

# LINUX FORMAT

Главное в мире Linux

Сентябрь 2012 №9 (161)

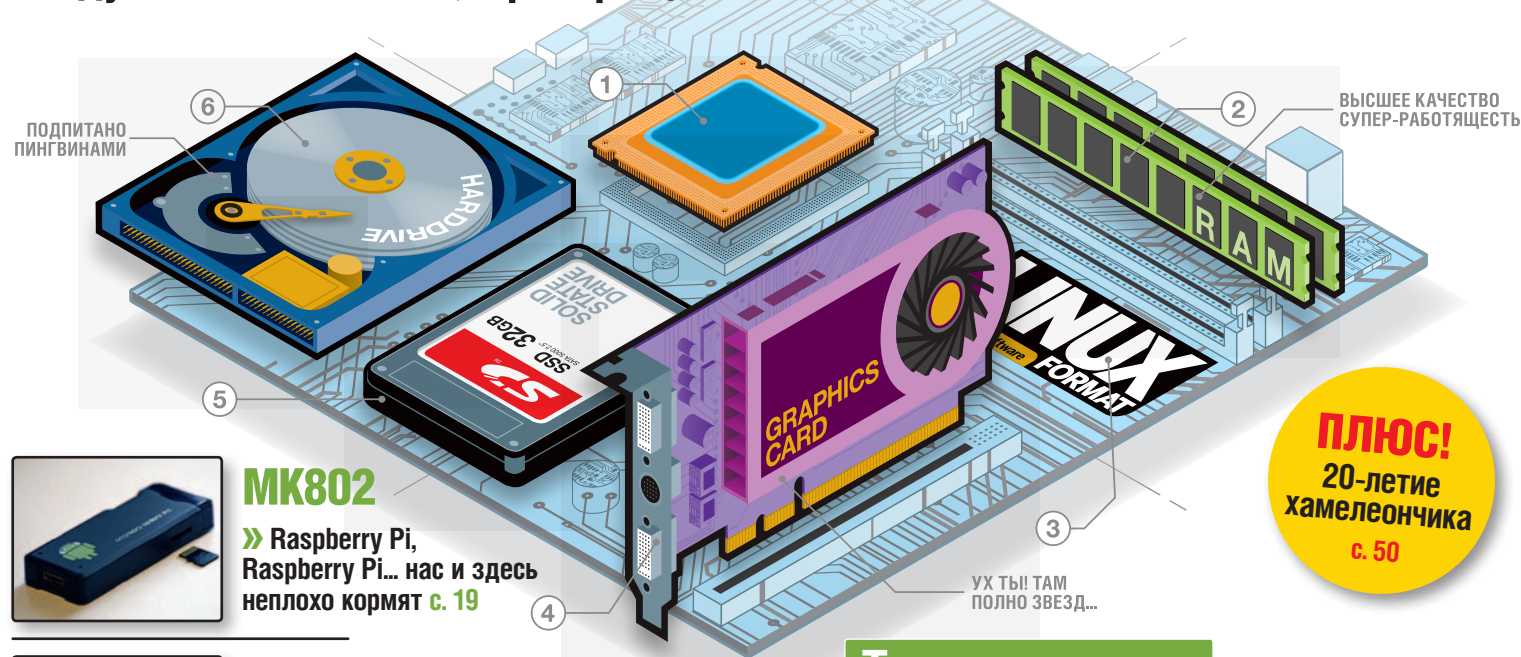


**Raspberry Pi: Рулим Гвидо ван Роботом с. 66**

**3 ДИСТРИБУТИВА!**  
Rosa 2012 (Возрождение Mandriva),  
Vector 7, Fedora LXDE, FreeNAS и более того!

# Обновляйтесь!

Получите лучший комплект оборудования для своей установки, следуя нашим советам, и распрощайтесь с головной болью несовместимости!



ПОДПИТАНО ПИНГВИНАМИ (6)

1

2

3

4

5

ВЫСШЕЕ КАЧЕСТВО СУПЕР-РАБОТАЮЩЕСТЬ (2)

УХ ТЫ! ТАМ ПОЛНО ЗВЕЗД... (3)

**ПЛЮС!**  
20-летие хамелеончика с. 50



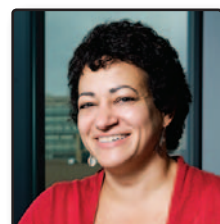
### МК802

» Raspberry Pi, Raspberry Pi... нас и здесь неплохо кормят с. 19



### Курсы Red Hat

» Украсьте свое резюме при помощи нашего обзора с. 46



### Глас Canonical

«Идея нетбука была неплохой, но вот оборудование подкачало»

Джейн Силбер: главный босс Ubuntu с. 38

### Также в номере...

#### VirtualBox 101

» Запускайте ПК внутри ПК внутри ПК – до головокружения... с. 62



#### Анатомия жесткого диска

» Вскроем внутренности вашей Linux-машины с. 42

#### Строим классный сервер

» И предоставляем всем доступ ко всему с. 76

Шифрование данных  
**Файлы под защитой**

» Поместите приватные данные на безопасный USB-брелок

Онлайн-обучение  
**Moodle – мигом**

» Разработайте собственный курс и обучите ему весь мир

Управление контентом  
**Лучшая CMS**

» Забросьте WordPress ради блогов

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



# Купите свой дом в Испании на Коста Бланка у моря или в горах

Ипотека до 70%.

Полное юридическое сопровождение



Апартаменты  
от 50 000 евро



Таунхаузы  
от 90 000 евро



Виллы  
от 110 000 евро

  
KD Gest.com

[www.kdgest.com](http://www.kdgest.com)

+34 965 040 795

+34 678 765 038

# Red Hat Enterprise Linux

предоставляет вам **производительность, масштабируемость, безопасность и надежность**, ранее доступные только на очень дорогих платформах

Самая популярная в мире Linux платформа для бизнеса

Обеспечивает высокую производительность, надежность, масштабируемость и безопасность

Сертифицирована ведущими производителями оборудования и разработчиками ПО



Совместима с широким спектром оборудования от рабочих станций до серверов и мэйнфреймов

Обеспечивает одинаковые условия работы приложений при использовании в физической, виртуальной и облачной средах

Пользователи RHEL экономят на оборудовании, лицензиях на программное обеспечение и эксплуатационных расходах



## ГНУ/Линуксцентр — Linux-эксперт для вашего бизнеса

- Premier Business Partner компании Red Hat
- 12 специалистов по разработке и внедрению, сертифицированных компанией Red Hat
- Более 100 клиентов, использующих Red Hat
- 10 лет на рынке

## Red Hat — ведущий серверный дистрибутив Linux

- Более 15 лет промышленного использования
- Свыше 80% рынка корпоративного Linux по данным CIO Insight
- 5 лет среди лучших вендоров
- Выгодная совокупная стоимость владения (TCO)
- Поддержка в течение 10 лет

### Специальное предложение!

Закажите Red Hat Enterprise Linux в ГНУ/Линуксцентре и получите в подарок книгу «Полное руководство пользователя Red Hat Enterprise Linux»



Москва  
+7 (499)

271-49-54

Санкт-Петербург  
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. [www.linuxcenter.ru](http://www.linuxcenter.ru)

Linux  center

# 1С:ПРЕДПРИЯТИЕ 8

## Новое поколение систем автоматизации управления и учета

- 1С:Управление производственным предприятием
- 1С:Бухгалтерия
- 1С:Зарплата и управление персоналом
- 1С:Управление торговлей
- Отраслевые и специализированные решения

# Теперь работает в Linux!

Для запуска «1С:Предприятия» в Linux необходимо наличие WINE@Etersoft, который входит в состав дистрибутива Mandriva Linux 2011 Powerpack



Москва  
+7 (499)

271-49-54

Санкт-Петербург  
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. [www.linuxcenter.ru](http://www.linuxcenter.ru)



## Что мы делаем

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

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

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

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



## Кто мы

Какое бы устройство вы изобрели, чтобы улучшить наш мир?



**Гэри Уокер**  
Auto Edit 2000 год – тогда я исполнил свою мечту: ранний уход на пенсию и работа водителем автобуса на Мальте.



**Эндрю Грегори**  
Плазменную пушку, которая взаправду будет уничтожать планеты. Белл и Брейбен, зачем вы нам лгали?



**Эфраин Эрнандес-Мендоса**  
Детектор лжи, чтобы спрашивать Ника про срок сдачи материалов.



**Бен Эверард**  
Передачик сигнала электрошока штату админов Минздрава при каждой отмене моих назначений.



**Маянка Шарма**  
Настоящий голубой Shell, на случай, если с бензином в Нью-Дели ни с того ни с сего вдруг попохнет.



**Джонатан Робертс**  
Космический корабль, чтобы сбежать в другой мир и отстроить все заново.



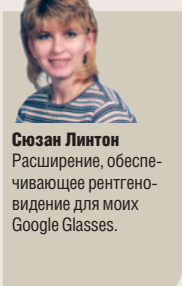
**Майк Сондерс**  
Совок для топлива, который подойдет моему космолету Panther Clipper.



**Валентин Синецын**  
Полностью поддерживаемое в Linux. Чтобы не пришлось искать драйверы и ядро посвежее.



**Ник Вейч**  
Устройство для автоматической отделки моих программ на Python и оформления моих учебников по GIMP.



**Сюзан Линтон**  
Расширение, обеспечивающее рентгеновидение для моих Google Glasses.



**Шашанк Шарма**  
Генератор акустического удара – применять, когда почтальон доставит мне счет за газ. Упрек вам, электроснабженцы.



**Нейл Боткин**  
Пищевой автомат, раз в час выдающий бутерброд со стейком, как в «Облачно, возможны осадки в виде фрикаделек».



## «Хард» и «софт»

» В данном номере много материалов о «железе», причем самом разном – от миниатюрного MK802, выполненного в форм-факторе USB-брелка, и Raspberry Pi (чуть побольше) до серверов на базе самых современных многоядерных процессоров архитектуры x86-64. И на всем этом работает Linux.

Самый маленький из описываемых компьютеров имеет «на борту» 1 ГБ оперативной памяти, 4-ГБ флэш-диск, стоит 74 доллара США и легко помещается в кармане. А вот первый компьютер, который мне довелось «админить» четверть века назад – ЕС-1061 – имел «целых» 8 МБ памяти и 15 дисков по 200 МБ. Стоила эта установка примерно 4 миллиона долларов США (по курсу того времени, в 60 полновесных копеек СССР за доллар) и занимала помещение примерно в 150 квадратных метров. А теперь вот – в карман...

За эти четверть века случилась революция. Процесс разработки «железа» ускорился на порядки, и само оно подешевело в той же пропорции. Каждый месяц электронная индустрия стран Юго-Восточной Азии выпускает на рынок все новые и новые «девайсы», один чуднее другого. А вот процесс разработки «софта» ускорился не сильно и дешевле не стал. Поэтому, кстати, Linux и портируется на все возможные платформы – так быстрее, а значит, и дешевле.

Так что встает резонный вопрос: что сегодня более «хард» с точки зрения разработки – «железо» или «софт»?

**Кирилл Степанов**

Главный редактор

» [info@linuxformat.ru](mailto:info@linuxformat.ru)

## Как с нами связаться

Письма для публикации: [letters@linuxformat.ru](mailto:letters@linuxformat.ru)

Подписка и предыдущие номера: [subscribe@linuxformat.ru](mailto:subscribe@linuxformat.ru)

Техническая поддержка: [answers@linuxformat.ru](mailto:answers@linuxformat.ru)

Проблемы с дисками: [disks@linuxformat.ru](mailto:disks@linuxformat.ru)

Общие вопросы: [info@linuxformat.ru](mailto:info@linuxformat.ru)

Вопросы распространения: [sales@linuxformat.ru](mailto:sales@linuxformat.ru)

Web-сайт: [www.linuxformat.ru](http://www.linuxformat.ru)

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

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

# Содержание

Данный номер в основном спонсирован Red Hat... это мы так понадеялись...

## Обзоры

### Zorin OS ..... 16

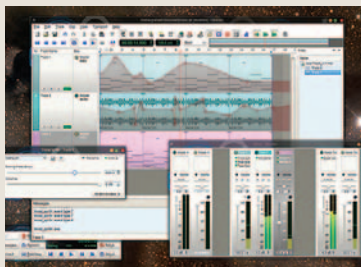
Еще один Linux-дистрибутив, нацеленный на бизнес. А вы отдали бы 15 евро за диск с Linux?

### Fit PC 3 Pro ..... 17

Крошечные низковольтные машинки – сейчас последний писк моды; данный конкретный экземпляр работает под Mint и выглядит круто.

### Qtractor ..... 18

Производит аудио! Работает с MIDI! И даже тянет плаг по раскишему полю! Истинная отрада фермера.



» **Qtractor**: вот так фермеры на селе пишут сладковзвучную музыку.

### MK802 ..... 19

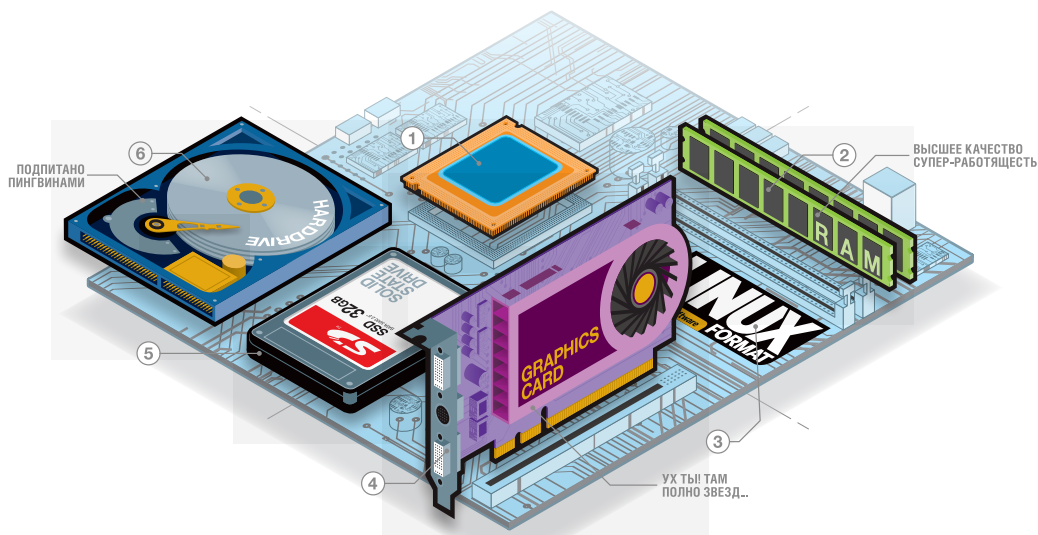
Микрокомпьютер на Android собрался затмить Raspberry Pi. А не слабо предоставить домашний кинотеатр, умещающийся на ладони?



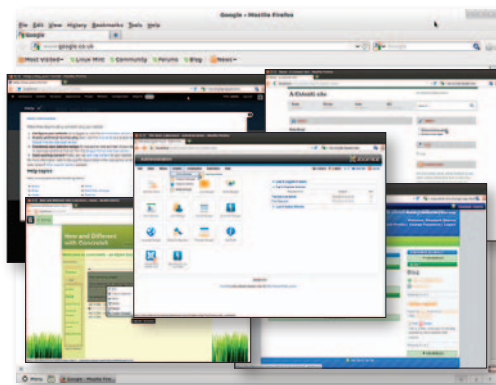
» Масштаб рисунка легко оценивается по размеру SD-карты.

### Mate ..... 20

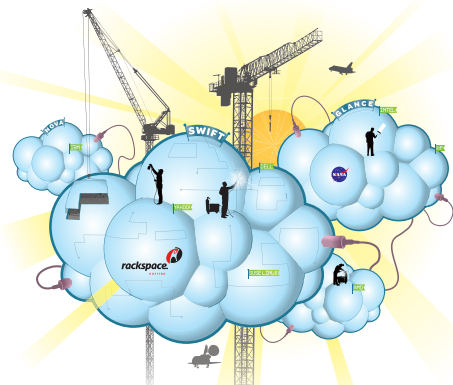
Дело Gnome 2 не погубило – знамя подхватил проект Mate. Что не мешает ему преспокойно уживаться на одном компьютере с Gnome 3.



**Обновите свое железо: мы тестируем графические карты, материнские платы, CPU и SSD! с.30**



**Сравнение: Системы управления контентом с.24**



**Что за штука – OpenStack? с.54**

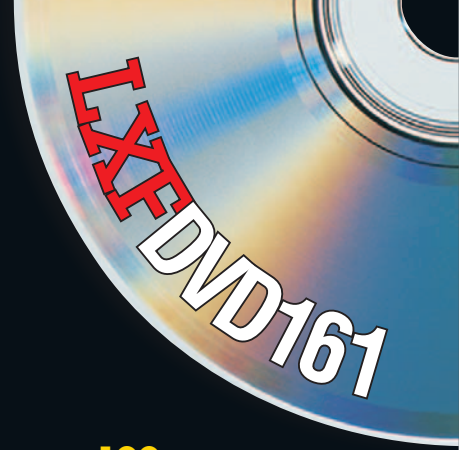
## Люди говорят



« Сообществу проще развиваться при четкой концепции, и так работает Ubuntu »

**Джейн Силбер из Canonical – о доброй диктатуре с.38**

# На вашем бесплатном DVD



## Rosa

» Новое направление KDE

## Fedora LXDE

» Невесомое чудо

**ПЛЮС:** Горячие новинки и коды к учебникам... **с. 102**

## Ищите в этом номере



### Конференция в Обнинске ..... 13

Разработчики ПО на всех платформах, объединяйтесь!

### Анатомия жесткого диска ..... 42

Эта вращающаяся пластина внутри вашего ПК.

### Red Hat ..... 46

Стоит ли разоряться на официальные курсы?



## Пропустили номер?

Узнайте на с. 104, как получить его прямо сейчас!



## Учебники

**Виртуализация**  
**VirtualBox** ..... 62  
Соберите, установите и запустите виртуальную машину с нашим другом *VirtualBox*.

**Хакерство**  
**Raspberry Pi** ..... 66  
Заполните своего мини-робота и помыкайте им через *Guido van Robot*.

**Шифрование**  
**TrueCrypt при вас** ..... 70  
**Нейл Ботвик** демонстрирует портативный метод шифрования данных.

**Образование**  
**Moodle для любого** ..... 72  
Разработайте собственный университетский курс и дерите по £9000 со студента за 4-часовое общение и пропуск в библиотеку.

**Хранение данных**  
**FreeNAS** ..... 76  
Следуйте мудрым советам **Майка Сондерса** и не тратьтесь на дорогостоящие NAS.

**Электроника**  
**Arduino** ..... 80  
Еще один повод не связываться с реальными 3D-дружками: лучше чиркать с виртуальными онлайн.

**Языки программирования**  
**Erlang** ..... 84  
Функции, способных упростить вам жизнь, накопилось уже полным-полно.

**Аудит сети**  
**NMap** ..... 88  
Узнайте всю подноготную про сканируемый объект: про протоколы, ОС, версии сервисов...

## Постоянные рубрики

### Новости ..... 4

Вышел экспериментальный менеджер пакетов *Deepsolver*, *Raspberry Pi* дождался серийного выпуска, *Nokia* передала *Qt* на сторону, а HP открыла коды *webOS*.

### Android ..... 22

*CyanogenMod* – альтернатива *Android*; разработана техника атаки на телефоны с *Android*.

### Сравнение ..... 24

Какая система управления контентом лучше?

### Интервью LXF ..... 38

**Джейн Силбер** заверяет, что разработка *Ubuntu* была и будет прозрачной и, чего доброго, доберется и до утюгов. Все к тому идет.

### Что за штука ..... 54

*OpenStack* – Linux облачных вычислений. Сказать более точно мы не рискуем.

### Рубрика сисадмина ..... 56

Перебежим по мостику между *Windows* и *Linux*: *Bitvise Tunnelier*.

### Ответы ..... 92

**ПРОБЛЕМЫ LINUX РЕШЕНЫ!**  
Обо всем подумает **Нейл Ботвик**, а вы можете не париться.

### Hotpicks ..... 96

Отведаете горяченького: лучшие в мире новинки свободного ПО.

### Диск Linux Format ..... 102

Содержимое двустороннего DVD этого месяца.

