.

RText

Текстовый редактор Java, чтобы можно было заниматься программированием одинаково на всех платформах.

TMSU

Избавьтесь от тирании иерархической файловой системы, обратившись к тэгам.

Toppler

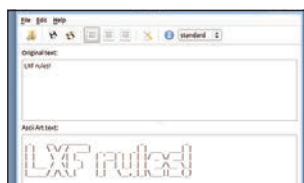
Новая реализация старой доброй игры *Tower Toppler*.

Typert

Отточите свои навыки набора текста.

ZedLog

Кросс-платформенный клавиатурный шпион.



» *Ascii Design* переводит ваши слова в произведения искусства старой школы.

И еще!

Системные инструменты

Главное

Checkinstall Установка tar-архива с помощью менеджера пакетов.

GNU Core Utils Основные утилиты, которые должны быть в каждой операционной системе.

Hardinfo Инструмент для тестирования системы.

Plop Простой менеджер загрузки для запуска операционных систем.

RaWrite Создавайте загрузочные диски в Windows.

SBM Независимый от ОС менеджер загрузки с простым в использовании интерфейсом.

Wvdial Соединяйтесь с Интернетом через телефонный модем dial-up.

Чтение

Книжная полка

The Cathedral and the Bazaar [Собор и Базар] Классический текст Эрика С. Реймонда [Eric S Raymond], объясняющий преимущества открытой разработки.

Linux Kernel in a Nutshell [Ядро Linux в двух словах] Введение в ядро, написанное великим магистром Греггом Кроа-Хартманом [Greg Kroah-Hartman].

Справочник администратора Debian Руководство для системных администраторов.

Словарь Linux Linux от А до Z.

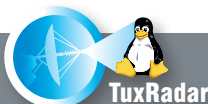
Окунитесь в Python Мастер-класс по этому популярному языку.

Руководство по созданию скриптов Bourne Shell Начните осваивать скрипты оболочки.



» Гуру ядра и писатель Грег Кроа-Хартман.

Подкасты



S5 E8 Последний Рубеж + 1

Новости [News] Уязвимости ядра и Linux на космической станции.
Открытое голосование [Open Ballot] Где бы вы применили Linux?
Открытия недели [Discoveries of the week] Canaima, Fonts, IRC, Kate, BitTorrent sync и SSH с X.

S5 E7 Ядерный Сондерс

Новости [News] Ветеран LXF Майк Сондерс вернулся ради одного эпизода.
Открытое голосование [Open Ballot] А вы бы стали платить за Linux?
Открытия недели [Discoveries of the week] Retro Arch, Into The Dead, Terminator, IRC.



Пропустили номер?



Закажите его через «Линуксцентр» по адресу www.linuxcenter.ru! Журналы доставляются и в печатной, и в электронной форме, так что с момента открытия браузера до получения нужного вам выпуска LXF может пройти не более нескольких минут!

Прямо сейчас для заказа доступны следующие номера:

LXF170 Май 2013



250 руб.

- » **Взлом Сети** Учимся парировать атаки
- » **Дистрибутивы для продвинутых** Характер нордический
- » **Samba 4** Докладываем: есть совместимость с Active Directory!
- » **Kazan** Рабочий стол как герой кинофильма

LXF DVD: PCLinuxOS, Porteus 2.0 и еще 3 дистрибутива, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_170/
PDF-версия: shop.linuxformat.ru/elxf_170/

LXF171 Июнь 2013



250 руб.

- » **Звездные полсотни** Дистрибутивы Linux на любой вкус
- » **Инфографика для всех** Выбираем достойный инструмент
- » **Secure Boot** Защита или приманка в капкане?
- » **Web-приложения «под ключ»** Развернем за пару щелчков

LXF DVD: OpenSUSE 12.3, Trisquel и игровой дистрибутив LXF, 10 книг о Linux (на английском языке), лекция Ричарда Столлмена, горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_171/
PDF-версия: shop.linuxformat.ru/elxf_171/

LXF172 Июль 2013



250 руб.