### Пропустили номер? ... 104

Отлично перелистывается перед телевизором – чтоб отвлечься от выпусков новостей.

### Школа LXF ..... 108

Жалобы педагога. А вот это уже необоснованный пессимизм!

### Через месяц ..... 112

В нашем эпическом ратоборстве дистрибутивов может быть только один победитель.



**ГЛАВНОЕ** Российский менеджер пакетов » Массовое производство одноплатника » Свято место Nokia пусто не бывает » HP открывает исходники » Gnome станет операционкой

**QT МЕНЯЕТ ХОЗЯИНА**

## 16-летняя разработка завершена



» Рубрику готовил  
**АРТЕМ ЗОРИН**

Nokia отказывается от разработки Qt и продает проект компании Digia.

**В** марте 2011 года Nokia передала обслуживание коммерческих клиентов инструментария Qt другой финской компании – Digia. А 9 июля 2012 года было объявлено о подписании соглашения, по которому Digia покупает у Nokia весь бизнес и программные технологии, связанные с Qt. Соглашение включает разработку продукта, коммерческое и open-source лицензирование и сервисное обслуживание. Финансовые детали сделки не разглашаются. Завершить ее планируется в третьем квартале нынешнего года.

Digia собирается сразу приступить к работе по добавлению возможности использования Qt на платформах Android, iOS и Windows 8. Также сообщается о включении Digia в работу по подготовке Qt 5 и участию в работе сообщества Qt Project. Одна из целей компании в рамках приобретения Qt – улучшить свою позицию в этом проекте, увеличить ассигнования в R&D и обеспечить работу Qt на новых платформах. По соглашению, до 125 разработчиков из Nokia перейдут в Digia.

Подписанное соглашение – закономерный результат отказа Nokia от всех плат-

форм для смартфонов, кроме Windows Phone, на котором Qt не работает, так как единственным поддерживаемым языком программирования для Windows Phone является C#. Напомним, что за последние два месяца из Nokia были уволены команды разработчиков платформы смартфонов N9 (MeeGo/Maemo) и ОС Meltemi на базе Linux, созданной для замены Series 40 на телефонах низкой ценовой категории.

» Штаб-квартира компании Digia в Хельсинки.



**«Digia собирается сразу приступить к работе по использованию Qt.»**

Что до возможных изменений в политике развития и распространения Qt как проекта, то компания Digia пока не намерена что-либо менять и продолжит развитие и открытого, и коммерческого варианта Qt. Также продолжится сотрудничество с разработчиками KDE. Более того, за последний год Digia зарекомендовала себя как активный участник проекта. Например, большинство исправлений в ветке Qt 4

были внесены силами сотрудников Digia. В уже озвученных планах Digia – увеличение инвестиций в Qt для настольных, встраиваемых и мобильных систем.

В любом случае, компания Nokia до продажи связанного с Qt бизнеса передала управление над проектом Qt и полномочия по принятию решений отдельному некоммерческому проекту Qt Project, в котором члены сообщества могут участвовать в развитии Qt согласно принципам меритократии. В руки Qt Project передана разработка и экспериментальной ветки Qt 5, и стабильной Qt 4. Кроме того, между компанией Nokia и организацией KDE Free Qt Foundation заключено соглашение, предусматривающее в случае ужесточения лицензионной политики, банкротства Nokia или прекращения развития Qt передачу проекту KDE права перелицензировать код Qt под лицензией BSD или любой другой открытой лицензией, чтобы продолжить разработку своими силами. Соглашение не теряет силу и при смене владельца Qt.

### Перспективы у Qt хорошие

Вскоре после объявления о продаже Qt разработчики проекта KDE получили письмо от директора по разработке и исследовательским проектам Digia с заверением о сохранении открытого характера разработки и желании укрепить сотрудничество с проектом KDE и другими представителями экосистемы Qt. Digia считает сообщество KDE локомотивом развития Qt и активным участником разработки, и намерена перевести отношения с KDE на новый уровень.

Digia планирует встречу с представителями KDE и других ведущих Qt-проектов, чтобы обсудить совместную работу и согласовать направления развития Qt. Digia также продолжит участие в Qt Project и предоставит инфраструктуру для работы ключевых систем проекта. Digia считает важным участие в разработке как можно большего количества представителей сообщества, и рассматривает Qt Project как площадку для организации сотрудничества.



# ПОДПИШИСЬ НА ЖУРНАЛ LINUX FORMAT!

2012 **LINUX**  
FORMAT  
Главное в мире Linux



Оформи в редакции  
подписку на печатную  
версию журнала  
и получи в подарок  
диск с архивом номеров,  
а также подписку  
на электронную версию  
издания в формате PDF.



**Стоимость подписки** Годовая — 2280 руб., полугодовая — 1230 руб. без учета стоимости доставки.  
**Адреса и телефоны редакции** Санкт-Петербург, Лиговский пр., 50, корп. 15, тел. (812) 309-06-86.  
Москва, Красноказарменная ул., 17, тел. (499) 271-49-54.

**Варианты доставки** Почтой по России простой бандеролью — журнал доставляют прямо в почтовый ящик  
>> Почтой по России заказной бандеролью — в почтовый ящик приходит извещение, номера выдают на почте >> Курьером  
«ГНУ/Линуксцентра» по Москве и Санкт-Петербургу >> Курьерской службой СПСР по России >> В виде PDF-файлов  
для подписчиков электронной версии.

[shop.linuxformat.ru](http://shop.linuxformat.ru)

ЭКСПЕРИМЕНТАЛЬНАЯ РАЗРАБОТКА

# Вклад программиста из ALT Linux

Вышла первая бета-версия отечественного менеджера пакетов *Deepsolver*.

В середине июля в Сети появились сообщения о новой отечественной разработке. Михаил Пожидаев из компании Альт Линукс создал экспериментальный менеджер пакетов *Deepsolver* для установки и обновления программных приложений, в перспективе способный заменить инструментарий *apt-rpm* в дистрибутивах ALT Linux. При росте числа пакетов в репозиториях сказываются ограничения масштабируемости *APT*, что негативно влияет на развитие дистрибутивов ALT Linux. Эти и другие архитектурные ограничения, присутствующие в *APT*, не дают возможности реализовать в нем ряд дополнительных функций, необходимых в современном мире для удовлетворения всех требований пользователей и разработчиков. Отличительные черты *Deepsolver*:

- » Обеспечение более высокой производительности в условиях роста пакетной базы. Например, полное разрешение зависимостей для пакета с KDE4 занимает в *Deepsolver* 0,44 сек, тогда как *apt-rpm* тратит на эту же операцию 3,52 сек.
- » Оригинальная система разрешения зависимостей, годная и для готовых систем, и для создания сборочных окружений. Планируется интеграция с библиотекой *libsolv*, созданной в рамках проекта openSUSE и внедряемой в Fedora.
- » Гибкая масштабируемая архитектура с возможностью подключения поддержки различных форматов пакетов через взаимодействие со сторонними библиотеками (базовая реализация основана на *librpm*).
- » Попытка визуально разделить для пользователя разные классы пакетов – например, логически отделить пакеты с при-

- ложениями от пакетов с библиотеками и дополнительными данными, которые часто мешают при попытке формирования списков программ для установки.
- » Расширенные возможности поиска и получения информации о пакетах.
- » Хорошо структурированный исходный код и наличие исчерпывающей документации для разработчиков.
- » Компоненты *Deepsolver* оформлены в виде разделяемых библиотек, для построения CLI/GUI предоставляется специальный API. Базовых компонентов три: *OperationCore* (установка/удаление пакетов, загрузка индексов и прочие операции, затрагивающие изменения системы), *InfoCore* (подсистема управления информацией о пакетах и организации поиска) и *IndexCore* (подсистема для сборки индексов и поддержания репозитория).

МИНИ-КОМПЬЮТЕР

# Raspberry Pi пошел в серию

Под массовое производство Raspberry Pi выпущен Linux-дистрибутив Raspbian.

Raspberry Pi сейчас у всех на слуху. И вот хорошая новость для его фанатов: сняты ограничения на заказ устройств Raspberry Pi модели B. Раньше один человек мог заказать только одно устройство; теперь ограничения сняты благодаря переходу к массовому производству, предусматривающему выпуск до 4000 устройств в день.

- Спецификация Raspberry Pi B:
- » Размер: 8,5×5,4×1,7 см. Вес 45 г.
  - » SoC-платформа: Broadcom BCM2835 700 МГц (ARM11 ARM1176JZF-S) с графическим процессором, поддерживающим OpenGL ES 2.0 и функциями ускорения декодирования видео (1080p30 H.264).
  - » 256 МБ ОЗУ.
  - » Питание: 5 В через microUSB-порт.
  - » Интерфейсы: USB 2.0, Ethernet, слот для карт SD/MMC/SDIO, 3,5-мм звуковой выход.
  - » Видеовыход: композитный выход RCA («тюльпан») и цифровой порт HDMI.

Дополнительно объявлено о выпуске Raspbian, нового специализированного дистрибутива Linux для Raspberry Pi. Дистрибутив основан на пакетной базе Debian

Wheezy, специально оптимизированной для Raspberry Pi. Рабочий стол – *LXDE*, в качестве web-браузера принят *Midori*, а для работы с мультимедиа-контентом – медиа-плеер *omxplayer*, в полной мере использующий средства аппаратного ускорения в Raspberry Pi.

Создание отдельного порта Debian для Raspberry Pi обусловлено тем, что штатный порт Debian Wheezy armhf поддерживает только процессоры с архитектурой ARMv7-A, а в Raspberry Pi используется ARMv6 с расширениями hard float, позволяя использовать только более медленный порт Debian armel (soft float). От прежней сборки для Raspberry Pi, Debian Squeeze, Raspbian отличается значительно возросшей производительностью, благодаря задействованию VFP-инструкций (Vector Floating Point) для вычислений с плавающей точкой, а также ARMv6-оптимизации для библиотечных функций.

Образ для записи на SD-карту «весит» 439 МБ. Для установки доступно около 35000 добавочных пакетов в репозиториях. Кроме Raspberry Pi, Raspbian будет работать и на других устройствах

с процессором ARMv6 и VFP-расширениями. Raspbian позиционируется как замена ранее поставляемой сборки Debian GNU/Linux, поддержка которой прекращена. Кроме Raspbian, энтузиастами развиваются специализированные сборки Fedora Linux, Arch Linux и других дистрибутивов.

» Блок-схема мега-популярного у умников малыша.

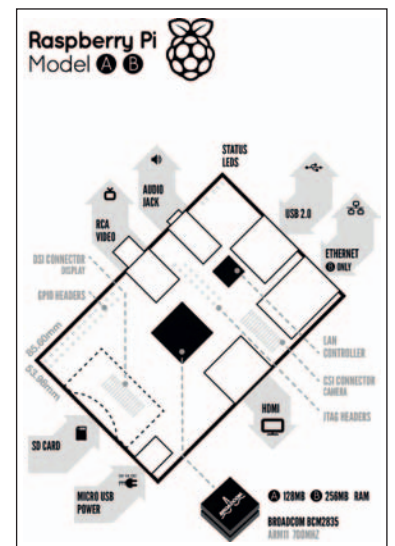


Фото © www.raspberrypi.org

# softline®

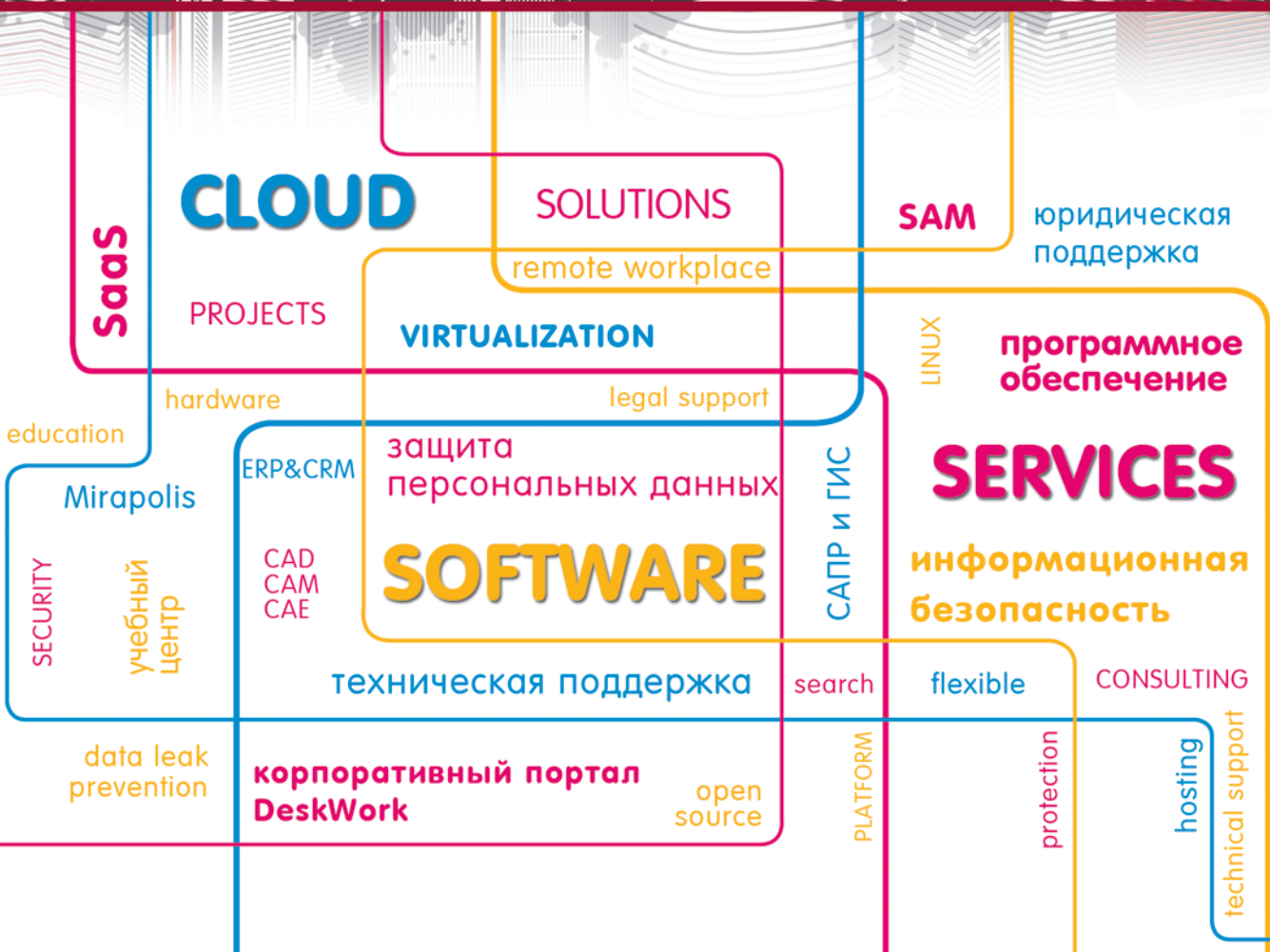


Services

Software

Cloud

## ИТ-архитектура вашего бизнеса



КРИЗИС ОКРУЖЕНИЯ

# Будущее популярной среды

Грандиозные планы по выпуску Gnome 4 и созданию Gnome OS.

На прошедшей в конце июля в Испании конференции GUADEC 2012 (Gnome Users and Developers European Conference, или The Gnome Conference – ежегодная европейская конференция пользователей и разработчиков среды рабочего стола Gnome) был представлен доклад “A bright future for Gnome [Светлое будущее для Gnome]”, авторы которого попытались не только озвучить текущие проблемы развития Gnome, но и предложить несколько идей по увеличению востребованности платформы. Озвученные в докладе предложения сводятся к усилению ориентации на мобильные системы, продолжению линии развития Gnome 3 и подготовке самодостаточной операционной системы Gnome OS.

По мнению докладчиков, фокусирование на традиционных настольных системах мешает превращению Gnome в массовую платформу. В настоящее время наблюдается бум мобильных систем, и, судя по проводимым маркетинговым исследованиям, взрослая часть населения США все больше склоняется к использованию портативных устройств – наблюдается рост продаж смартфонов, планшетов и ноутбуков, в то время как интерес

к настольным ПК неуклонно падает из года в год. При этом рынок мобильных систем еще не достиг насыщения, и на нем главенствуют платформы iOS и Android, но существует категория пользователей, которых не все устраивает в данных платформах.

Следующий значительный скачок предлагается совершить в марте 2014 года – выпуск Gnome 4.0, который можно будет рассматривать как платформу для построения собственного дистрибутива Gnome OS.

За остающиеся до выхода Gnome 4.0 полтора года планируется довести до зрелого и отточенного состояния пользовательскую оболочку и все базовые приложения Gnome. Gnome 4.0 будет обладать

**«Gnome 4.0 можно рассматривать как платформу для Gnome OS.»**

полной поддержкой управления через сенсорные экраны, что даст возможность использовать Gnome OS на планшетных ПК. Кроме непосредственно пользовательско-



Групповая фотография всех участников GUADEC.

го окружения, предлагается подготовить и такие компоненты, как инсталлятор и систему установки обновлений. Также будет сформирована сборочная среда для создания готовых дистрибутивов, пригодных для использования на конечном оборудовании.

Для разработчиков приложений в состав Gnome 4 будет введен специальный SDK, позволяющий создавать для Gnome OS и нативные классические программы, и приложения, основанные на использовании web-технологий. Для распространения созданных при помощи SDK приложений будет подготовлены средства доставки программ для пользователей, в том числе каталог-магазин приложений. Прототипом для создания подобной системы может стать уже работающий каталог дополнений для Gnome Shell.

Формирование Gnome OS также позволит решить проблемы, связанные с низкой вовлеченностью корпораций в разработку и ограниченностью ресурсов. Для привлечения финансирования и дополнительных разработчиков можно будет использовать бизнес-модели, основанные на поставке Gnome OS в качестве системы, предустановливаемой на устройства производителей-партнеров. Упоминается амбициозный план 20x20 – добиться в 2020 году 20% доли рынка, выдвинутый на смену представленному в 2005 году проекту 10x10, в рамках которого ожидалось добиться к концу 2010 года использования Gnome на 10% рабочих станций. LXF

## Болезни роста

За последние полтора года некогда самая популярная свободная среда рабочего стола изменилась до неузнаваемости. С переходом разработчиков на GTK+ третьего поколения изменились не только способы и методы написания приложений: коренным преобразованиям подвергся и интерфейс Gnome. Причем изменения эти далеко не всегда в лучшую сторону. В погоне за простотой использования, интуитивностью и унификацией интерфейса для различных классов устройств разработчики, похоже, забыли о тех, для кого они все это делают – о людях.

Третья версия Gnome стала куда большим провалом, чем приснопамятная KDE версии 4.0. Дойдя до пика популярности и удобства использования (вспомните доведенный до совершенства Gnome 2.32), Gnome скатилась в пропасть, из которой будет очень и очень непросто выбраться... Проект покинули некоторые ключевые разработчики, разочаровавшись в перспективах проекта, а фрагментация кода и сил

достигла невиданных высот (все эти Mate, Cinnamon и прочие Unity возникли как раз после выхода третьей версии многостраничной DE). Уже сейчас многие «мейнстримовые» дистрибутивы Linux отказываются от использования Gnome в качестве рабочего стола по умолчанию – Ubuntu, Debian, Fedora. И это только начало. Шутки в сторону – сам Линус Торвалдс [Linus Torvalds] отказался от Gnome в пользу более привычного Xfce.

Верится, что разработчики рано или поздно вернуться к своим корням – к простоте и легкости использования, задумаются о миллионах простых пользователей, перестанут проводить безумные эксперименты и доведут-таки лучшую среду рабочего стола во Вселенной до ума. Взгляните на досуге на экранные снимки Gnome от первой версии до версии 2.32, а затем поработайте немного в вашем Gnome 3.4, и вы поймете, почему не стоит экспериментировать с самым совершенным интерфейсом в истории человечества...

# Нордавинд Северо-Запад

## Охранные системы на базе GNU/Linux

Компания «Нордавинд» существует с 2004 года и является пионером в области разработки и производства компонентов для охранных систем на базе высоконадежной операционной системы GNU/Linux

## Наши услуги и продукты

- Проведение НИР
- Разработка и изготовление систем охранного телевидения и сопутствующих компонентов
- Проектирование сложных систем и комплексов
- Широкий ассортимент готовой продукции

## Наши системы видеонаблюдения предназначены для:

Паркингов

Зданий

Бизнес-центров

Торговых комплексов и складов

Открытых площадок

Жилых домов

Объектов повышенной секретности

Режимных предприятий

## Сертификаты ФСТЭК и ФСБ

4 канальная система видеонаблюдения стоимостью **40 тыс. руб.** —  
**в 2 раза дешевле**, чем предыдущие технологии.



### ТелеВизард

Многофункциональный сервер общего назначения.



### ТелеВизард Авто

Самая бюджетная и высокоэффективная система распознавания автомобильных номеров.



### Источники видеосигнала

Комплекс eyeSense, в состав которого входят видеочамера и объектив. Дополнительное оборудование — ИК-прожектор и климат-защита.

«Нордавинд Северо-Запад»

Научно-производственная компания

Тел. +7 (812)

**309 0686**

Тел./факс +7 (812)

**640 4990**

<http://spb.nordavind.ru>

e-mail: [spb@nordavind.ru](mailto:spb@nordavind.ru)

Приглашаем к партнерству дилеров и проектировщиков систем

В СВОБОДНОЕ ПЛАВАНИЕ

# Открыта еще одна мобильная ОС

Компания HP открыла базовые приложения мобильной платформы webOS.

**К**омпания Hewlett-Packard завершила очередной этап открытия компонентов мобильной платформы webOS, на базе которых в сентябре планируется подготовить полностью открытый релиз Open webOS.

Открыт код следующих составных частей webOS:

» Базовые пользовательские приложения платформы: почтовый клиент, календарь-планировщик, система ведения заметок, часы, калькулятор, менеджер управления распределением финансов.

» Активная ветка разработки системного менеджера [System Manager], известного под кодовым именем Luna. Системный менеджер управляет выполнением web-приложений, отвечает за организацию работы интерфейса Card View, обеспечивает запуск приложений, формирует экран блокировки системы, управляет меню и статусной строкой. В представленном коде улучшена архитектура и обновлены компоненты QtWebKit и Qt.

Следующей стадией, запланированной на август, станет формирование бета-выпуска Open webOS и публикация переработанной системы сборки. Ожидается, что с этого момента начнется активное привлечение сил сообщества к развитию webOS, которое приведет к портированию на новые аппаратные платформы и новые

типы устройств, а также к интеграции поддержки популярных открытых технологий. Для упрощения экспериментов с проектом будет сформирована специальная среда разработки, включающая эмулятор и рабочий стол.

Релиз Open webOS планируется построить на основе ядра Linux 3.3 или более поздней версии. Одновременно принято решение полностью избавиться от используемых в webOS проприетарных прошивок и закрытых драйверов. Open

## «Начнется активное привлечение сил сообщества к webOS.»

webOS изначально будет поддерживать только SoC, для которых доступны открытые драйверы.

К сожалению, изначально поставляемые с webOS устройства поддерживаться не будут, так как для них доступны только проприетарные драйверы сторонних производителей, без наличия открытых альтернатив. Те, кто заинтересован в использовании webOS для TouchPad, могут использовать ранее опубликованную сборку Community Edition, включающую проприетарные компоненты. **LXF**



## Поздний старт Hewlett-Packard

WebOS от компании Hewlett-Packard – встраиваемая открытая ОС на базе ядра Linux, разработанная компанией Palm для смартфонов и адаптированная под планшетные компьютеры. С 2010 по 2011 год поддерживалась и разрабатывалась HP, поглотившей Palm. В сентябре 2012 года HP должна опубликовать исходные коды webOS (Open webOS 1.0), и ОС перейдет в ряд открытого ПО (как и ОС Google Android). В дальнейшем HP будет поддерживать и разрабатывать webOS вместе с открытым сообществом разработчиков.

Первые webOS была представлена в Лас-Вегасе 8 января 2009 года, под именем Palm webOS. Операционная система не основана на Palm OS (основной операционной системе для устройств Palm в период 1997–2008 годов). В webOS особое внимание уделено интеграции с социальными сетями и Web 2.0, а также многозадачности.

Казалось бы, успех обеспечен: серьезная кодовая база, поддержка крупнейшего производителя ПК и мобильных устройств, прекрасная ОС, команда профессионалов, создававшая легендарные Palm'ы, открытый исходный код ядра... Однако HP опоздала с выходом на сцену как минимум на 4,5 года. В начале 2008 года еще можно было первой выйти на рынок, опередив Google (Microsoft, кстати, тоже опоздала, но тому виной скорее излишняя консервативность руководства гиганта индустрии), но тогда у HP попросту не было никакой webOS – она принадлежала умирающей Palm.

Сейчас, когда рынком мобильных ОС безраздельно правят Google и Apple, HP будет трудно занять место на площадке, учитывая тот факт, что успех двух гигантов желает разделить третий, очень грозный – Microsoft. В таких условиях webOS может и угодить на обочину шоссе дороги мобильных ОС.

## Новости короткой строкой

» Представлен порт мобильной платформы Firefox OS (Boot to Gecko) для одноплатного компьютера Raspberry Pi, построенного на базе процессора Broadcom BCM2835  
Источник: <http://mozillalinks.org>

» Компания Microsoft официально анонсировала обеспечение полноценной поддержки открытого стандарта Open Document Format (ODF) 1.2 в готовящемся к выходу MS Office 2013.  
Источник: <http://blogs.office.com>

» Проект Debian отпраздновал свое двадцатилетие. Дистрибутив был впервые анонсирован Яном Мердоком [Ian Murdoch] 16 августа 1993 года в списке рассылки [comp.os.linux.development](http://comp.os.linux.development).  
Источник: [www.debian.org](http://www.debian.org)

» По невыясненным причинам с 31 июля недоступен хостинг открытых проектов SourceForge у всех операторов связи на территории Китая. Предпринятые попытки восстановления доступности сервиса для пользователей в Китае к успеху не привели.  
Источник: [www.opennet.ru](http://www.opennet.ru)

» Организация X. Org Foundation объявила о присоединении к инициативе Open Invention Network (OIN), ставящей перед собой цель защиты экосистемы Linux от патентных претензий.  
Источник: <http://lists.x.org>

» В открытый доступ выложены относящиеся к категории свободного контента музыкальные записи, сформированные в результате работы проекта Musopen. Проект ставит своей целью поиск и запись музыки, на которую истек срок действия авторского права и которая находится в статусе общественного достояния.  
Источник: [www.opennet.ru](http://www.opennet.ru)

» В процессе подготовки установочных дисков для Debian 7.0 Wheezy разработчики проекта осуществили переход на использование по умолчанию рабочего стола Xfce. Возможность использования Gnome, KDE и LXDE оставлена в полном объеме  
Источник: [www.opennet.ru](http://www.opennet.ru)

» После четырехлетнего перерыва в разработке представлен тестовый выпуск новой версии Damn Small Linux, некогда популярного мини-дистрибутива для устаревшего оборудования. Размер ISO-образа DSL 4.11 RC1 – 50 МБ.  
Источник: [www.opennet.ru](http://www.opennet.ru)

# За год до юбилея

Кирилл Степанов делится впечатлениями о IX Конференции разработчиков СПО.

**К**ак создать хорошую традицию? Да очень просто – сделать что-нибудь хорошее, а потом повторять это регулярно. Убедительным доказательством данного утверждения стала IX Конференция разработчиков свободных программ, которая прошла 23 и 24 июля. Конференция традиционно проводится на последней неделе июля (в преддверии Дня Сисадмина). Традиционным же местом проведения служит Центральный институт повышения квалификации сотрудников МинАтома в исторически первом наукограде нашей страны – городе Обнинске. Само место волей-неволей настраивает на серьезную работу.

Мне довелось принять участие в этой конференции лет шесть или семь назад. Могу уверенно утверждать: главная традиция строго соблюдается и, по моему мнению, делает конференцию уникальной. Это единственное из известных мне мероприятий, где название полностью соответствует содержанию. Речь идет исключительно о разработке СПО. Все, ну или почти все, выступающие являются разработчиками и рассказывают о своих проектах. Любые иные, маркетинговые (о преимуществах СПО с точки зрения бизнеса) или политические (о путях развития Национальной программной платформы) вопросы традиционно оставляются для других мероприятий, и организаторы – компания AltLinux и Институт логики, когнитологии и развития личности – строго за этим следят.

Среди участников и докладчиков, опять же традиционно, были коллеги из Украины и Белоруссии. А вот спонсором в этом году впервые в истории конференции выступила... корпорация Microsoft. Нет, небо не обрушилось на землю, и река Протва не потекла вспять. Просто у «оплота мирового проприетаризма» появились на то веские причины. Но об этом чуть позже.

## «Высшая лига»

На всех восьми предыдущих конференциях большое внимание уделялось проектам, от которых зависит сам процесс разработки СПО – системам сборки и разрешения зависимостей, средствам ведения репозитория и автоматизации тестирования. Это не удивительно. Команда AltLinux долгие годы была единственным в России обладателем собственного репозитория Sisyphus и собственной системы сборки. На нынешней конференции, как и в прошлые годы, можно было услышать очень интересные доклады майнтейнеров Sisyphus и разработчиков AltLinux Игоря Влащенко, Виталия Кузнецова, Дмитрия Левина, Михаила Пожидаева и Михаила Шигорина, посвященные средствам автоматизации разработки и сопровождения программных пакетов. Говорилось

о перспективах развития дистрибутива – Седьмой платформе, работах по портированию на архитектуру ARM. «В кулуарах» можно было увидеть сборку для планшетов на Intel Atom с поддержкой мультитач

и датчиков ориентации, но отдельного доклада об этой работе, к сожалению, не прозвучало.

А вот система сборки ABF (Automatic Build Farm) компании РОСА (ЗАО «РОСА») была представлена на конференции впервые. Основанная в 2011 году ROSA Lab. стала второй в нашей стране компанией, обладающей «полным циклом» создания собственного дистрибутива, а это дорогого стоит! Как сказал заместитель генерального директора AltLinux Алексей Новодворский, именно собственная система сборки служит пропуском в «высшую лигу» создателей СПО. Кстати, первый результат их работы – дистрибутив ROSA Marathon 2012 – вы можете загрузить с DVD, приложенного к этому журналу.

Сотрудники компании РОСА Евгений Буданов, Роман Вялов, Владимир Рубанов и Денис Силаков рассказали о самой ABF

»

## «Почти все выступающие рассказывают о своих проектах.»



» Алексей Новодворский в девятый раз открывает конференцию.





➤ Общий взгляд на «облака». Справа — Алексей Боков, Microsoft.

и ряде перспективных технологий, таких как RPM5. Своим опытом автоматизированной сборки *Wine* под разные дистрибутивы (универсальная сборочная система *Kornif* и единая команда управления пакетами *ERM*) поделились коллеги из питерской компании Etersoft Виталий Липатов и Даниил Михайлов. У разработчиков популярных пакетов и приложений разнообразие дистрибутивов давно вызывает головную боль, но средства от нее есть и активно разрабатываются.

## Кто смотрит на облака

А теперь о главной интриге этого года: впервые в истории конференции в ней участвовал Microsoft. Не странно ли, что ведущий изготовитель проприетарного софта даже спонсирует конференцию разработчиков СПО. Разве могут у Microsoft быть общие интересы и даже совместные проекты, например, с AltLinux? Оказывается — да, а проекты — вполне успешные.

Так вот, Алексей Боков из московского представительства Microsoft рассказал о возможностях запуска Linux на «облачной» платформе Azure и совместной с командой AltLinux работе по адаптации дистрибутива для работы в этой среде. Собственно, адаптировать пришлось агент управления гостевой операционной системой, без которого работа ее в облаке Azure невозможна. Там даже консоль виртуальной машины посмотреть нельзя — все только через агент. Впрочем, оно и понятно — Azure предназначен в первую очередь для запуска виртуальных серверов и служб на них.

## «Бывший противник вполне может стать хорошим союзником.»

Поначалу агент работать не хотел. Но как только программисты AltLinux установили прямые контакты с инженерами Microsoft (что потребовало некоторых усилий по преодолению корпоративной бюрократии), все проблемы удалось решить достаточно быстро. После обычной в таких случаях «доработки напильником» код заработал, и теперь AltLinux сертифицирован Microsoft для работы в среде Azure — кстати, первым из отечественных дистрибутивов.

Тут может возникнуть резонный вопрос: а зачем, собственно, компании Microsoft поддерживать и продвигать Linux в своих «облаках»? Ведь все прекрасно помнят совсем другие выступления представителей этой фирмы, где убедительно доказывалось, что в корпоративной среде совокупная стоимость владения (TCO) серверами под Windows ниже, чем аналогичными под Linux, а сама идея СПО вообще губит индустрию программирования, «убивая рабочие места».

Судя по всему, за последние год-полтора в Редмонде произошла революция. Как рассказали коллеги, побывавшие в июне на Всемирной партнерской конференции Microsoft в Торонто, там не было ни одного выступления, где речь шла о конкуренции между Windows и Linux в качестве серверных платформ. Зато почти в каждом докладе подчеркивалось, что облачные платформы и сервисы — это наше «светлое будущее», а главный конкурент скорее *VMware*, чем RedHat или SUSE.

Дальнейшая логика известна со времен Сун-Цзы — бывший противник вполне может стать хорошим союзником. Поэтому в московском и других представительствах Microsoft появилась должность ответственного за поддержку Linux.

Что ж, пусть в облаках Azure «расцветают все цветы». Если заказчик хочет Linux, мы не препятствуем — и даже поможем. Тенденция, однако...

## Не только инфраструктура

Темы докладов на конференции не ограничивались инфраструктурой для разработки и операционными системами. Обсуждались и проекты вполне прикладные. Так, Александр Боровский рассказал о проекте Linux Vacation / Eastern Europe (LVEE engine) для организации сайтов различных конференций. Код написан на Ruby on Rails и позволяет выполнить все необходимые для организаторов конференции действия: регистрацию участников, работу с тезисами, публикацию статей и многое другое.

Не были забыты свободные решения для бизнеса. Андрей Михеев из Консалтинговой группы РУНА рассказал об опыте разработки и внедрения свободной системы управления бизнес-процессами RunaWFE, а представители казанской компании «Центр» Радик Юсупов и Ленар Шакиров — о своих решениях на базе СПО, в частности Zarafa Collaboration Platform, для предприятий и органов государственной власти.

Традиционно говорили об СПО в образовании. Игорь Воронин из ИЛИП РАН рассказал об образовательном проекте УМКИ, позволяющем строить сенсорные сети из групп роботов. При демонстрации не обошлось без проявлений «адмиральского эффекта», но через некоторое время робот поехал куда сказали. Геннадий Кушнир рассказал об электронном классном журнале РУЖЭЛЬ и проблемах ведения электронного документооборота в школах Москвы.

Были и другие интересные доклады. Тезисы опубликованы на сайте AltLinux (<http://www.altlinux.ru/media/protva-2012.pdf>).

В общем, конференция явно удалась, что тоже уже стало традицией. Будем ждать следующего лета и встречи со старыми друзьями. **LXF**







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

## Бич свободных лицензий

**К**ак это ни парадоксально, но один из главных тормозов прогресса свободного программного обеспечения – это свободные лицензии. И причин тому несколько.

Во-первых, лицензий этих множество, и число их всё растёт. И хотя все они свободны, некоторые свободнее других. Впрочем, в последнем вопросе существуют кардинальные расхождения взглядов: приверженцы буквально каждой из свободных лицензий полагают, что исключительно их лицензия воплощает собой идеал свободы.

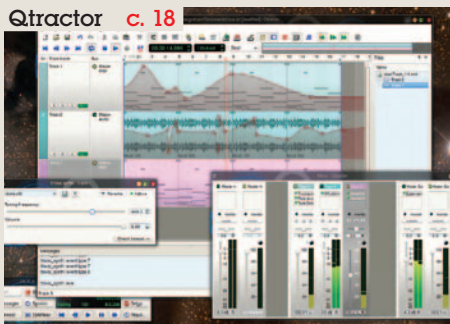
И оно бы и ладно – меряться процентным содержанием свободы ничуть не хуже, чем погонными метрами иного органа. Однако из этого вытекает «во-вторых»: все свободные лицензии в той или иной мере несовместимы друг с другом, достигая абсолютной совместимости только сами с собой. То есть код под разными лицензиями в едином проекте может использоваться либо с теми или иными ограничениями, либо не может вообще.

Примером последнего случая является судьба ZFS on Linux, описанная в колонках **LXF133** и **LXF160**. Тут, правда, инженерная мысль в конечном итоге одержала победу над крючкотворством лицензиатов. Но это потребовало дополнительных усилий разработчиков и до сих пор продолжает доставлять неудобство пользователям.

Хотя, казалось бы, именно авторы свободных лицензий декларируют, что «всё на благо пользователя, всё для счастья пользователя».  
[alv@posix.ru](mailto:alv@posix.ru)

## Сегодня мы рассматриваем:

- Zorin OS** ..... 16  
Мы аплодируем усилиям всех желающих довести преимущества свободной от вирусов, гибкой и бесплатной ОС Linux до масс. Когда их доблесть дополнена инструментом смены внешнего вида рабочего стола с Windows XP на OS X или Gnome и обратно, мы в дамках. Или нет?
- Fit PC 3 Pro** ..... 17  
Практически бесшумный низковольтный ПК работает под управлением Linux Mint и накаляется достаточно для поджарки фунтового бифштекса – нам он кажется отличным, тем более что потребляет всего-навсего 24 Вт углеродного сока.
- Qtractor** ..... 18  
Полный аудио/MIDI-секвенсор, названный по имени машины для фермеров? Конечно, это *Qtractor*, последняя жемчужина от человека, который принес нам *Qjackctl*, *Qsynth* и *Qsampler*.
- MK802** ..... 19  
Микроминиатюризация идет стремительными темпами: мы едва успели восхититься малюткой Raspberry Pi, а ему на смену уже спешит еще более компактное устройство. Оно умещается на ладони, но годится «и в пир, и в мир, и в добрые люди».
- Mate** ..... 20  
Пользователи, разгневанные Gnome 3, вдохновили ответвление Gnome 2 под названием Mate. Про название возможны варианты: то ли это тонизирующий индейский напиток, то ли попросту «дружбан» (если рассматривать его с англоязычных позиций).



**Qtractor** с. 18

➤ Можно добавлять/удалять дорожки, эффекты, клипы, аудио прямо в матричном редакторе.

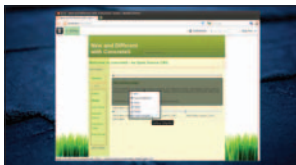


**MK802** с. 19

➤ Зеленый гуманоид привольно себя чувствует на этом крошечном гаджете.

## Сравнение: Управление контентом

### Concrete5



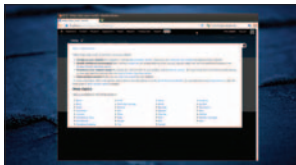
### Exponent CMS



### Cotonti



### Drupal



### Joomla



Если вам нужно увязать содержание различных типов от разных источников в один гармоничный сайт, потребуется система управления контентом – и вот вам пять лучших.

# Zorin OS

Станет ли Zorin долгожданными воротами в мир Linux для бесстрашных пользователей Windows? Бен Эверард надежды не теряет.

## Вкратце

» Коммерческий дистрибутив Linux, ориентированный на пользователей других систем. См. также: Rosa, Ubuntu, Mageia.

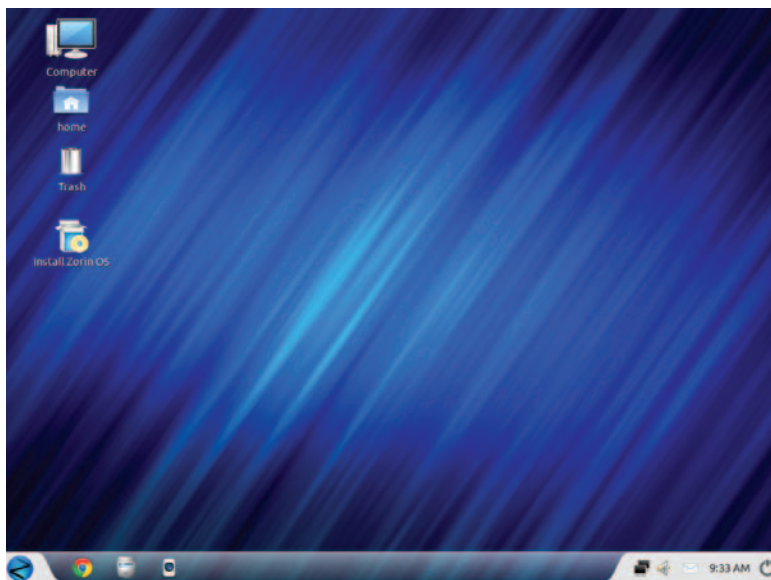
**L**inux для любителей Windows: эта концепция родилась не вчера, но на ней пока что не поставили крест. Linux/Linspire и иже с ними – уже давнее прошлое Linux, однако это еще не значит, что сама идея нежизнеспособна. Возможно, в этом преуспеет Zorin OS – о ее предыдущем релизе, ориентированном на сферу бизнеса, мы писали в **LXF148**.