- » **Ох уж этот Ubuntu** Все — прогресса ради
- » **Linux на планшете** Не Android'ом единым
- » **Trisquel** Рожденный свободным
- » **Tizen** Мобильные устройства, откройтесь!
- » **Детям, детям** Поменьше бы чахли за компом

LXF DVD: Ubuntu Privacy Remix, Kubuntu, Xubuntu, Manjaro, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_172/
PDF-версия: shop.linuxformat.ru/elxf_172/

А чтобы не упустить ничего, оформите подписку! Все, кто подписался на печатную версию журнала через www.linuxformat.ru/subscribe или www.linuxcenter.ru, получают электронную версию в подарок! Через shop.linuxformat.ru вы также сможете приобрести предыдущие выпуски LXF (если успеете).

Подписывайтесь на www.linuxformat.ru/subscribe/

Телефоны отдела подписки:

- » Санкт-Петербург (812) 309-0686
- » Москва (499) 271-4954



Linux Format ВКонтакте

Вступайте
в нашу группу
vk.com/linuxform

На странице
журнала Linux Format
ВКонтакте вы найдете:

- » Множество новостей о Linux
- » Статьи из архива LXF
- » Анонс свежего выпуска LXF и часть статей из него
- » Живое общение и онлайн-консультацию по подписке на наше издание

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 взглянуть именно туда.

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
- » **имя_программы-1.0.1.tar.gz** — обычно это исходный код;
- » **имя_программы-1.0.1.tgz** — тот же файл, что и выше этажом 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!

А ТАКЖЕ: Developer's Assistant, OpenShift Origin и поддержка 3D-печати

Fedora 19

Ядро 3.9 » Gnome 3.8 » KDE Plasma Workspaces 4.10 » MATE Desktop 1.6
64-разрядная сборка



А ТАКЖЕ: text 2.0.7, AsciiDesign 1.0.1, 10 книг о Linux и многое другое



debian

Версия 7
Wheezy



Август 2013
LXF DVD 173

LINUX
FORMAT

Август 2013
LXF DVD 173

LINUX
FORMAT

Содержание

Сторона 1

ДИСТРИБУТИВЫ

- Debian 7 Wheezy** с рабочим столом XFCE, 32-разрядная сборка (загрузка с LXFDVD)
- Debian 7** Загрузочные диски для установки по сети, 32- и 64-разрядные сборки (ISO-образы)
- Magpie 3** Новая версия ответвления Mandriva, 32-разрядная сборка (загрузка с LXFDVD)
- Sabayon 13.04** Дистрибутив на базе CentOS с системой установки двоичных пакетов, 32-разрядная сборка (загрузка с LXFDVD)

Сторона 2

ДИСТРИБУТИВЫ

- Fedora 19** Установочный DVD, 64-разрядная сборка (загрузка с LXFDVD)
- ДОКУМЕНТАЦИЯ — 10 КНИГ О LINUX (НА АНГЛИЙСКОМ ЯЗЫКЕ)**
- Bash Scripting** Подробное руководство по программированию в Bash
- Bourne Shell Scripting** Начальное руководство по программированию в Bash
- Cathedral Bazaar** Классический текст Эрика Раймонда [Eric S Raymond] «Сбор и базар»
- The Debian Administrator's Handbook** Руководство администратора, написанное разработчиками Debian
- Dive Into Python** Учебник по программированию на Python
- Intro to Linux** Начальное руководство по Linux

- Linux Dictionary** Словарь Linux, объясняющий специфическую терминологию
- Linux Kernel in a Nutshell** Описание ядра Linux, созданное одним из его выдающихся разработчиков — Грегом Кроа-Хартманом [Greg Kroah-Hartman]
- System Administrators Guide** Руководство по базовому администрированию Linux
- GNU Tools Summary** Руководство по работе в командной строке и обзор основных утилит GNU

NOTPICKS

- AsciiDesign 1.0.1** Программа создания произведений ASCII-графики
- Converseen 0.6.2** Пакетный конвертор изображений
- CosmoSmash 1.4.7** Игра, космическая стрелялка в стиле 1980-х

- Crayonizer 0.2** Программа раскраски командной строки
- NCSDU 1.10** (NCurses Disk Usage) Анализатор использования диска
- OMentat 0.9.0** Программа обучения счету в уме
- RhexT 2.0.7** Редактор текстов для программистов
- TMSU 0.2.0** Органайзер файлов
- Toppler 1.6** Игра-платформер
- Typex 0.4.0** Программа обучения быстрой набору текста
- ZedLog 0.1b** Клавиатурный шпион (Кейлоггер)

Пожалуйста, перед использованием данного диска ознакомьтесь с опубликованной в журнале инструкцией!