В основе настольной версии – безопасный режим Gnome 3 с добавленным доком AWN, более гибким, чем в Gnome, и разработчики выбрали именно его, чтобы пользователи Windows чувствовали себя дома.

По умолчанию рабочий стол похож на Windows 7, но при помощи менеджера Zorin Look Changer вы можете переключить его на режим XP или Gnome 2. Это должно позволить вам менять различные стили, не закрывая окон, в которых вы работаете. Однако мы обнаружили, что приложение регулярно застревает в одном из режимов, и единственным способом его переключить становится перезапуск.

## Сравните предложения

Какой бы стиль вы ни выбрали, все будет очень аккуратно и доступно, хотя иногда и медленно на не новых машинах. В последние месяцы наблюдается активный рост числа окружений в стиле традиционных рабочих столов. Что побуждает сравнить их с теми, у которых, напротив, новый интерфейс: самые популярные – Cinnamon и Mate. На их фоне рабочий стол Zorin выглядит немного примитивным – возможно, из-за ориентации на нетехническую аудиторию – но также и незавершенным. Мы обнаружили визуальные артефакты,



» Большинство дистрибутивов Linux соперничают за пользователей друг с другом, а Zorin приводит пользователей извне.

которые, не влияя на функциональность, создавали впечатление неряшливости. Будем надеяться, что в последующих версиях все это подчистят.

Связь с Windows в Zorin – не только пользователи: разработчики постарались сделать так, чтобы и программы тоже сокращали разрыв с Microsoft. Потому в составе стандартного рабочего стола есть *Wine*, *Wine Tricks* и *Play On Linux*. Они выполняют важную функцию, позволяя запускать ПО от Windows – но только если уметь ими пользоваться. В том-то и главная проблема Zorin OS: несомненно, разработчики вложили много усилий в это ПО, и вышла вполне приятная в использовании система, но по ней практически нет документации. Раздел Справка в главном меню отправляет к Gnome 3, что наверняка только запутает пользователей, поскольку их внешний облик уж очень различен. Логично предположить, что пользователи Debian или Fedora прекрасно знают, как запускать приложения Windows через *Wine*, или что Ubuntu One – это облачное хранилище, но пользователи Windows вряд ли настолько эрудированны.

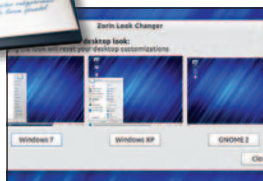
Естественно, охотников рыться в толстых справочных талмудах мало. Но при переходе к новой системе иметь доступный источник информации о ней просто необходимо. Особенно это касается *Wine*

и его расширений, поскольку настроить через них программу может быть делом довольно кропотливым.

Zorin OS – не только проект сообщества, но и коммерческий продукт. Помимо бесплатной Стандартной версии, есть версия Премиум, доступная для скачивания за €10 или на диске, за €18. В общем и целом, эта настольная система не уступает детищу Редмонда. Хотя им определенно есть чем еще порадовать пользователей. Было бы неплохо, чтобы появилась Справка, а еще лучше – ноутбуки с предустановленной системой. **LXF**



## Свойства навскидку



### Zorin Look Changer

Позволит вам работать с привычным рабочим столом, будь то Windows XP, Windows 7 или Gnome 2.



### Панель AWN

Аккуратна и проста в использовании, но не без странностей отображения.

## LINUX FORMAT Вердикт

### Zorin OS

Разработчик: Zorin Group

Сайт: <http://zorin-os.com>

Цена: Стандартная версия – бесплатно, Премиум-версия – €10 за скачивание, €15 за диск

Функциональность 8/10

Производительность 7/10

Удобство использования 8/10

Документация 5/10

» По ощущениям – как Windows, но без всяких лицензий. Подводит отсутствие документации.

Рейтинг 8/10

# Fit PC 3 Pro

Грэм Моррисон решил взглянуть на ПК с характеристиками прославленного MintBox и пассивным охлаждением.

## Вкратце

» Почти бесшумный Linux ПК с низким энергопотреблением и предустановленным Mint 12. См. также: Linutop или DIY.

**П**ервое, что ошарашит вас, когда вы возьмете в руки этот миниатюрный ПК – его вес. Со своими 1,3 кг он сгодится в качестве пресс-папье даже самому заядлому графоману. Дизайн тоже очень необычный, благодаря крепкому литому алюминиевому корпусу с ребрами радиатора для вентиляции. При помощи этого ПК можно не только управиться с бумагами, но и пожарить вафли/стейк/халу-ми (судя по нагреву, это так). Начинка этой коробочки может варьироваться; в основе нашей расширенной модели Pro – платформа AMD Inagua и двухядерный AMD G-T56N, встроенная видеокарта Radeon HD 6320, 4 ГБ ОЗУ и 250 ГБ дисковой памяти. Сзади расположены порты USB 2 и USB 3, аудиовыход, HDMI и DisplayPort для подключения видео и два разъема для интегрированного беспроводного адаптера. Как ни крути, все по последней моде.

## Прием без исключений

ПК укомплектован встроенной 64-битной версией Linux Mint, что делает его практически идентичным MintBox. Изначально стоит версия 12, и мы были рады увидеть проприетарные драйверы, особенно после тестирования интегрированных

## «Тестируя графику, мы сыграли пару раундов в Alien Arena.»

графических чипов AMD для темы этого номера. Беспроводное соединение тоже не вызвало нареканий: не знаем, заслуга ли это удачной формы корпуса или пары антенн, прикрепленных по углам задней стенки, но уровень приема сигнала был исключительным, позволяя нам ловить Wi-Fi из кафе Starbucks за 3 улицы (это шутка! Красть данные, дети, очень стыдно). Немного поиграв с конфигурацией, мы начали проводить тесты производительности, и хоть мы не ожидали чудес от этого не самого мощного/энергоэффективного устройства, пользователи настольных систем результаты должны весьма порадовать. Наиболее точными показались результаты выполнения John the Ripper (из комплекта тестов Phoronix



» В Pro версии Fit PC 3 есть пластины для вентиляции, в то время как Стандартная версия имеет гладкий корпус.

Test Suite). Согласно им, CPU системы находится на уровне полноценного настольного ПК образца 2009 года – ближайшим к PC 3 в рейтинге оказался Opteron 8220. Что уже очень неплохо, с учетом того, что это практически бесшумное устройство, и без учета графической карты Radeon HD и скорости памяти DDR3.

Тестируя графику, мы сыграли пару раундов в *Alien Arena*, и все прошло как по маслу. Затем мы попробовали посмотреть видео в высоком качестве через BBC iPlayer на базе Flash. Это тоже не составило никаких проблем, что уже превосходит PlayStation 3. Оффлайн-контент 1080p тоже прекрасно воспроизводился через VLC. Наконец, мы попытались запустить графический тест *Heaven 3*, но не особо удивились, обнаружив, что ряда новых функций, необходимых для демонстрации, в нашей скромной встроенной видеокарте не нашлось. Но на скорости 5 кадров в секунду все работало без графических ошибок, что опять же не сравнить с более старым оборудованием, драйверами с открытым кодом или видеокартами Intel; а когда мы ухудшили качество до низкого, производительность выросла вдвое.

Когда загрузка процессора составила 100 %, то в нашем офисе, который правильнее назвать парилкой, корпус очень сильно нагрелся – настолько, что к нему было невозможно прикоснуться. После того, как он пропахал всю ночь, внутренние датчики зафиксировали температуру в 88°C, что явно немало,

но в пределах спецификации. Пожалуй, больше нас обеспокоило то, что температура жесткого диска составила 54°. Это тоже в пределах спецификации, но вот на счет долговременной эксплуатации машины при такой температуре все же есть сомнения, особенно если использовать ее как сервер.

К счастью, есть вариант приобрести такую систему без дисковой памяти, и на нее можно установить SSD, что кажется более надежным. Даже при высокой нагрузке системы энергопотребление остается очень низким – всего 24 Вт, что превращает эту коробочку в идеальную машину для офиса. Вещица недорогая, но ощущается как продукт высокого качества, который долго прослужит. Чего не скажешь о многих ей подобных. **LXF**

## LINUX FORMAT Вердикт

### Fit PC3 Pro

Разработчик: Tiny Green PC  
Сайт: [www.tinygreenpc.com](http://www.tinygreenpc.com)  
Цена: £460 без НДС

Функциональность	8/10
Производительность	7/10
Удобство использования	9/10
Оправданность цены	6/10

» Вещь недорогая, но непременно окупится в деловой смете.

**Рейтинг 8/10**

# Qtractor 0.5.5

Грэм Моррисон нашел себе очередное музыкальное приложение, замаскированное под образец сельхозтехники.

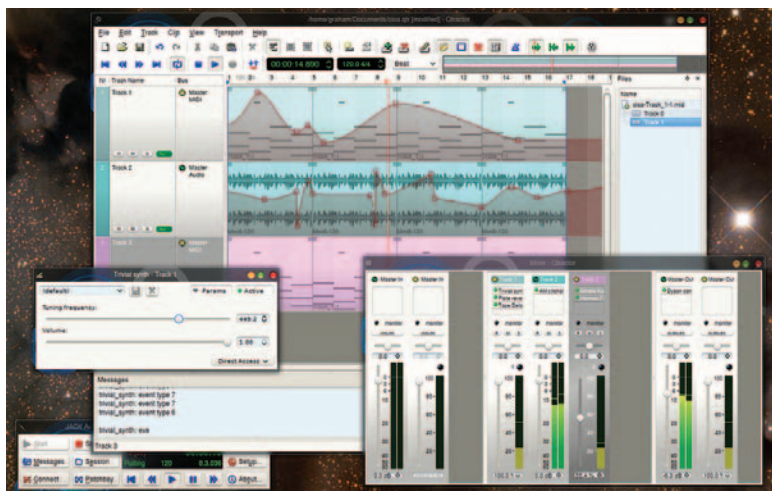
## Вкратце

» Полнофункциональная среда для создания аудио. См. также: *Rosegarden* и *Renoise*.

**П**од Linux не так уж много аудио-сенсоров, но те, что есть – достаточно функциональные и мощные (см. наш обзор *Rosegarden* в прошлом номере). Но все же есть ниша для приложения с лучшим графическим представлением, по крайней мере, пока не выйдет *Ardour 3* с полной поддержкой MIDI. И эту нишу мог бы занять *Qtractor*. Возможно, потому, что это детище Руя Нуно Капела [Rui Nuno Capela] – разработчика многих других аудиоредакторов под Linux, таких как *Qjacktl*, *Qsynch* и *Qsampler*. Общий префикс намекает, что здесь есть некая связь с инструментарием *Qt*. И *Qtractor* – не исключение. В его основе – *Qt 4*, и, как и *Rosegarden* и *Ardour*, он тоже требует рабочей конфигурации Jack. А Jack – штука хитрая, но помаленьку становится проще; и что может быть лучше, чем поработать с ним на приложении от того же человека, что создал и интерфейс Jack?

## Поддержка VST

Сразу после запуска вам дано немного. Свой шедевр придется создавать с нуля, включая добавление и MIDI-, и аудиодорожек, коли они вам нужны. Мы считаем, что этот путь гораздо естественнее, чем применяемый в *Rosegarden*, где вы вынуждены придерживаться стандартной раскладки независимо от своих потребностей. Подготовив дорожки, создавайте клип, чтобы собрать ваш аудио- или MIDI-контент. Но давать ему имя и указывать путь сохранения вовсе не обязательно, если вы просто решили черкнуть пару нот. MIDI-треки могут также содержать вирту-



» Невзирая на капризы и муторный процесс сведения, мы полюбили *Qtractor*.

альные инструменты и эффекты, и если у вас *SDK* от Steinberg, можно создать (или установить) для этого приложение с поддержкой VST. Тогда у вас появится доступ к некоторым инструментам Linux, использующим этот формат, включая *Pianoteq* и *Loomer*, о которых мы говорили пару лет назад. Но даже при отсутствии VST в вашем распоряжении форматы LADSPA, DSSI и LV2 (наследник LADSPA).

А вот если нужно создать музыку для внешнего MIDI-синтезатора или плагина, то матричное редактирование здесь оказалось делом очень нервным. Создание новых нот – испытание для терпения: придется вдосталь пощелкать по мелодии, приводя в порядок продолжительность и области выделения. Хорошо, что есть функция копирования и вставки, позволяющая указать область для вставки фрагмента через предпросмотр. При копировании и вставке клипов представление также дублируется, хотя лучше бы это делалось уже после вставки. Замечательной нам показалась функция автоматизированной настройки эффектов и инструментов, хотя разобраться с ней непросто, особенно если имеешь дело с плагинами, имеющими множество параметров. Можно переключаться между логарифмическим, ступенчатым и линейным режимом, а отображение действий в реальном времени приходится очень кстати, что бы вы ни редактировали. Проблемы возникают только в связи с тем, что после пере-

ключения иногда неясно, какие параметры сейчас активны, а курсор автоматического режима абсолютно такой же, как обычный. Пока мы боролись с графическим интерфейсом, обнаружилось также и то, что окно микшера исчезает при изменении фокуса главного окна, что особо мешает при редактировании параметров плагинов.

Но лучшую функцию мы приберегли напоследок. В отличие от всех прочих подобных приложений Linux, практически вся редакция может происходить прямо в режиме Воспроизведения, а значит, можно добавлять/удалять дорожки, эффекты, клипы и аудио, а также редактировать музыку прямо в матричном редакторе. И разве это не то, что нужно? **LXF**



## Свойства новскидку



### Документация — ОК!

*Qtractor* – один из лучших образцов документации среди всех открытых проектов.



### Прямое управление

Выберите нужный параметр и используйте микшер, чтобы изменить его значение.

## LINUX FORMAT Вердикт

### Qtractor 0.5.5

Разработчик: Rui Nuno Capela  
Сайт: <http://qtractor.sourceforge.net>  
Цена: Бесплатно под GPL

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

» Есть проблемы со стабильностью, но радует гибкость интерфейса и большой набор функций.

**Рейтинг 9/10**

# МК802

Юрий Татарский в восхищении взирает на новое чудо, сотворенное китайской промышленностью.

## Вкратце

» Полноценный микрокомпьютер размером не больше обычной флэшки. См. также: Raspberry Pi, Cotton Candy, Fit PC 3 Pro.

**У** Raspberry Pi и Cotton Candy появился крутой конкурент. Встречайте: МК802 собственной персоной.

В основе МК802 лежит AllWinner A10. Его позиционируют как двоядерный, хотя в нем всего лишь одно ядро ARM Cortex-A8 с тактовой частотой 1–1,2 ГГц и графического ядро Mali 400 MP на 500 МГц. Но этого «всего лишь» вполне хватает, чтобы декодировать HD-видео (вплоть до Super HD 2160p/3D Film), поддерживать все популярные видеокодеки, включая VP8AVS, H.264 MVC, VC-1, MPEG-1/2/4, а также кодировать HD-видео (H.264 High Profile) с максимальным разрешением 1920×1080@30fps.

Как и положено миниатюрному гаджету, в списке его интерфейсов – все, что надо, и ничего лишнего: порт мини-USB (он служит и как вход блока питания), USB-порт, слот для карты microSD и выход HDMI для подключения к монитору или бытовому ТВ. Внешний блок питания на 2 А позволит подключить внешний диск без своего источника энергии. В поставку входят шнуры miniUSB и miniHDMI, переходник miniUSB–USB и компактный блок питания. А стоимость всего комплекта – около \$70; гуманно! МК802 выпускается в двух версиях: с 512 МБ или 1 ГБ ОЗУ DDR3.

Полный титул устройства – Android 4.0 mini PC. Очевидно, что управляется оно ОС Android, которая записана на внутренний NAND-флэш объемом 4 ГБ, где под ОС отдано только 1 ГБ. Остальные 3 ГБ доступны для установки программ и сохранения настроек и файлов пользователей. ОС – «чистый» Android 4.0.4, без доработок интерфейса, хотя в русской локализации встречаются забавные переводы системных сообщений. Android нацелен на управление с сенсорного экрана, и удобнее всего будет клавиатура с тачпадом для обработки жестов мультитач, требуемых рядом игр и программ под Android.

Работа с МК802 в Android 4.0.4 Ice Cream Sandwich оставила приятнейшее впечатление. HD-видео с внешнего диска идет плавно, без рывков. Нареканий не вызвали и просмотр видео с фото, как и прослушивание музыки с домашнего медиа-сервера по DLNA. Работа с почтой и web-серфинг на ТВ-экране также весьма удобны. Отлично работает IPTV и приложения интернет-кинотеатров. А недавно выпущен XBMC для устройств на ARM-процессорах



» Крошка Android 4.0 mini PC умеет работать с образом Linux, записанным на SD-карту.

под Android. Итак, мы фактически получаем домашний кинотеатр на ладони: современный ТВ со входом HDMI благодаря МК802 станет устройством, мощнее и богаче функциями любого SMART TV.

Эти чудеса были вполне ожидаемы, с учетом избранной платформы. Зато приятный сюрприз – возможность запуска образа Linux с внешней карты памяти.

С сайта производителя можно скачать готовые образы оптимизированных под МК802 модификаций Ubuntu (Lubuntu, Xubuntu и Ubuntu с Unity), в зависимости от объема ОЗУ и разрешения монитора. Самым функциональным оказался вариант с Lubuntu. Тестирование, правда, не выявило сверхпроизводительности – вероятно, из-за отсутствия драйвера Mali 400 в данном образе системы.

Какие же плюсы ждут нас с Ubuntu? Во-первых, это офисный пакет, полноценный браузер и почтовые приложения – можно организовать настоящее рабочее место, с обычной мышью и клавиатурой. Во-вторых, mini-PC в такой конфигурации отлично справляется с серверными задачами, при наличии *transmission-daemon* (торрент-клиент), *minidlna* (медиа-сервер), SSH- и FTP-сервера, web-сервера, Asterisk (сервер IP-телефонии).

Готовых образов серверов на сайте производителя нет, но вследствие большого интереса к этому гаджету уже появились таковые на базе Debian, что, безусловно, повысит производительность.

На наших глазах народился новый класс устройств – и, судя по количеству

посвященных ему обсуждений и ресурсов в сети, уже обрел немало приверженцев. Ожидаются новые версии и модификации устройств класса и стоимости МК802 для решения различных задач: к примеру, мобильная версия с автономным питанием или проводными сетевыми интерфейсами. Вариантов может быть много.

Ничтожное энергопотребление, крошечный размер и чисто условная стоимость обеспечат МК802 широкое применение как минимум в трех областях: на Android этот прекрасный домашний медиа-центр с лихвой перекрывает возможности любого Smart TV; на Ubuntu легко работает с документами, Интернетом и почтой; а его достоинства как сервера неоспоримы. Видеоиллюстрация на LXF DVD демонстрирует возможности МК802. **LXF**

LINUX Вердикт  
FORMAT

**MK802 (Android 4.0 mini-PC)**

Разработчик: Miniand Tech  
 Сайт: www.miniand.com  
 Цена: \$70

Функциональность	9/10
Производительность	8/10
Удобство использования	9/10
Оправданность цены	10/10

» Супер-миниатюрный полнофункциональный компьютер сгодится и как медиа-центр, и как домашний ПК, и как сервер.