КОММЕНТАРИЙ: Присылайте ваши пожелания и предложения по электронной почте: info@lxf.com

ДВОИЧНЫЕ ДИСКИ: В маловероятном случае обнаружения дефектов на данной диске, обращайтесь, пожалуйста, по адресу disk@lxf.com

Настоящий диск тщательно тестировался и проверялся на всех старых платформах, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать апликативный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных в случае жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, предоставленных нашей программой или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, скачайте программу для загрузки с нашего сайта www.linuxformat.ru

Тираж изготовлен ООО «Уральский электронный завод», 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПР ВАР 77-15.

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-*root*. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле */etc/default/cdrecord*. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени *root* и выберите вкладку *Burn* и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на *Combust*! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам 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 разработчика дистрибутива.

КЕЙ
ТЕХНИКА И ЧЕЛОВЕК

СПОРТИВНЫЕ
ГАДЖЕТЫ

ТВОИ ПОМОЩНИКИ В ТРЕНИРОВКАХ

Тонометры Термометры Весы

Узнай подробнее на key.ru. Товар сертифицирован. Количество товара ограничено.

V МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ
ПО ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

ИНФОФОРУМ ШАНХАЙ

ДОВЕРИЕ И БЕЗОПАСНОСТЬ
В ИНФОРМАЦИОННОМ ОБЩЕСТВЕ

23-28
СЕНТЯБРЯ

Участник
SIBT-2013

CHINA.INFOFORUM.RU

LINUX FORMAT

Главное в мире Linux

Как разместить рекламу в разделе Classifieds?

1/4 полоса (210 × 297 мм)	165 200 руб.
1/2 полосы горизонтально (197 × 144 мм)	88 500 руб.
1/2 полосы вертикально (102 × 278)	88 500 руб.
1/4 полосы вертикально (98 × 138 мм)	53 100 руб.
Фотоблок (44 × 113 мм)	15 000 руб.

Тел.: +7 812 309 06 86

Цены указаны с учетом НДС

UnixEducationCenter

redhat
Certified Training Partner

Обучение & Сертификация **RHCSA + RHCE**
Впервые в Санкт-Петербурге:

RHCV4®

EX318 EX401 EX436 EX442

RHCDS®

Санкт-Петербург, Черноморский переулок, дом 4
Телефон: +7 (812) 611-15-75
mail@unixedu.ru www.unixedu.ru

ГНУ/Линуксцентр приглашает на работу!

Linux center
www.linuxcenter.ru

ВАКАНСИЯ: Разработчик систем электронного документооборота

ТРЕБОВАНИЯ:

- » Опыт внедрения и сопровождения систем электронного документооборота на базе Alfresco
- » Умение описать бизнес-процесс
- » Знание SQL, понимание принципов построения и функционирования баз данных
- » Знание СЭД, отличных от Alfresco, приветствуется

ОБЯЗАННОСТИ:

- » Разработка систем с нуля, сопровождение
- » Настройка системы в соответствии с требованиями бизнеса
- » Написание инструкций для конечных пользователей

ПОДРОБНЕЕ: www.linuxcenter.ru/vacancy/

Футболки GNU/Linux

БОЛЕЕ 100 ВИДОВ. ВСЕ РАЗМЕРЫ И ЦВЕТА

WWW.LINUXCENTER.RU/SHOP/GIFTS/



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 3000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор
Кирилл Степанов info@linuxformat.ru
Литературный и выпускающий редактор
Елена Толстякова
Переводчики
Елена Ессяк, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова, Ирина Шулакова
Редактор диска
Кирилл Степанов
Верстка, допечатная подготовка
Сергей Рогожников
Технический директор
Денис Филиппов
Директор по рекламе
Владимир Савельев advert@linuxformat.ru
Генеральный директор
Павел Фролов
Учредители
Частные лица
Издатель
ЗАО «Мезон.Ру»