Рейтинг 9/10

# Mate 1.4

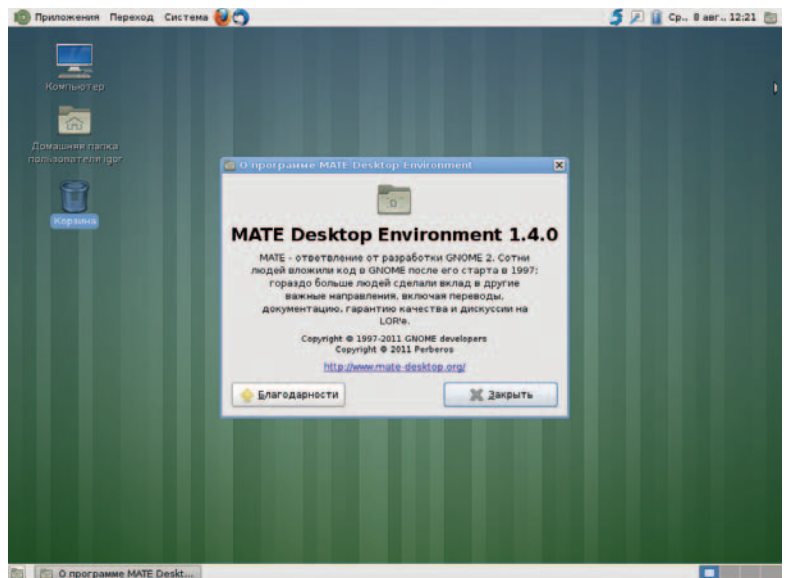
Игорь Штомпель заинтересовался ответвлением Gnome, инициированным сообществом как противовес политике разработчиков.

## Вкратце

» Mate — от- ветвление рабо- чего окружения Gnome 2. См. так- же: Gnome 2, Cin- namon, Xfce.

**В**ыход Gnome 3 вызвал много крити- ческих откликов. Справедливости ради надо сказать, что были и по- ложительные, но тем не менее немало количество пользователей стало искать альтернативу. Среди них были и известные разработчики — например, создатель Linux Линус Торвальдс, который призвал создать ответвление Gnome 2.

В итоге таких призывов появилось на свет ответвление Gnome 2, а его осно- вателем стал пользователь дистрибу- тива Arch Linux (<http://mate-desktop.org/2011/12/05/hello-world/>). Новый про- ект был анонсирован 5 декабря 2011 году под именем Mate (<http://mate-desktop.org>), происходящим от названия напитка mate, который готовят из побегов дерева падуба. За время существования состоялось два релиза — 1.2 и 1.4. Последний появи- лась на свет 30 июля 2012 года. С офици- альным анонсом версии 1.4 можно озна-



» Рабочее окружением Mate 1.4.

## «Немалое количество пользователей стало искать альтернативу.»

комиться по адресу <http://mate-desktop.org/2012/07/30/mate-1-4-released>.

Кстати говоря, развитие Gnome 3 продолжает вызывать недовольство сообществом. Вот, например, совсем недавно

разработчики проекта Linux Mint представили ответвление файлового менеджера *Nautilus*, который получил название *Nemo* (он станет составной частью сборки дистрибутива на базе рабочего окружения *Cinnamon*). А почему им пришлось это делать? Да потому, что разработчики Gnome в экспериментальной версии *Nautilus 3.5.x* убрали часть функциональных возможностей — по их мнению, для упрощения работы с программой (например, были

удалены двухпанельный режим, боковая панель и т. д.).

Рабочее окружение Mate продолжает успешно развиваться. Рассмотрим, в чем оно стало лучше.

## Первое знакомство

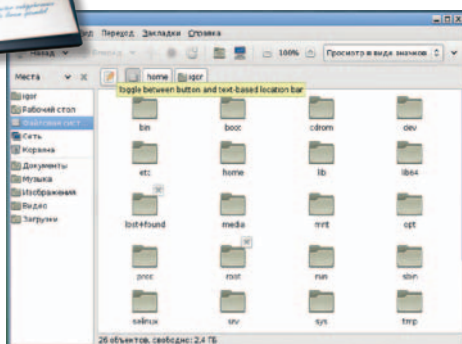
Релиз 1.2 был основан на кодовой базе Gnome 2.32. Основным отличием Mate 1.2 от Gnome 2.32 стало отсутствие пересечений с Gnome 3, а значит, ничто не мешает установить Mate наряду с Gnome 3. А если вы хотели бы установить совместно Gnome 2 и Gnome 3, то это не увенчалось бы успехом, так как возникал конфликт из-за ряда имен файлов. Выход Mate 1.2 решил эту проблему. Начало развития проекта можно считать удачным: его сразу же подхватил один из самых популярных дистрибутивов Linux, проект Linux Mint, и с версии 12 выпускает сборку своей ОС с рабочим окружением Mate.

## Установка Mate 1.4

Пакеты с Mate 1.4 доступны для ряда дистрибутивов — среди них Debian, Ubuntu, Linux Mint, Fedora, openSUSE, Arch Linux, Slackware.

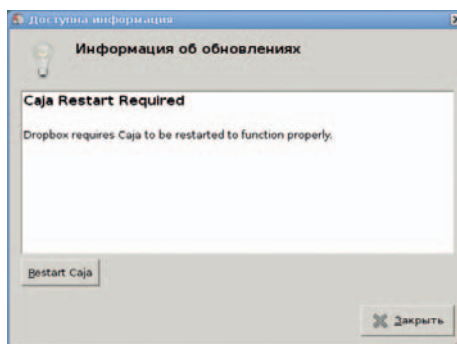
Для установки Mate 1.4 в Ubuntu 12.04 (которую мы использовали) необходимо, как указано на странице официального сайта проекта Mate, посвященной установ-

## Свойства навскидку



### Переключение навигации

Используйте специальную кнопку, чтобы указать путь вручную в текстовой строке.



### Интеграция с Dropbox

Файловый менеджер *Caja* получил возможность интеграции с сервисом Dropbox.

ке рабочей среды в Ubuntu (<http://mate-desktop.org/install/#ubuntu>), подключить следующий репозиторий:

```
sudo add-apt-repository "deb http://
packages.mate-desktop.org/repo/ubuntu
precise main"
```

Далее следуем инструкции с указанной выше страницы официального сайта. Вначале обновим менеджер пакетов:

```
$ sudo apt-get update
```

Затем загрузим ключ:

```
sudo apt-get install mate-archive-keyring
```

Опять обновим менеджер пакетов:

```
sudo apt-get update
```

Установим базовые пакеты –

```
sudo apt-get install mate-core
```

и остальные пакеты, необходимые для работы Mate:

```
sudo apt-get install
mate-desktop-environment
```

Далее завершаем сеанс работы, выбираем необходимого пользователя и рабочее окружение Mate. После чего, войдя в систему, вы увидите нечто подобное показанному на рисунке (мы дополнительно открыли окно с информацией о Mate).

## Новшества и особенности Mate 1.4

Многие приложения Gnome 2 получили в Mate новые имена (более подробно можно ознакомиться с ними, например, здесь: <http://wiki.mate-desktop.org/applications>). Так, файловый менеджер *Nautilus* получил имя *Caja*, а текстовый редактор *Gedit* стал называться *Pluma*. Если в Gnome для быстрого открытия текстового файла можно было дать команду

```
$ gedit имя_файла
```

то в Mate это будет выглядеть следующим образом:

```
$ pluma имя_файла
```

Среди других приложений, получивших новые имена: программное обеспечение для просмотра различных документов *Evince* получило имя *Atril*; эмулятор терминалов *Gnome Terminal* стал *Mate Terminal*; менеджер архивов *File Roller – Engrampa*; программа для просмотра изображений *Eye of Gnome (EOG) – Eye of Mate (EOM)*; калькулятор *gcalctool – MateCalc*; оконный менеджер *Metacity – Marco*; конфигурационная система *GConf – MateConf*; дисплейный менеджер *GDM – MDM*; редактор меню *Alacarte – Mozo*; приложение для изображения диалоговых окон из командной строки *Zenity – MateDialog*.

Конечно, переименование такого количества знакомых приложений способно вызвать досаду. Но зато пользователи Mate могут установить *Pluma* и *Gedit* одновременно, как и *Engrampa* с *File Roller*, не опасаясь конфликтов между приложениями. Таким образом обеспечивается совместимость Mate и Gnome, давая воз-

можность параллельно использовать приложения обоих рабочих сред.

Файловый менеджер *Caja* получил поддержку работы с облачным хранилищем данных *Dropbox*, для чего необходимо дополнительно установить пакет *caja-dropbox*. Но учтите, что для использования *Dropbox* вам не обойтись и без *deb*-пакета *dropbox*, который необходимо установить до пакета *caja-dropbox*, скачав его с сайта <https://www.dropbox.com/install>. После установки пакета *dropbox* вам предложат докачать все необходимое для приложения *Dropbox*, а затем в течение нескольких шагов вы будете либо зарегистрированы на сервисе, либо задействуете свою учетную запись. В итоге всех этих действий в вашем домашней папке появится папка *Dropbox*, содержимое которой

## «Многие приложения Gnome 2 получили в Mate другие имена.»

будет синхронизировано с вашим содержимым на сервисе. Плагин *caja-dropbox* работает по тому же принципу, что и плагин *nautilus-dropbox*.

Разработчики улучшили и другие аспекты файлового менеджера *Caja* – например, работу с закладками. В *Places* [Места] (боковая панель) их теперь можно открывать при помощи клавиши пробела и *Enter*. Еще одно функциональное улучшение – реализация поддержки обмена файлами по протоколу *Bluetooth*. Добавлена поддержка, при включенном композитном режиме оконного менеджера *Marco*, переключения между окнами с использованием сочетания клавиш *Alt+Tab*. А поддержка переключения пользователей через *GDM* появилась в *mate-screensaver*. Кроме того, достигнут необходимый уровень интеграции движков *mate-keyring* и *libmatekeyring* (после их обновления), которые позволяют работать со связками ключей – до обновления у пользователей бывали с ними проблемы. Также добавлен новый пакет *mate-icon-theme-faenza*, с новой темой *Faenza* для иконок рабочего окружения Mate. Осуществлено ветвление библиотеки *libwnck* (для написания просмотрщиков и программного обеспечения для работы со списком задач), которая стала называться *libmatewnck*. Заодно разработчики исправили немало ошибок.

Среди недостатков можно отметить неполноту перевода интерфейса на русский язык. Например, подсказка для кнопки переключения между навигацией с помощью кнопок и текстовой строкой для указания пути отображается по-английски.

## Планы на будущее

Проект Mate призван продолжить развитие рабочего окружения в традициях *Gnome 2*. Насколько это получится успешным, пока судить рано – слишком малым представляется срок существования проекта. У проекта есть определенные планы на будущее. Скажем о них, придерживаясь последовательности выхода будущих версий. Так, в версии 1.6 разработчики планируют заменить *MateConf* на *GSettings* ([http://wiki.mate-desktop.org/mateconf\\_to\\_gsettings](http://wiki.mate-desktop.org/mateconf_to_gsettings)), а *MateCorba* – на *DBus*. Для файлового менеджера *Caja* планируется улучшить работу со съемными устройствами, а также обновить интерфейс поддержки *DBus*. Разработчики Mate обещают для большинства тем добавить поддержку *GTK3*. Из версии 1.4 был удален апплет *indicator-applet-session* – в версии 1.6 ему на смену должен прийти *applet mate-session-applet*.

Кроме того, разработчики отмечают, что появится начальная поддержка *GTK3* (это, конечно, станет основным новшеством), но, к сожалению, не указывают, с какой именно версии, а относят ее к будущим релизам. То же касается миграции *libunique* (библиотека позволяет создавать приложения, имеющие один экземпляр) на *GtkApplication*.

Стоит отметить один важный момент. Согласно «дорожной карте» развития проекта (<http://wiki.mate-desktop.org/roadmap>), разработчики полностью осуществили все планы на релизы версий 1.2 и 1.4. Так что можно ожидать реализации указанных выше возможностей в версии 1.6.

Лидер команды Linux Mint Клеман Лефевр [Clement Lefebvre] сказал следующее (см. пост в блоге Linux Mint от 1 декабря 2011 года): «Мы считаем, что Mate – такой же рабочий стол, как и KDE, Gnome 3, Xfce и т. д. В связи с высокой популярностью Gnome 2 в предыдущих версиях Linux Mint, мы стремимся поддержать и улучшить проект». **LXF**

LINUX FORMAT Вердикт

---

**Mate 1.4**

**Разработчик:** Проект Mate  
**Сайт:** <http://mate-desktop.org/>  
**Цена:** Бесплатно, под свободными лицензиями

<b>Функциональность</b>	9/10
<b>Производительность</b>	9/10
<b>Удобство использования</b>	9/10
<b>Документация</b>	9/10

» Удачная ветка проекта Gnome 2, с совместным использованием приложений. Не хватает полного перевода интерфейса на русский язык.

Рейтинг 9/10



ИМ ВСЁ НЕЙМЁТСЯ...

## Держись, любимая ОС!

Компания Apple требует от Samsung и Honeywell выплаты лицензионных отчислений.

Патентные войны продолжаются. В рамках проходящего судебного разбирательства между компаниями Apple и Samsung компания Apple опубликовала документ, свидетельствующий о том, что Samsung было предложено выплатить за каждый проданный Android-телефон по \$30 и за каждый проданный Android-планшет по \$40 лицензионных отчислений за использование запатентованных технологий Apple. Эти деньги Apple пыталась получить за патенты, описывающие столь тривиальные вещи, как выявление и выделение телефонных номеров в тексте, система автоматического дополнения набираемых слов, разблокировка телефона и прокрутка фотографий экранными жестами, дизайн в виде прямоугольника со скругленными углами. Samsung отказался выплачивать данные отчисления, после чего компания Apple развязала судебную тяжбу и пытается через суд заблокировать продажи Android-устройств Samsung во многих странах.



» Профессиональный коммуникатор на основе Android скоро будет стоить дороже...

А вот компания Honeywell, крупнейший американский производитель средств управления и автоматизации, объявила о заключении с Microsoft патентного соглашения, в рамках которого компания признала наличие в платформах Android и Chrome OS технологий, нарушающих патенты Microsoft, и согласилась выплачивать лицензионные отчисления за каждое проданное устройство, укомплектованное Android и Chrome OS. Сумма отчислений держится в тайне. Примечательно, что о сделке объявлено одновременно с анонсом начала по-

ставки варианта промышленного коммуникатора Dolphin 7800 на базе платформы Android, ранее

данное устройство комплектовалось ОС Windows Embedded Handheld 6.5.

В настоящее время около 70% из продающихся в США устройств на базе платформы Android выпускаются компаниями, выплачивающими дань Microsoft. Сделки за закрытыми дверями, касающиеся этих выплат, ранее были подписаны с компаниями Samsung, Acer, ViewSonic, HTC, LG, Quanta Computer, Velocity Micro, General Dynamics, Onkyo, Wistron Corp, Coby, Aluratek и Compal Electronics. Во всех случаях список фигурирующих в соглашениях патентов и финансовые детали сделок не разглашаются.

Из крупных производителей, не заключивших соглашение с Microsoft, остались только Motorola, Huawei и Sony.

**«Список патентов и финансовые детали не разглашаются.»**

### ХАКИНГ СО СКАНЕРОМ

## Ваш смартфон под угрозой

Появилась техника удаленного взлома телефонов на базе Android и MeeGo через NFC.

На прошедшей в конце июля конференции хакеров Black Hat продемонстрирована техника атаки на телефоны под управлением платформ Android (Samsung Nexus Galaxy) и MeeGo (Nokia N9) с использованием беспроводного канала связи NFC (Near Field Communications, «коммуникация ближнего поля»), позволяющего передавать данные на расстояние около 10 см.

Атака позволила получить полный контроль над телефоном при непосредствен-

ном приближении аппарата жертвы к устройству атакующего.

Причиной продемонстрированных проблем безопасности является серия уязвимостей в приложениях, вызываемых при использовании NFC. Для телефонов

на базе платформы Android система была атакована через задействование функции автоматического открытия сайта в браузере при чтении определенной NFC-метки (подразумевается, что пользователь пожелал узнать подробности о товаре).

В качестве сайта передавалась ссылка на открытие страницы с эксплойтом, использующем уязвимость в браузере Android (уязвимость исправлена в последних выпусках Android, но, по данным исследователя, 90% телефонов продол-

**«Атака позволила получить полный контроль над телефоном.»**



АЛЬТЕРНАТИВНАЯ СИСТЕМА

# Не нравится готовый интерфейс?

Попробуйте релиз CyanogenMod 9, независимую сборку мобильной платформы Android.

Представлена стабильная версия альтернативной сборки платформы Android – CyanogenMod 9.0, развиваемой независимым сообществом. CyanogenMod 9 является первым выпуском, основанным на открытой кодовой базе Android 4.0. Все усилия при подготовке новой версии были сосредоточены на миграции оригинальных наработок CyanogenMod на новую кодовую базу. Сообщается, что ветка CyanogenMod 9 в дальнейшем развиваться не будет: все усилия сразу будут переброшены на развитие ветки CyanogenMod 10, которая уже разрабатывается на базе Android 4.1. Ветка CyanogenMod 9 выпущена в качестве промежуточного стабильного релиза для тех, кто не хочет ждать, пока CyanogenMod 10 окончательно стабилизируется. В дальнейшем также будет продолжено формирование корректирующих выпусков для ветки CyanogenMod 7.2, основанной на Android 2.3.

В настоящее время CyanogenMod 9 поддерживает установку на 45 моделей устройств. Так как системные требования Android 4 существенно возросли по сравнению с Android 2.3, общее число аппаратов, для которых доступен CyanogenMod 9, немного меньше числа устройств, поддерживаемых CyanogenMod 7.2. Для каждого поддерживаемого устройства доступно



» CyanogenMod предоставляет больше расширенных возможностей, нежели Android по умолчанию.

пошаговое руководство по замене штатной прошивки на CyanogenMod; для упрощения установки подготовлено специальное приложение ROM Manager, входящее в комплект CyanogenMod.

CyanogenMod 9 поддерживает все базовые возможности платформы Android 4.0, но по сравнению оригинальными прошивками Android содержит ряд полезных для энтузиастов и разработчиков улучшений.

Ключевым изменением CyanogenMod 9 по сравнению с веткой CyanogenMod 7.2 является переработка подхода к организации интерфейса взаимодействия с пользователем, направленная на упрощение работы пользователя и предоставления более ясного понимания, что про-

исходит в настоящий момент. Например, существенно переработана организация работы с настройками, проведена реструктуризация категорий, расширено описание сути представленных опций. В интерфейсе также отражены элементы нового оформления оболочки пользователя Android 4.0 и наработки по унификации интерфейсов для смартфонов и планшетов.

Среди других улучшений, заимствованных из Android 4:

- » новый интерфейс управления задачами с поддержкой принудительного завершения приложений;
- » переработанная область уведомлений;
- » программный интерфейс для распознавания лиц;
- » Android Beam для быстрого обмена сообщениями по NFC;
- » режим Wi-Fi Direct для связи двух устройств;
- » наглядная система контроля расхода трафика;
- » улучшенное приложение для работы с камерой;
- » расширение возможностей и производительности web-браузера;
- » унифицированный инструментальный для формирования интерфейса пользователя телефонов, планшетов и других устройств;
- » единый набор компонентов интерфейса и стилей.

шают сохранять старые прошивки и, следовательно, подвергнувшись атаке, могут быть взломаны).

В Nokia N9 была использована похожая схема, только вместо открытия веб-страницы был инициирован показ изображения или файла с документом, что вызывало переполнение буфера в программах для просмотра изображений и документов. После совершения успешной атаки был получен полный доступ к файловой системе, возможность отправлять SMS и совершать телефонные звонки на платные номера.

Примечательно, что недавно в ядре Linux было исправлено несколько опасных уязвимостей в NFC-стеке, которые позволяли организовать выполнение кода через отправку на устройство жертвы специально оформленных пакетов с полями некорректной длины. LXF



» Именно таким образом могут взломать и ваш смартфон на Android.

## Что такое NFC?

Near Field Communication – это технология беспроводной высокочастотной связи малого радиуса действия, обеспечивающая обмен данными между устройствами на расстоянии около 10 см (около 4"); по сути, это расширение стандарта бесконтактных карт, объединившее интерфейс смарт-карты и считывателя в одном устройстве. Устройство NFC поддерживает связь с существующими смарт-картами, считывателями стандарта ISO 14443 и с другими устройствами NFC, и совместимо с инфраструктурой бесконтактных карт, применяемой в общественном транспорте и платежных системах. NFC нацелена прежде всего на использование в мобильных телефонах.

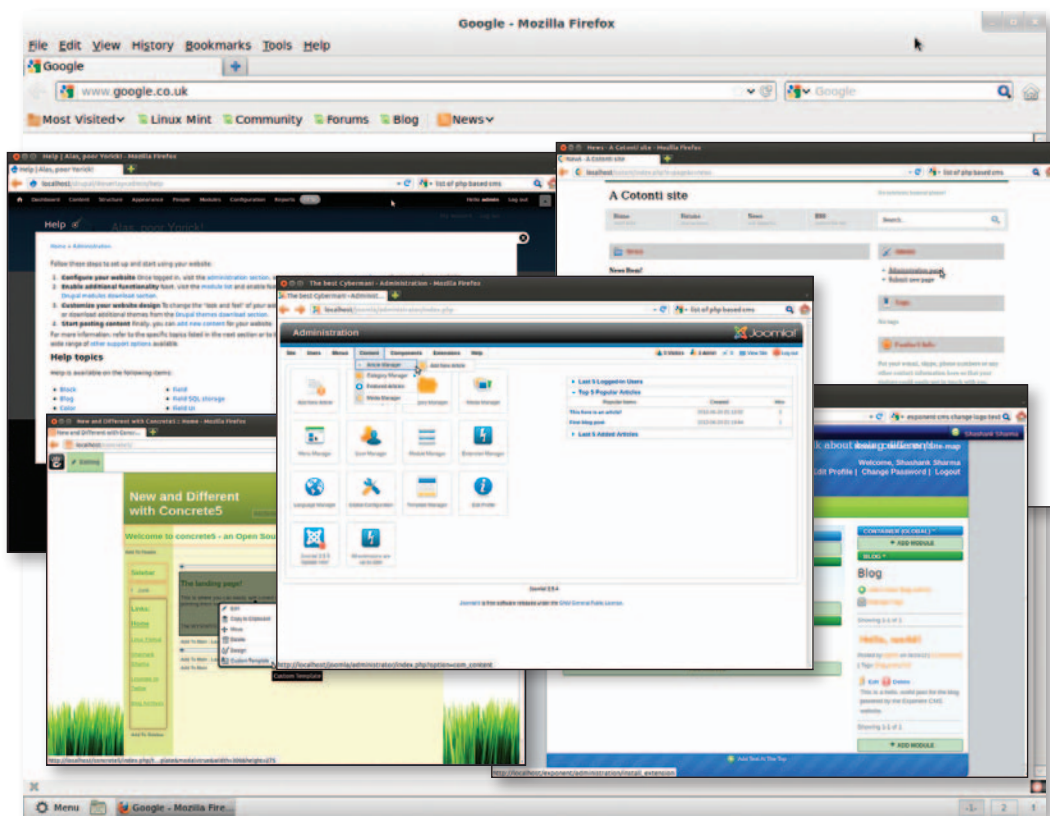
# Сравнение

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

## Получите новый CMS

Затрудняетесь в выборе CMS для вашего сайта?

Шашанк Шарма, Кибермен, перечислит вам достойные варианты.



### Про наш тест...

В отличие от приложений рабочего стола, CMS весьма нетребовательны в плане системных ресурсов. Им не нужно большого объема ОЗУ или много места. Просто дайте им поддерживаемую версию *Apache*, *MySQL* и языка скриптов вроде *PHP*, и большинство из них с радостью выполнят все, что вы пожелаете.

При отборе CMS для этого сравнения мы сознательно решили избегать специализированных. То есть CMS исключительно для ведения блогов и *wiki* не были сюда включены. Мы ищем проект, подходящий для большинства пользователей, от частных лиц до школ и среднего и малого бизнеса.

Мы тестировали каждую CMS по части простоты установки и использования, количества предлагаемых модулей расширения и того, что они могут сделать сами по себе. Для CMS также важно предлагать достаточное количество тем и быть в общем и целом настраиваемым, чтобы вы смогли придать ему неповторимый вид.

### Наша подборка

- » Concrete5
- » Cotonti
- » Drupal
- » Exponent CMS
- » Joomla

**М**ногообразные и уникальные, как и большинство систем управления контентом, и практически такие же разные, как реинкарнации Доктора Кто, все они существуют для одной цели — помочь вам настроить сайт с минимумом хлопот. Системы комментариев и контактных форм, создание опросов, размещение фото, поддержка блога и ведение электронной коммерции... CMS помогут вам создать сайт почти для любой надобности.

О большинстве CMS стоит знать то, что они подобны Тардис [Tardis — машина вре-

мени и космический корабль из сериала «Доктор Кто», снаружи она значительно меньше, чем внутри, — прим. пер.] — их содержимое всегда значительно больше, чем можно предположить, исходя из размера пакета. А еще есть модули расширения, которые добавляют функции, не включенные по умолчанию.

И хотя основанные на *PHP* CMS продолжают лидировать в этой области, за последние годы появилось немало других. В настоящее время имеется немало опций на базе *Perl*, *Java* и *Ruby*. Язык, на котором они основаны, не столь важен для

конечного пользователя и имеет значение только для разработчиков, намеренных внести улучшения в код или добавить какие-то функции. А все пакеты в нашем списке совершенно непредумышленно основаны на *PHP*.

Со временем сайты растут над собой, и CMS следует выбирать в уверенности, что он просуществует долго. Кроме того, он должен обеспечивать функции анти-спама, встроенные или в виде модулей расширения, и должен быть прост в работе, обновлении и модификации, чтобы вы могли настроить его по своему вкусу.

# Установка

Нам подавайте побыстрее и попроще!

**Б**ольшинство систем управления контентом могут работать на разных платформах, в том числе на Linux, Windows и Mac. Все, что им нужно в плане зависимостей – это поддерживаемые версии Apache, MySQL или PHP.

Однако прежде чем приступить к установке из браузера, надо создать для CMS

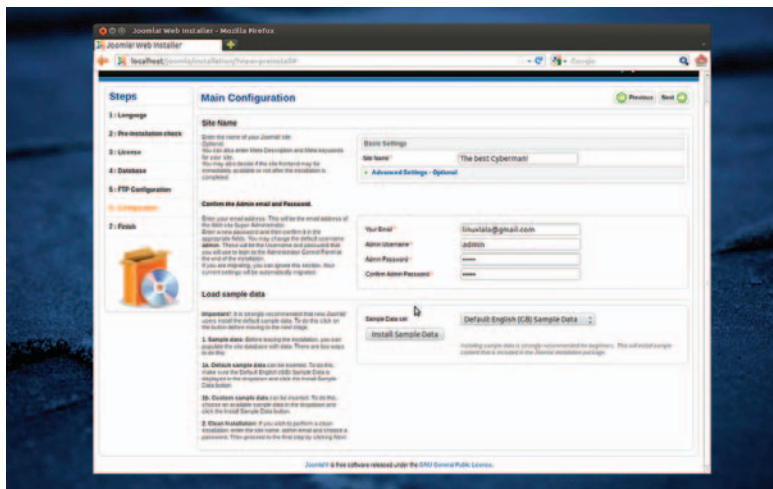
базу данных и убедиться, что разрешения на файлы даны в правильном порядке. Рекомендуется также создать отдельного пользователя для администрирования базы данных. Это важное требование для многих CMS, и, к счастью, файл установки или readme, включенный в пакет, содержит более подробную информацию.

Для *Exponent CMS* установка сорвалась на полпути без отображения сообщений об ошибке. При выборе темы *Exponent* дает вам шанс добавить образец контента на сайт. Это безусловно удобно, и почти все CMS предоставляют вам такую возможность. Однако при выборе этой возможности и темы установка срывается. Мы перепробовали разные темы, с добавлением контента и без добавления, и во всех случаях установка рухнула, кроме темы Retro. Но после установки вы сможете переключиться на любую другую тему без всяких проблем.

*Concrete5* предлагает самую простую установку, которая завершится быстрее, чем вы ухитритесь произнести «Эйяфьядлайокудль».

Сильнее всего нас впечатлил своей установкой *Cotonti*: она отличается большей интерактивностью, чем у *Concrete5*.

Во время установки вы можете выбирать, какие блоки предусмотреть – например, Spam control, SEO, облака тэгов и т.д. Предлагается большое количество опций с удобными описаниями, которые объясняют смысл разных блоков и то, что они делают.



Обычно установка требует от вас ненамного больше, чем название сайта, выбор темы и данные об учетной записи администратора.

**Вердикт**

- Concrete5 ★★★★★
- Cotonti ★★★★★
- Drupal ★★★★★
- Joomla ★★★★★
- Exponent CMS ★★★★★

» Установка Joomla и Drupal проста, и скорость не хуже прочих.

# Наличие модулей расширения

Расширения бесплатно, дешево и... ну очень дорого!

**У**становка по умолчанию для всех CMS предоставляет разные блоки, но вы можете и дальше расширить список функций в каждом из них с помощью модулей расширения. Благодаря тысячам дополнений вы легко превратите рядовую установку *Drupal* в шедевр

для обслуживания данных. *Joomla* тоже предлагает тысячи модулей, но не все они бесплатны.

Хотя остальные не настолько богаты дополнениями, у них у всех есть модули для e-Commerce и SEO, вместе с очень популярными модулями антиспама и look-and-feel.

Хотя *Concrete5* является свободной программой с открытым кодом, ее модули расширения, как и у других CMS, предлагаются не бесплатно. Но не волнуйтесь: существует также и множество даровых модулей, и если вам потребуется, чтобы на сайте появилась лен-

та Twitter, с вас не возьмут ни гроша. Цена варьируется от \$15 за модуль для импорта/экспорта метатэгов в файл CSV до \$125 за официальное дополнение для e-Commerce.

В отличие от других CMS, где вы можете загрузить модуль, скинуть его в соответствующую директорию и запустить его, *Concrete5* имеет более трудоемкие требования. Сначала вы покупаете модуль, потом загружаете его с администраторского интерфейса своего сайта и, наконец, активируете его. Это несложно, но требует больше работы, чем остальные инструменты. При просмотре рыночных предложений можно также проверить обзоры и документацию на предмет имеющихся дополнений.

*Exponent CMS* и *Cotonti* предлагают достаточно модулей расширения, чтобы помочь вам настроить полнофункциональный сайт, но все же некоторых важных модулей им не хватает.



Перед установкой модулей расширения убедитесь, что они совместимы с вашей версией CMS.

**Вердикт**

- Concrete5 ★★★★★
- Drupal ★★★★★
- Joomla ★★★★★
- Exponent CMS ★★★★★
- Cotonti ★★★★★

» Рыночное предложение Concrete5 включает всякие скидки и акции.

# Удобство и производительность

Просто ли с ними работать администраторам и пользователям?

**В** таких проектах, как CMS, простота в использовании, возможно, является самым важным фактором, влияющим на выбор. Дизайн CMS определяет ее подход к созданию контента. Хорошая CMS должна как можно более упрощать создание и редактирование контента, в то же время не являясь чересчур сложной в администрировании. Найти нужный баланс между пользователями и администраторами, чтобы и те и другие остались довольны – трюк хитрый. CMS – сложные организмы

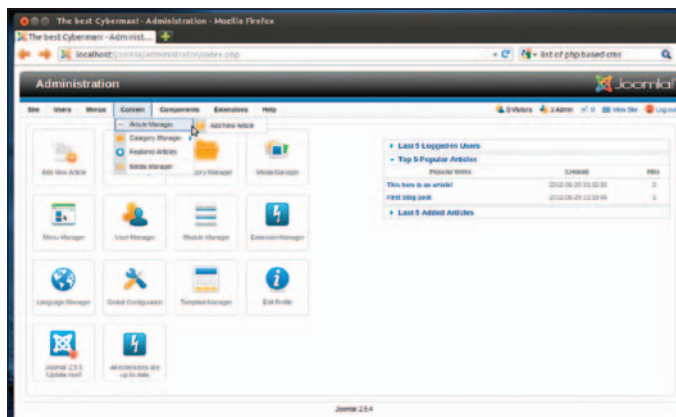
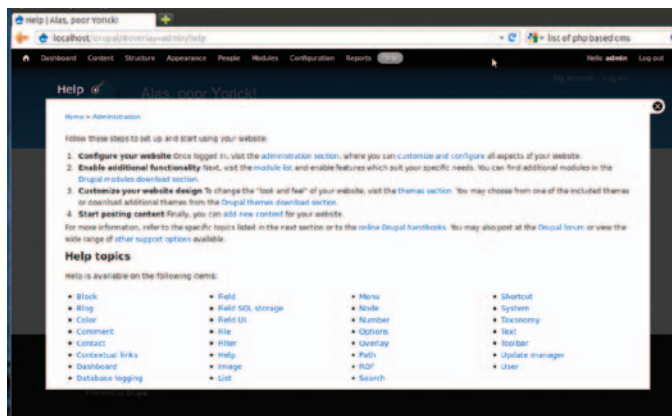
со всякими-разными блоками, темами и файлами настройки, увязанными между собой. Трудно даже представить себе панель Administration, которая помогает осуществить навигацию от одного элемента к другому, не заблудившись, и все же некоторые проекты идут еще дальше, чтобы облегчить эту задачу.

Так же, как и с любым другим ПО или приложением, не помешает определенная степень знакомства с лежащей в его основе технологией и языками.

## Drupal ★★★★★

Войдя в систему как администратор, вы сможете обратиться к настройкам с панели Administrative наверху страницы. Эта постоянно отображаемая панель предоставляет быстрый доступ к разным областям администрирования, таким, как Appearance [Вид], Modules [Блоки] и Configuration [Настройка].

Чтобы добавить на сайт контент, просто щелкните по кнопке Content на панели Administrative. Просматривая опубликованный контент, вы заметите маленькую кнопку справа. Если на нее нажать, обнаружатся две опции: Edit [Редактировать] и Delete [Удалить]. Вы можете редактировать созданный вами контент, не путаясь с администраторской кухней. При наличии обновлений *Drupal* сообщит вам о них, если вы нажмете на Configuration. Вы также можете настроить график проверки наличия обновлений и индексирования контента, нажав на Configuration > Cron. Установка включает документацию, вызываемую кнопкой Help на панели Administrative.



## Joomla ★★★★★

Экран приглашения на главном сайте – только для пользователей. Чтобы попасть в раздел Administrative, перейдите на <http://yoursiteaddress/administrator> или <http://yoursiteaddress/joomla/administrator>, в зависимости от вашей установки. В отличие от простой панели Administrative в *Drupal*, *Joomla* предпочитает выпадающие списки и кнопки. Войдя в систему как администратор, вы попадете в Control Panel. Отсюда можно перейти в любую область. Альтернативно, панель сверху использует выпадающие списки, для быстрой навигации по вложенным разделам. Например, чтобы создать нового пользователя, щелкните Users > User Manager > Add New User [Пользователи > Менеджер пользователей > Добавить пользователя]. Control Panel также предоставляет подробную информацию по доступным обновлениям *Joomla* и установленным расширениям, и их можно обновлять автоматически. И можно просматривать документацию прямо из интерфейса администратора, нажав Help.

# Документация/поддержка

Вы в замешательстве? Принимайте помощь.

**П**омимо документации, проекты имеют собственные wiki и форумы, где вы можете пообщаться с членами сообщества и найти решение для возникших проблем.

Также вы найдете руководства и статьи, предоставляемые членами сообщества, которые описывают базовую конфигурацию и управление и делятся разными хитростями и советами. Есть и руководства на YouTube, и хотя они иногда касаются более старых версий, но сохраняют по-

лезность. *Concrete5* также проводит курсы – уроки по таким темам, как управление файлами и оптимизация серверов. Однодневное занятие стоит \$295, но есть и иные опции. Вы можете напрямую озадачить разработчиков и поощрить их чтобы сделать по вашему требованию.

По *Drupal* почти все издатели технической литературы, от O'Reilly до Apress, выпустили одну-две книги. Активное сообщество и подробная документация – это дополнительный бонус, и то же можно

сказать о *Joomla*. Для обеих можно найти несколько компаний, способных помочь с внедрением, настройкой и применением модулей; часть их перечислена на сайте.

Документация *Exponent CMS* предлагает объяснение различных терминов и технологий, но на этом практически все. Вы не найдете здесь подробных процедур или руководств по решению сложных задач, проиллюстрированных экранными снимками. У *Cotonti* документация явно оставляет желать лучшего.

**Вердикт**

<b>Concrete5</b>	★★★★★
<b>Drupal</b>	★★★★★
<b>Joomla</b>	★★★★★
<b>Exponent CMS</b>	★★★★★
<b>Cotonti</b>	★★★☆☆

» Более подробная документация создала бы *Exponent CMS* популярность.



**Cotonti** ★★☆☆☆☆

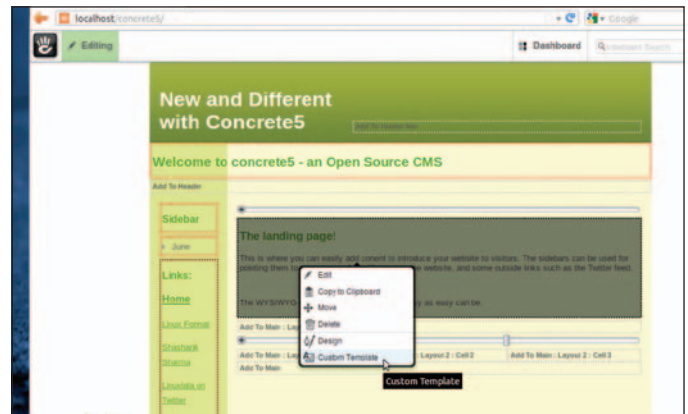
Войдя в систему как Admin, щелкните в панели Administrative по правой панели для доступа к интерфейсу администратора. У *Cotonti* он выглядит примитивным. Вместо удобных выпадающих списков и умных пунктов меню здесь имеются большие кнопки для Configuration, Structure, Extensions и т.д. Отсутствие хорошей документации тоже не помогает. На самом деле, проект проводит конкурс на статьи и руководства, предлагая приз за первое место \$100.

Если хотите добавить новый контент, вместо больших кнопок сверху обратитесь к правой боковой панели и щелкните Add [Добавить] под Pages [Страницы]. Сверх блока Forums, позволяющего за пару минут создавать полнофункциональный форум, *Cotonti* особенно нечего предложить обычным пользователям. Не рекомендуется для лиц, которые не являются разработчиками.

**Concrete5** ★★★★★★

*Concrete5* допускает встроенное редактирование контента. Войдя в систему, щелкните по кнопке Editing [Редактирование] в верхней левой части экрана, после чего можете редактировать все элементы страницы. Так, чтобы добавить контент в боковую панель, щелкните по кнопке Add to Sidebar, а чтобы редактировать элемент, щелкните по нему и выберите Edit. Вы также можете изменять дизайн для каждого элемента, щелкнув по элементу и выбрав Design. Это настолько просто и эффективно, что даже странно, почему остальные CMS не предлагают таких же функций.

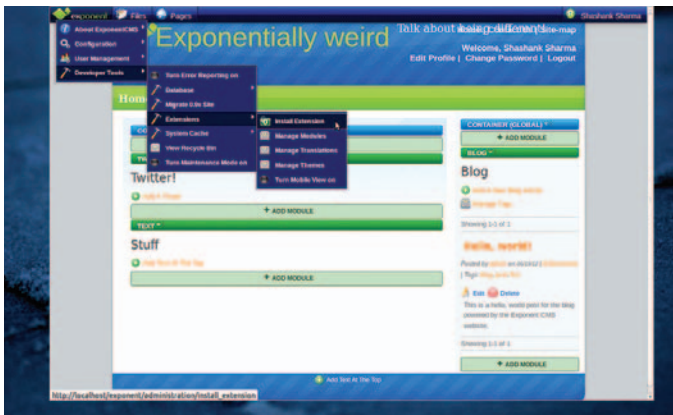
Для доступа к другим аспектам сайта наведите мышь на кнопку Dashboard сверху справа. Для внесения особых изменений на сайт, вроде настройки заданий cron или обновления *Concrete5*, щелкните по System and Settings. Будь то добавление или изменение контента или настройка сайта, работа с *Concrete5* – истинное удовольствие.