Отпечатано в ООО «Ланинь»

188330, Ленинградская обл., Гатчинский р-н, пос. Сиверский, Вокзальная ул., 4
Заказ 8072

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Грэм Моррисон [Graham Morrison] graham_morrison@futurenet.com
Заместитель редактора Эндрю Грегори [Andrew Gregory] andrew.gregory@futurenet.com
Редактор диска Бен Эверард [Ben Everard] ben_everard@futurenet.com
Выпускающий редактор Крис Торнетт [Chris Thornett] chris.thornett@futurenet.com
Художественный редактор Эфраин Эрнандес-Мендоса [Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ

Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Бен Эверард [Ben Everard], Марко Фиоретти [Marco Fioretti], Эндрю Грегори [Andrew Gregory], Стюарт Джарвис [Stuart Jarvis], Грэм Моррисон [Graham Morrison], Джонатан Робертс [Jonathan Roberts], Майк Сондерс [Mike Saunders], Маянк Шарма [Mayank Sharma], Евгений Балдин, Артем Зорин, Андрей Прахов, Петр Семилетов, Андрей Ушаков, Алексей Федорчук

Художественные ассистенты

Стейси Блэк [Stacey Black], Анна Мак-Ги [Anna McGee]
Иллюстрации Шейн Коллиндж [Shane Collinge], Саймон Миддлвик [Simon Middleweek], Ели Уолтон Иллюстрация [Ely Walton Illustrations]

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Тел. +44 01225 442244, email: linuxformat@futurenet.com

РОССИЯ:

Санкт-Петербург (редакция):
Лиговский пр., 50, корп. 15
Тел. +7 (812) 309-0686
Представительство в Москве:
Красноказарменная ул., 17, м. «Авиамоторная» (в помещении АТС МЭИ)
Тел./факс +7 (499) 271-4954
По вопросам сотрудничества, партнерства, оптовых закупок: partner@linuxcenter.ru

Авторские права: статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, 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 повреждение или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds]. "GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт <http://www.futureplc.com>



© Linux Format 2005
© Future Publishing Ltd 2005



BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

В сентябрьском номере Руки на стол!

Освежите свой компьютер, подобрав созвучную себе рабочую среду, с помощью нашего исчерпывающего руководства к современным рабочим столам.

Дистрибутивы для анонимов

Повернитесь к Интернету эквивалентом маски Дракулы и забудьте об опасениях, что кто-то вторгнется в вашу личную жизнь.

Громкие слова

GNU, MBR, GUI, HURD, TTY... что все это означает? И откуда взялось извращенное пристрастие к аббревиатурам/акронимам?

Содержание будущих выпусков может меняться — вдруг мы погрязнем в проблеме выбора каши на завтрак...



Mandriva Linux — один из самых популярных дистрибутивов GNU/Linux в мире. Главные преимущества Mandriva — дружелюбный интерфейс, простота настройки, возможность быстрой адаптации пользователей, ранее не знакомых с этой ОС, совместимость с широким спектром программного и аппаратного обеспечения.

Корпоративные продукты Mandriva Linux

Mandriva 2010.2 Powerpack

Дистрибутив Mandriva 2010.2 Powerpack включает набор офисных и серверных приложений, и подходит для установки на офисной или домашней рабочей станции и на сервере. Дружелюбный интерфейс, простота настройки Mandriva Powerpack, совместимость с широким спектром аппаратного обеспечения и совместимость с «1С:Предприятие» обеспечивают корпоративным пользователям возможность легкого перехода с Windows на GNU/Linux.

Mandriva Enterprise Server 5

Mandriva Enterprise Server 5 (MES 5) — это надежный и производительный дистрибутив GNU/Linux для корпоративного сервера. MES 5 поможет вам снизить текущие расходы и упростить инфраструктуру. В MES 5 интегрированы серверные разработки программистов Mandriva, в том числе сервер каталогов пользователей Mandriva Directory Server, а также ведущие свободные серверные приложения, которые помогут вам с минимумом затрат времени и энергии настроить и поддерживать необходимые вам серверы. Срок поддержки дистрибутива — 5 лет.