**Exponent CMS** ★★★★★★

Подобно *Concrete5*, *Exponent CMS* допускает встроенное создание и редактирование контента. Благодаря своей структуре, *Exponent* рассматривает все элементы страницы как различные блоки. Поэтому любой тип контента, блог или простой текст, рассматривается в виде отдельного блока. Добавив блок, вы затем можете добавлять в него контент. Это несколько непривычно, но встроенное редактирование значительно облегчает жизнь.

Вы можете перейти к другим функциям администрирования с помощью кнопок в верхней левой части экрана. Кнопка Exponent сверху слева отображает выпадающий список, предоставляющий быстрый доступ к управлению пользователями, настройке сайта и расширениям. Чтобы изменить текст на изображении логотипа наверху страницы, вам придется вручную редактировать тему. Найдите файлы `index.php` и `config.php` в соответствующей тематической директории.



# Персонализация и темы

Не нравится? Меняйте!

Если вам несимпатичны темы CMS по умолчанию, всегда можно скачать другие. Более того, владея навыками работы с HTML и CSS, вы можете изменить тему по вкусу или надергать элементы из разных тем и создать собственного монстра-химеру.

Большинство CMS в нашем списке рассчитывает на ваше копание в файлах вручную, даже для внесения самых банальных изменений – вроде цвета фона для блока комментариев. *Exponent CMS*, по-

добно *Cotonti*, не предоставляет редактор WYSIWYG для работы с темами, но если вы понимаете HTML и CSS, то вас ничто не ограничивает в плане изменений дизайна. Однако без таких знаний документация вас не спасет.

Многие шаблоны и дизайны для *Joomla* разработаны третьими сторонами на продажу. Продается и графический редактор.

*Concrete5* предлагает для редактирования тем минималистский редактор WYSIWYG, позволяющий графически ме-

нять цвета для текста и фона и даже вводить собственный код CSS прямо в браузере. Если в соответствующих файлах можно писать, то *Concrete5* самостоятельно добавит ваши дополнения в таблицу стилей.

Хотя и непохожий на *Concrete5*, *Drupal* тоже позволяет менять цвет всех элементов на странице: верхнего и нижнего колонтитулов, заглавия, фона боковой панели, и т.п., а также их интерфейса администрирования.

**Вердикт**

- Concrete5 ★★★★★★
- Drupal ★★★★★★
- Exponent CMS ★★★★★★
- Joomla ★★★★★★
- Cotonti ★★☆☆☆☆

» Всем проектам есть чему поучиться у *Concrete5*.

# Функции

## Заработают ли они с ходу?

**В** зависимости от требуемого вам типа сайта, вам, возможно, придется потрудиться над установкой по умолчанию, чтобы CMS смогла сделать все, что вы задумали. Но это не в порядке критики. Упрощение создания сайтов – благородная цель, и все системы стремятся именно к ней, но не стоит ожидать, что CMS обслужат все виды контента точно по вашим запросам сразу, без всякой подготовки.

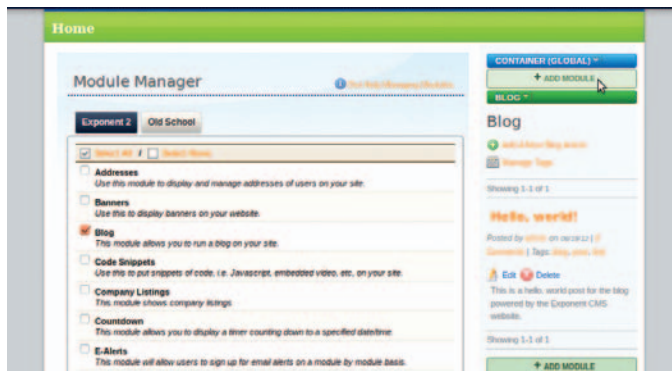
В *Exponent CMS* все работает в виде блоков. Блоги, форумы, поиск по форумам и т.д. – все это блоки, которые нужно включить, чтобы ими воспользоваться. Блок Email Alerts можно использовать для предупреждений по электронной почте для остальных блоков. Это означает, что вместо обычного предупреждения «на сайте появился новый контент» пользователи могут получить информацию об обновлениях в интересующих их областях сайта, будь то блоги, ссылки на скачивание или новости. Вы можете импортировать/экспортировать базу данных, что облегчает миграцию сайтов.

В *Concrete5* можно работать с блогом и генерировать статические страницы. Для каждой страницы и боковой панели вы можете добавлять стеки и блоки – именно так *Concrete5* идентифицирует типы контента. Изображения, поисковые формы, скачи-

вание файлов, внешние ссылки и ленты RSS – все это считается блоками. Вам может понадобиться приобрести другие стеки и блоки сторонних разработчиков. Система позволяет сделать резервные копии базы данных, из которых вы сможете восстановить свой сайт в случае какой-либо неприятности.

*Cotonti* позволяет работать с форумами, опросами и лентами RSS – они классифицируются как модули по умолчанию, установленные, но не включенные. Из панели Administrative [Административная] щелкните по Extensions [Расширения] и включите модули, которыми хотите воспользоваться. Помимо этих модулей есть и другие, такие, как контакты, рейтинги звездочками и новости, которые вы можете установить и включить.

Прямо сразу, без подготовки, *Joomla* может создавать статьи и статические страницы, но это, собственно, и все. Конечно, вам позволяется загружать изображения вместе с каждой написанной вами статьей, но даже для простейших действий с сайтом придется устанавливать расширение.



» В *Exponent CMS* мы рекомендуем потратить время на ознакомление с документацией, чтобы понять различные технические термины, такие, как блоки и стеки.

*Drupal* устанавливает несколько модулей по умолчанию, но не включает их. Включите модули блога, контакта и опроса, и установите другие, если вы собираетесь делать что-то сверх создания статических страниц. Для резервного копирования и восстановления базы данных вам придется устанавливать внешний модуль.

Все проекты имеют расширенные опции управления пользователями, позволяющие определять разрешения для зарегистрированного пользователя. Каждый проект обеспечивает и другие настройки безопасности, такие, как блокировка IP-адресов и создание черных списков.

### Вердикт

- Drupal ★★★★★
- Exponent CMS! ★★★★★
- Concrete5 ★★★★★
- Cotonti ★★★★★
- Joomla ★★★★★

*Exponent CMS* идеален для всех, у кого малый опыт работы с CMS.

# Представление данных

## Главные инструменты товарообмена.

**П**о мере роста количества людей, соединяющихся с Интернетом через телефоны и другие устройства, многие сайты завели версии для мобильных. Поскольку цель CMS – создание сайтов, они должны помочь вам создать и отдельную мобильную версию. Обычно это означает создание двух разных дизайнов для сайта, а затем CMS определит вид устройства, с которого осуществляется доступ на ваш сайт, и отобразит идеальную для него версию.

Также, когда вы ставите свой сайт на техобслуживание, вам не нужно вручную создавать сообщение, информирующее об этом посетителей – CMS должна послать его сама. За исключением *Concrete5*, где для такого сообщения требуется внешний модуль, все остальные

с этим справятся. В *Concrete5* вам придется скачать и установить одно из множества бесплатных дополнений для определения устройства посетителя. Можете поместить сайт в режим Maintenance [Техобслуживание], нажав Dashboard > System Settings > Maintenance Mode [Панель инструментов > Системные настройки > Режим обслуживания].

Чтобы перевести сайт, построенный на *Exponent CMS*, в режим Maintenance, нажмите Exponent > Developer Tools > Turn Maintenance Mode on [... > Средства разработчика > Включить режим обслуживания]. Есть модуль, переключающий темы, но все остальное вам придется делать вручную, если вы хотите создать мобильную тему для своего сайта. В создании мобильных версий по простоте нет равных

*Cotonti* – просто установите модуль Cotonti Mobile, и все! Чтобы перевести сайт в режим Maintenance, включите опцию из Configuration > Main Settings [Конфигурация > Основные настройки]. Модуль Mobile Tools предоставляет набор инструментов, которые помогут вам подготовить сайт на *Drupal* для пользователей мобильных устройств, и снабжен отличной документацией.

*Joomla* моментально предоставляет расширение в помощь созданию мобильной версии вашего сайта – Mobile Joomla. Для работы с изображениями и другими типами файлов все наши CMS имеют встроенные менеджеры файлов и медиа. Однако некоторые проекты, в частности, *Joomla*, требуют наличия у браузера поддержки Flash.

### Вердикт

- Cotonti ★★★★★
- Drupal ★★★★★
- Joomla ★★★★★
- Concrete5 ★★★★★
- Exponent CMS! ★★★★★

» У всех есть плагины для получения контента из Facebook и Twitter.

Системы управления контентом

# Вердикт

**П**роектов CMS пруд пруди, и мы решили взять лишь полезные большинству. И хотя нашей главной целью были частные лица, школы и предприятия малого бизнеса, мы постарались включить сюда проекты, легко адаптируемые для обслуживания любого сайта.

Сайты неизбежно растут с течением времени, и вы никогда не предугадаете, какой контент будете размещать на своем сайте через несколько месяцев или лет. Хорошая CMS должна по этой причине предлагать множество модулей для расширения функциональности, иначе она рискует потерять своих пользователей, которые отдадут предпочтение другой CMS.

Для любого сайта важен внешний вид, и никто не захочет, чтобы его сайт выглядел двойником других на базе той же CMS. Поэтому проекту CMS важно предоставлять достаточное количество тем и обеспечивать индивидуальную настройку. При всем нашем восхищении *Concrete5*, действительно ставшей находкой, коммерческая природа ее предложения

несколько охладил нас. Это не обязательно недостаток, особенно учитывая то, что *Joomla* тоже предлагает приобрести некоторые модули. Плюс к тому, очень хорошо, что разработчики могут зарабатывать на достойную жизнь, продавая продукты для CMS, а не только для Android и iPhone.

Если вас не устраивают платные модули расширения, то победителем для вас станет *Drupal*. Однако для нас *Concrete5* победил с заметным отрывом.

## Тестирование без проблем

Тур по экранным снимкам – прекрасный способ познакомиться с проектом, но он ничего не скажет о впечатлениях пользователя от работы с ним. Вот почему большинство проектов делают демо-версию онлайн, чтобы вы ощутили, чего ждать от проекта. Если блуждание с сайта на сайт для пробы разных демонстрационных копий кажется вам несколько утомительным, отправляйтесь



на вечно популярный [www.opensourcecms.com](http://www.opensourcecms.com), где размещены разные проекты и позволяется запустить демо для каждого. Как мы отмечали, каждому проекту CMS нужен сервер, база данных и язык скриптов. Когда вы созреете испробовать для себя CMS, рекомендуем сначала запустить ее на тестовой машине. Рассмотрите такие проекты, как *XAMPP* и *Bitnami*, которые помогут вам настроить среду, где работают изолированные экземпляры *Apache* и *MySQL* и где вы можете свободно протестировать любую CMS, какую пожелаете.

➤ **Непревзойденная пятизвездная *Concrete5* – истинное наслаждение в работе, независимо от вашего изначального набора навыков и умений.**

**«Когда вы созреете испробовать CMS, запустите ее на тестовой машине.»**

**I Concrete5** ★★★★★  
 Сайт: [www.concrete5.org](http://www.concrete5.org)  
 » Явный победитель. Не пугайтесь коммерческих модулей расширения.

**IV Exponent CMS** ★★★★★★  
 Сайт: [exponentcms.org](http://exponentcms.org)  
 » Сильный конкурент. Уязвимое место – модули расширения и документация.

**II Drupal** ★★★★★★  
 Сайт: <http://drupal.org>  
 » Хотя она следует второй за *Concrete5*, но успевает не без труда.

**V Cotonti** ★★★★★★  
 Сайт: [www.cotonti.com](http://www.cotonti.com)  
 » Неделана по сравнению с другими. Есть несколько впечатляющих функций.

**III Joomla** ★★★★★★  
 Сайт: [www.joomla.org](http://www.joomla.org)  
 » *Joomla* следует поддержать носочки и кое-чему поучиться у *Concrete5*.

**Обратная связь**  
 Вы использовали какую-нибудь из этих систем? Полагаете, что они потрясающие? Сообщите нам об этом на [lxf.letters@futurenet.co.uk](mailto:lxf.letters@futurenet.co.uk)

## Рассмотрите также...

Чтобы осознать, сколько на свете разных CMS, поищите в Wikipedia «CMS с открытым кодом». Вы найдете списки для Java, Perl, PHP, Python и пр. Мы включили лишь универсальные проекты, что отменило *WordPress* и *MediaWiki*. Хотя все наши проекты поддерживают SEO и имеют модули e-Commerce, мы ограничились теми, что не были разра-

ботаны для предприятий. А кому нужно именно это, рекомендуем *WebGUI*. Написанный на Perl, он богат модулями поддержки коммерческой деятельности: размещение объявлений о вакансиях, тесты самооценки, данные биржевых торгов, прогноз погоды, управление проектами и отслеживание времени. И все это – в дополнение к фору-

мам, опросам, блогам и прочим атрибутам CMS. Все наши проекты нацелены на создание обычных сайтов; а если вы хотите создать среду собственной социальной сети? На это есть *Elgg* – популярная программа для создания сайта микроблогов. Проекты типа *Hoops*, многофункциональные, но сложные в установке, мы игнорировали. **LXF**





# «Железо»: Полное руководство

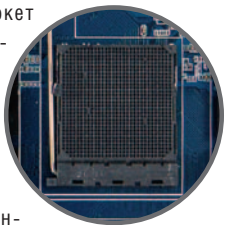
Начнем с устройств, к которым подключаются все остальные компоненты.

## 0 Материнская плата

Материнские платы бывают всевозможных форм и размеров, но большинство отвечают форм-фактору ATX. Он определяет, где должны находиться разъемы питания и где плата крепится к корпусу. Большинство плат были стандартного размера ATX, и их по-прежнему предпочитают многие, благодаря обилию возможностей расширения. Но популярна и Micro-ATX, особенно в ТВ-приставках и в автономных компьютерах. Во встраиваемых системах может использоваться Mini-ATX, но все, что меньше – удел специалистов. Для наших целей понадобятся ATX или Micro-ATX.

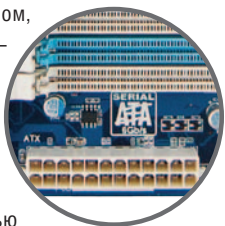
## 1 Сокет процессора (охлаждение)

Производителей процессоров на платформе x86, достойных нашего внимания, всего два – Intel и AMD, и у каждого из них большое разнообразие вариантов процессорных сокетов и разъемов для подключения кулеров. Какой сокет вам понадобится, зависит от процессора, и к нему понадобится соответствующая материнская плата. Последний сокет от Intel называется LGA1155 – он поддерживает и прошлогодние процессоры Sandy Bridge, и свежешедший Ivy Bridge. Последний сокет от AMD – AM3+, которым мы пользовались, рассматривая процессор AMD Phenom II. К обоим socketам нужны совместимые кулеры, хотя современные конструкции кулеров «адаптируются» отверткой.



## 2 Разъемы питания

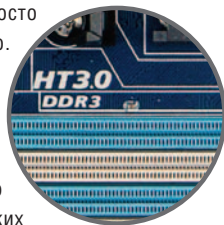
Современным компьютерам нужны и современные источники питания. Наряду с обычным 24-контактным разъемом, иногда разбиваемым на блоки из 20 и 4 контактов, вам понадобится 8-контактный/12-вольтовый разъем для процессора. Конструкция разъемов обычно такова, что в неправильное гнездо их не вставить. Дешевым видеокартам редко нужно дополнительное питание, но картам среднего диапазона может еще потребоваться питающий кабель для PCI Express с 6-контактным разъемом, а для мощной карты – даже два кабеля. Они должны идти от одного блока питания, а для мощного компьютера мы посоветуем блок питания мощностью



600 Вт с отдельными 12-вольтовыми «рельсами» для видеокарты.

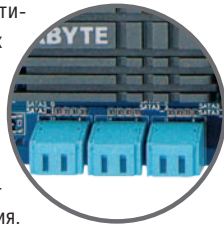
## 3 Слоты памяти

Память тесно связана с процессором и специально выбирается под вашу платформу. На современных материнских платах делать это гораздо проще – и Intel, и AMD используют одни и те же планки DDR3 (позже в этом году должна выйти DDR4), только купите память побыстрее, чем вам нужна. Если память окажется слишком медленной для процессора, она либо не будет работать, либо не даст раскрыть его возможности; если слишком быстрой, то вы просто переплатите лишнего. Мы взяли 4 Гб памяти G. Skill Ripjaw Gaming Series (F3-12800CL7D) с тактовой частотой 1600 МГц. Большинство современных материнских плат поддерживают до 32 Гб.



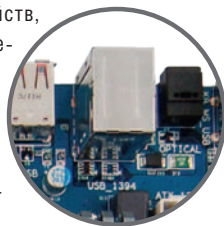
## 4 Порты SATA (2 и 3)

Старые разъемы IDE для жестких дисков и оптических приводов сейчас встречаются нечасто. На современных материнских платах все подключено через гораздо более простые разъемы SATA. Несмотря на то, что кабели у них одинаковы, многие устройства совместимы с SATA2, теоретическая скорость которого достигает 3 Гб/с, хотя на всех платах, которые нам попались, был и интерфейс SATA 3, который удвоит эту скорость, при наличии совместимого устройства хранения.



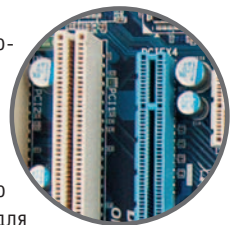
## 5 Порты USB

Аналогично, сейчас, когда все привыкли подключать все устройства через USB 2, этот стандарт медленно вытесняется USB 3. USB 3 повышает предел скорости с 480 Мб/с старого стандарта до 5 Гб/с, что позволяет ему идти в ногу с SATA 3 и значительно опередить Firewire 800. Однако передача данных – не столь однозначная тема. Скорость тут зависит от операционной системы, подключаемых устройств, драйверов для чипсета и передаваемых данных. Например, многие видеоредакторы убеждены, что Firewire 800 дает лучшую производительность, чем USB 3.



## 6 Слоты PCI

Скорее всего вы захотите подключить платы расширения в слоты PCI Express x1 или PCI Express x16. К этим последним обычно относится один слот для видеокарты с увеличенным питанием, с меткой 'PCIEX16', который расположен ближе всего к процессору, а более медленные слоты отмечены как 'PCIEX4'.



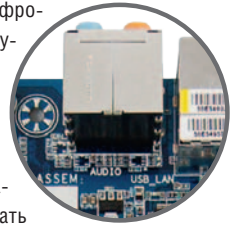
## 7 Видеовыход

Теперь, когда на многих платформах Intel и AMD есть процессор для обработки графики, на плате чаще всего имеется разъем видеовыхода. Обычно это DVI- или HDMI-разъем, который легко подключить к телевизору или современному экрану; на HDMI также есть цифровой аудиовыход.



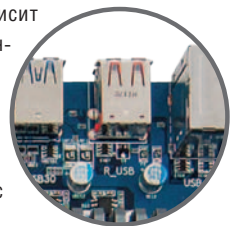
## 8 Аудиовыход

Вместе с цифровыми выходами вы найдете и аналоговые, обычно в виде оптических или коаксиальных разъемов для усилителя. На многих материнских платах для реализации звука используется чипсет Realtek, способный формировать многоканальный звук. Лучше всего сохранить весь звук цифровым, так как в этом случае при проигрывании фильмов с совместимым усилителем не потребуются никакие преобразования; вдобавок это позволит избежать помех.



## 9 Локальная сеть

Вы наверняка знакомы с проводными/Ethernet сетевыми подключениями. Предел скорости передачи данных уже некоторое время не менялся – это значит, что скорость сети существенно зависит от скорости подключенных устройств. Все современные платы теперь поддерживают подключения со скоростью 10/100 Мб/с и 1000 Мб/с (гигабит).



# Процессоры

Вычислительная мощность – это не только частота процессора.

**К**огда-то производительность процессора сводилась только к его тактовой частоте. Более быстрый процессор мог выполнять больше операций за заданный период времени и, следовательно, решать какую-то задачу быстрее медленного процессора. Тактовая частота измеряется в герцах (Гц) и представляет собой число операций, выполняемых в секунду (правда, здесь мы немного упростили – некоторым опе-

менных процессоров 32-битные или 64-битные – имеются в виду биты данных, обрабатываемые каждой командой. Значит, 64-битные вдвое быстрее 32-битных? Нет. Все зависит от того, сколько нужно вам: скорости операции над 20-битным числом на 64- и 32-битном процессоре будут одинаковы. Длина слова также может влиять на способ обращения процессора к ОЗУ. О том, как это влияет на производительность, можно прочесть во врезке «64-битные процессоры против 32-битных».