Сертифицировано ФСТЭК

Дистрибутивы Mandriva Linux сертифицированы по требованиям ФСТЭК по 5 классу для СВТ и 4 уровню контроля НДВ, что дает возможность использовать их для обработки конфиденциальной информации в автоматизированных системах класса до 1Г включительно и обработки персональных данных в информационных системах класса до К2 включительно.

- **Mandriva 2008 Spring Powerpack** — дистрибутив для рабочей станции или небольшого сервера.
- **Mandriva Corporate Server 4 Update 3** — дистрибутив для создания корпоративного сервера.
- **Mandriva Flash** — дистрибутив GNU/Linux, загружающийся и работающий прямо с USB-носителя.

EduMandriva — свободное ПО для образования

- Создано с участием российских преподавателей и методистов.
- Все ПО, необходимое для преподавания информатики.
- Методические материалы.

Наименование	Стоимость, руб.
Корпоративные продукты Mandriva	
Mandriva Linux 2010.2 Powerpack (DVD-Box)	1 300
Услуга подписки на Mandriva Enterprise Server 5 на 1 год, базовый уровень (с физическим носителем)	13 300
Услуга подписки на Mandriva Enterprise Server 5 на 3 года, базовый уровень (с физическим носителем)	34 800
Продукты Mandriva для образования	
Комплект программного обеспечения Mandriva Linux и EduMandriva для школ	3 500
Сертифицированные ФСТЭК продукты Mandriva	
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack на 10 рабочих мест	28 500
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack на 5 рабочих мест	14 500
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack на 1 рабочее место	4 990
Сертифицированный ФСТЭК Mandriva Corporate Server 4.0 Update 3	10 050

С вопросами по продуктам Mandriva обращайтесь в ГНУ/Линуксцентр!

MANDRIVA УЖЕ ИСПОЛЗУЮТ:
 МВД РФ, Минздравсоцразвития РФ,
 Минфин республики Саха (Якутия),
 Правительство Московской области,
 администрация Черниговского района,
 Приморского края, ОАО «Морион»,
 сеть магазинов «Компьютер-центр
 «КЕЙ», группа компаний «ИМАГ»,
 компания «Азбука мебели»,
 и многие другие.

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru



Новое поколение средств защиты

Межсетевые экраны ССПТ, не имеющие IP-адреса

ССПТ-2 — это сертифицированное ФСТЭК, ФСБ и ГАЗПРОМСЕРТ средство защиты информации нового поколения, реализующее функции межсетевого экрана, но при этом остающееся «невидимым» для любых протоколов и тестовых воздействий, что достигается за счет отсутствия физических и логических адресов на его фильтрующих интерфейсах. ССПТ-2 **невозможно обнаружить никакими известными средствами удаленного мониторинга сети.**

Скрытность функционирования межсетевого экрана повышает надежность системы защиты в целом и существенно упрощает процедуру установки ССПТ-2 в компьютерные сети и функционирующие на их основе информационные и телематические системы.

Защита для высокоскоростных корпоративных сетей Ethernet 100/1000 Мбит/с

Сертифицированы ФСТЭК и ФСБ (3-й класс защиты)

На базе процессоров с 64-разрядной многоядерной архитектурой



Назначение устройства

Основное средство защиты для реализации различных политик информационной безопасности с помощью:

- фильтрации пакетов на канальном, сетевом, транспортном и прикладном уровнях;
- управления транспортными соединениями между отдельными узлами ЛВС или виртуальной ЛВС (VLAN);
- контроля контента данных на прикладном уровне с учетом направления, времени и типа протоколов передачи трафика.

Дополнительное устройство защиты для:

- обеспечения безопасности функционирования ранее установленных в компьютерной сети средств защиты и устройств маршрутизации;
- мониторинга трафика с возможностью анализа данных регистрации пакетов по различным критериям и интеграции с IDS;
- обеспечения функционирования сетевых распределенных телематических приложений и GRID-ресурсов.

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center