Один из важнейших аспектов производительности процессора – число ядер. По сути, каждое ядро – это отдельный процессор, на котором можно запускать программу с минимальным взаимодействием с другими ядрами. Как и с длиной слова, количество ядер нельзя просто умножить на тактовую частоту, чтобы получить мощность процессора. Задача может эффективно использовать несколько ядер, только если она разделена на несколько потоков. Это означает, что разработчик разбил ее на отдельные подпрограммы, каждая из которых может выполняться на отдельном ядре. Не все задачи поддаются такому разбиению. Программа с одним потоком будет работать на многоядерном процессоре с той же скоростью, что и на одноядерном – однако две однопотоковые программы на многоядерном процессоре будут работать быстрее, чем на одноядерном. Нам представляется, что у компьютера есть некая память, которую он делит между запущенными программами; но на практике все чуть сложнее. Память – это не единая сущность, а иерархия различных уровней. Обычно чем быстрее память, тем она дороже, поэтому в большинстве компьютеров есть малый объем очень быстрой памяти, называемой кэшем, оперативная память гораздо большего объема и раздел подкачки на жестком диске, служащий для программ и функций чем-то вроде переполнения памяти. Для процессоров наиболее важен кэш, потому что он введен в схему: можно добавить дополнительную оперативную память и изменить размер раздела подкачки, но объем кэша фиксирован. Сам кэш разбит на уровни, и более низкие уровни меньше по объему и быстрее, чем более высокие.

## «Семейство процессоров ARM использует другой состав команд.»

ряциям нужно более одного цикла процессора). Большинство современных процессоров работают с тактовой частотой в несколько гигагерц (1 ГГц = 1 000 000 000 Гц). Состав команд зависит от типа процессора. Мы рассмотрим семейство x86, применяемое в большинстве настольных компьютеров и ноутбуков. Этот состав команд появился в 1978 г. на 16-битном процессоре Intel 8086. С тех пор к основным инструкциям прибавились новые, с целью реализации новых возможностей. У семейства процессоров ARM (применяются в большинстве мобильных устройств) другой состав команд, поэтому их производительность при той же тактовой частоте будет отличаться.

Кроме количества операций, у процессоров различается и структура данных. Большинство совре-

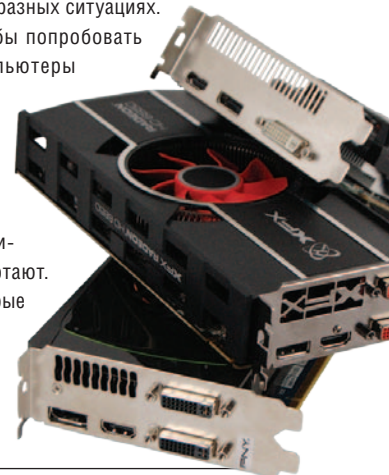
менного взаимодействия с другими ядрами.

Как и с длиной слова, количество ядер нельзя просто умножить на тактовую частоту, чтобы получить мощность процессора. Задача может эффективно использовать несколько ядер, только если она разделена на несколько потоков. Это означает, что разработчик разбил ее на отдельные подпрограммы, каждая из которых может выполняться на отдельном ядре. Не все задачи поддаются такому разбиению. Программа с одним потоком будет работать на многоядерном процессоре с той же скоростью, что и на одноядерном – однако две однопотоковые программы на многоядерном процессоре будут работать быстрее, чем на одноядерном. Нам представляется, что у компьютера есть некая память, которую он делит

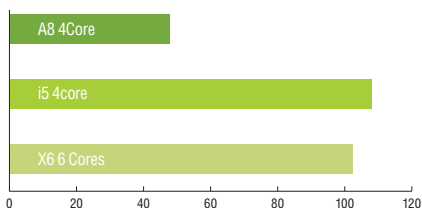
В свете всего этого нелегко понять, насколько производительными различные конфигурации могут быть в разных ситуациях.

Вместо того, чтобы попробовать понять, как компьютеры будут работать с разными конфигурациями процессора, мы запустим серию тестов и увидим, как они работают. Процессоры, которые мы будем тестировать, таковы:

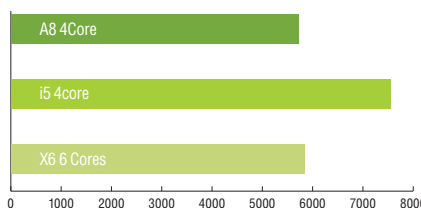
» AMD Phenom II X4 3400 МГц



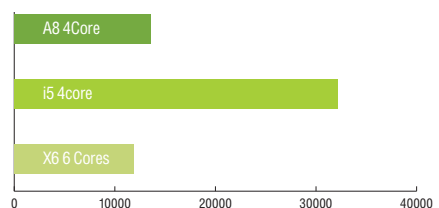
## Результаты основных тестов



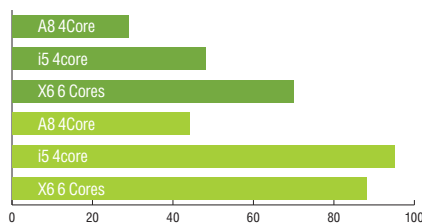
Видеокарта (Vdriфт)  
Кадров в секунду



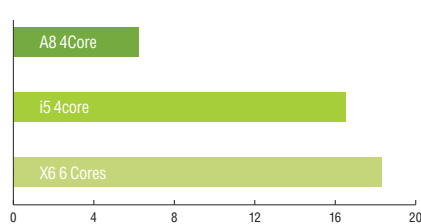
Скорость ОЗУ  
МБ/с



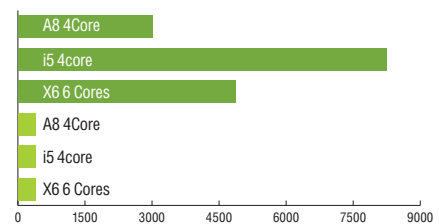
Статический Apache  
Запросов в секунду



Итераций в секунду  
Graphics Magic Sharpen  
Graphics Magic Blur



John The Ripper  
Проверок в секунду (в млн)



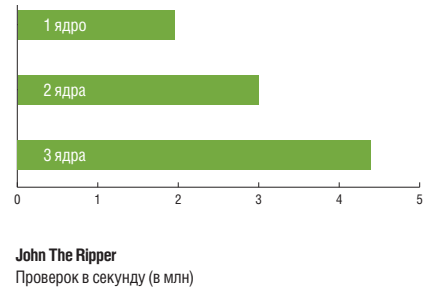
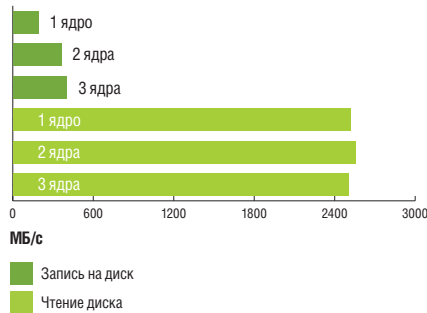
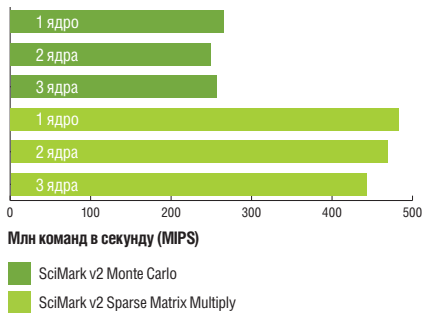
Чтение с диска (МБ/с)  
Чтение с диска 2 Гб с ioZone  
Чтение с диска 4 Гб с ioZone

## Влияние многоядерности на производительность

Влияние на производительность числа ядер можно оценить путем имитации различных процессоров в *VirtualBox*, выделив определенное количество ядер от главной системы дочерней и посмотрев, как система будет работать с произвольным числом ядер. Здесь показаны тесты системы с числом ядер от одного до трех. Увеличение производительности из-за увеличения числа ядер сильно зависит от задачи. В ряде случаев увеличение числа ядер замедляло выполнение задачи из-за накладных расходов по распределению процессов между ядрами. Иногда, например, при взломе пароля, мы наблюдаем почти линейный рост про-

изводительности с увеличением числа ядер. Стоит отметить, что мы выполняли эти тесты последовательно. Если бы мы выполняли несколько задач одновременно, многоядерные системы увеличивали бы производительность более явно.

При выборе процессора стоит учитывать, сколько интенсивных задач вы будете выполнять одновременно. Для сервера проверьте, влияет ли число ядер на производительность конкретных сервисов, которыми вы будете пользоваться. Задачи, хорошо работающие на многоядерных компьютерах, часто работают еще лучше на видеокартах с CUDA или OpenCL.



**Quad Core** (Кэш: 4×64 КБ уровень 1, 4×512 КБ уровень 2 и 6 МБ уровень 3), цена £79,00

» **AMD Phenom II X6 6 Core 3300 МГц** (Кэш: 6×512 КБ уровень 2, 6 МБ уровень 3), цена £100,27

» **Intel i5-2500K 3,6 ГГц** (Кэш: 2×32 КБ уровень 1, 256 КБ уровень 2, 6 МБ уровень 3), цена £162,43

Все процессоры мы запускали на рекомендуемых тактовых частотах. Разгон процессоров – само по себе искусство, и из каждого из них можно выжать дополнительную производительность, но данная тема находится за рамками нашей статьи.

В идеальном мире мы бы протестировали все процессоры с одинаковыми материнскими платами, чтобы исключить любые различия. Однако у разных процессоров разные схемы контактов, и они физически не подойдут ко всем платам (а если и подойдут, то не будут работать).

Мы обнаружили, что процессор Intel обогнал процессоры AMD по производительности почти во всех областях. И неудивительно: он стоит вдвое дороже самого дешевого из них. В некоторых

тестах – например, тесте статической страницы Apache – его производительность была выше вдвое. Пожалуй, более странно то, что он почти везде превзошел Phenom II X6, несмотря на то, что у него на два ядра меньше и тактовая частота лишь немногим выше. Исключениями стали тест по взлому паролей John the Ripper и некоторые тесты GraphicsMagic. Это тесты с высокой степенью распараллеливания, умеющие воспользоваться всей дополнительной вычислительной мощностью X6.

Не все различия в производительности объяснены только процессорам. Как мы сказали, они тестировались на различных материнских платах. На плате Intel есть твердотельный жесткий диск (SSD) для кэширования данных, отправляемых на главный SSD. Это привело к существенному росту скоростей чтения файлов размером до 2 Гб, но с файлами больше этого размера существенных различий не наблюдалось.

Скорость записи на различных системах была примерно равной. Выбор процессоров сегодня,

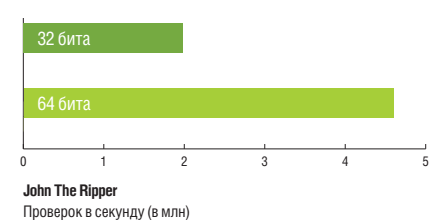
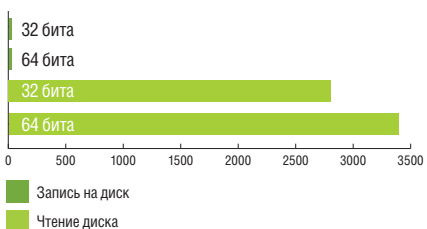
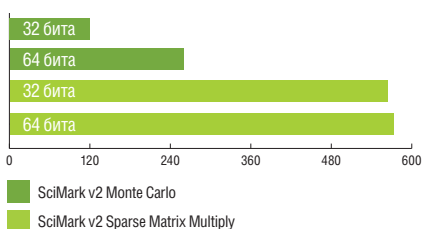
пожалуй, сложен как никогда. Появилось больше простых процессоров, процессоров с малым энергопотреблением, сложных процессоров, параллельных графических чипов и кластеров. Более актуальным, чем когда-либо раньше, становится не вопрос «Какой процессор самый лучший?», а «Какой процессор лучше всего подойдет для решения данной задачи?». Для ответа на него нужно знать, и какие процессоры представлены на рынке, и их стоимость, и то, как они выполняют различные задачи.

Ядра Intel высокого класса обладают наибольшей производительностью для решения повседневных задач, но эта скорость стоит денег. А дополнительные ядра X6 идут вровень, а иногда даже обходят i5 в тестах GraphicsMagic, имитирующей работу с изображениями, что позволяет вам сэкономить приличную сумму. Но если вы не будете пользоваться всеми виджетами и эффектами KDE, X4 более чем достаточно для решения большинства повседневных вычислительных задач.

## 64-битные процессоры против 32-битных

Даже если у вас 64-битный процессор, вы можете не ощутить преимуществ его возможностей. Для обеспечения обратной совместимости 64-битные процессоры были разработаны так, чтобы исполнять 32-битный код. Мы провели серию тестов на 64-битном процессоре с 32- и 64-битными версиями Linux,

чтобы посмотреть, как это влияет на производительность. На 64-битном процессоре в целом все работает быстрее, но для большинства задач – ненамного. При решении обычных задач вы едва ли заметите разницу, но если вы производите вычисления, то большая длина слова увеличит производительность.



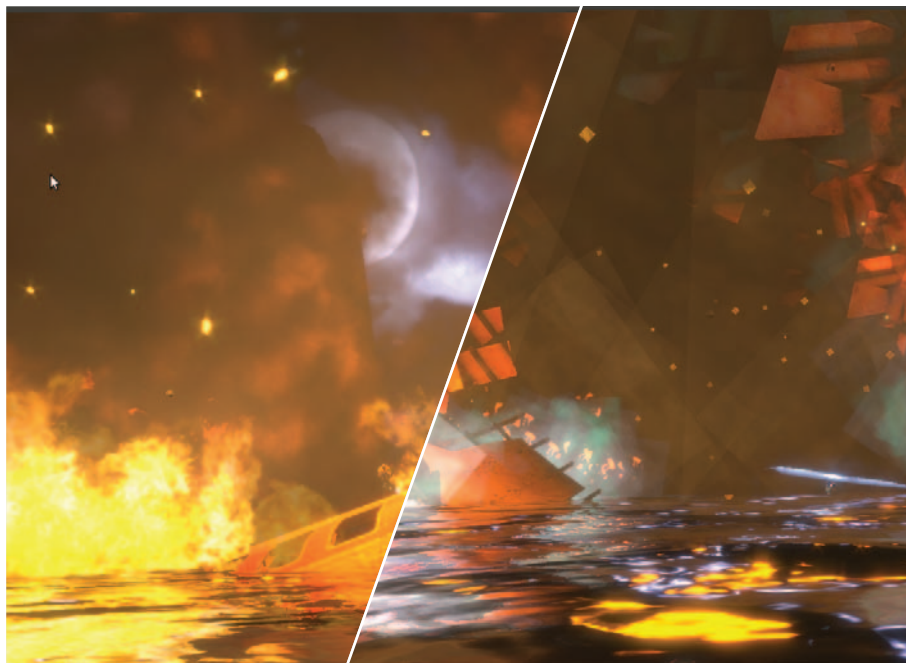
## «Железо»: Графические процессоры

С приходом Steam в Linux и ренессансом инди-игр настало время обновить вашу видеокарту.

Пожалуй, самая субъективная часть любого обсуждения «железа» – обсуждение устройств, отвечающих за генерацию графики. Все потому, что лучший выбор здесь зависит от того, насколько важна графика в вашей системе. Скажем, если вы предпочитаете командную строку или простой файловый менеджер, мощная дорогая видеокарта будет напрасной тратой денег. Дело в том, что наиболее сильны различия графических процессоров (GPU) именно в сфере 3D-графики. И хотя возможности 3D-рендеринга важны только для трехмерных игр, математические ресурсы графического процессора сейчас привлекаются к массе других задач, например, кодированию и декодированию видео с высоким разрешением, математическим операциям, воспроизведению контента, защищенного DRM, и для тех пульсирующих окон и теней, которые всем так нравятся на рабочем столе. Лучшая спецификация означает не только то, что игры будут запускаться с большим разрешением, лучшим качеством и более высокой частотой кадров – все это добавляет удовольствия игроку – но более эффективной станет и работа со всей настольной системой.

### Обработка

Развитие GPU, как и CPU, похоже, не ждет застой. Их мощность удваивается каждые 18 месяцев – это и хорошо, и плохо. Хорошо, что прошлогодние модели стоят вдвое дешевле, чем на момент выпуска. Плохо, что ваша карта почти всегда будет устаревшей, даже если вы купите самую последнюю модель. По этим причинам, а также потому, что большинство геймеров в Linux не будут пользоваться самыми последними моделями при отсутствии самых последних игр (разве что у вас двойная загрузка с Windows), мы сосредоточимся на стоимости, производительности, поддержке и совместимости устройств. С точки зрения стоимости мы рассмотрим чуть менее новые модели, включив пару дешевых и пару более дорогих вариантов. Для оценки производительности мы проверим каждую видеокарту версией 3.0 теста Unigine. Это суровый тест 3D-производительности, включающий отрисовку миллионов полигонов в окружении внешней среды, динамического глобального освещения, объемных дождевых облаков и рассеяния света. Он выглядит лучше, чем любая родная для Linux игра, и проверяет и возможности устройства, и качество драйверов. Так как движок Unigine используется в нескольких известных играх, включая *Oil Rush*, результаты теста должны дать хорошее представление об успехах GPU на любых современных играх.



► В расчет берется не только скорость: важно и качество рендеринга. У Radeon HD6850 (справа) есть проблема с текстурами в *Bioshock*. У Nvidia GTX570 (слева) – нет.

Однако мы хотели протестировать видеокарты и на тех играх, в которые вам хочется поиграть сейчас: например, с последней версией *Alien Arena*, а также с коммерческими инди-играми, вроде *World of Goo*. Что еще более важно, мы протестировали их и с некоторыми играми от Steam, запущенными на Wine. Steam – это игровой портал для Windows, который стал лучшим вариантом покупки и установки новых игр для этой операционной системы. Есть несколько серьезных свидетельств в пользу того, что до конца этого года Steam придет в мир Linux. В таком случае его производительность на Wine должна дать нам некоторое представление о том, как те или иные игры под Steam будут работать в Linux.

### Устройства

Мы протестировали пять различных устройств. Первые две карты были интегрированными, т.е. являлись частью процессора, а не отдельными картами, вставляющимися в слот на материнской плате. Эти процессоры и графические карты часто называют ускоренными обрабатывающими устройствами (APU). Мы начали с Sandy Bridge APU от Intel на процессоре i5-2500K с тактовой частотой 3,30 ГГц, и так как Intel серьезно относится к разработке драйверов для Linux, мы ожи-

дали от него хороших результатов. Другой APU имел гораздо лучшую спецификацию на бумаге, и это A8-3850 APU (также известный как AMD Fusion). Это пресловутое ядро PlayStation 4, и хотя графический процессор нашей модели, похоже, не такой мощный, как будущий наследник Sony, его вычислительную мощность можно объединить с внешней видеокартой Radeon с помощью параметра CrossFire, включаемого в BIOS. Это AMD Radeon HD 6550D, и мы использовали ее с отведенными 512 МБ ОЗУ. Остальные видеокарты были внешними и подключались в свободные слоты PCIe на материнской плате. При этом нужно убедиться, что



у вас есть два свободных слота, так как видеокарта часто занимает соседний слот для дополнительного охлаждения, и что ваш источник питания достаточно мощный. Мы пользовались источником питания мощностью 600 Вт с двумя отдельными «рельсами» 12 В для питания видеокарты. Наши карты требовали дополнительного питания: один 6-контактный разъем или два для самой энергоемкой карты – Nvidia. Модели, которые мы проверяли – дешевая AMD Radeon HD 6670 (одна из карт, предназначенная для работы с A8-3850 APU) и более мощные AMD Radeon HD6850 и Nvidia GTX570, и тестирование выполнялось и с открытыми, и с закрытыми драйверами.

## Тест: оценка видеокарт

Результаты Sandy Bridge были неоднозначны. Тест *Mesa 8.0.2* от Unigine еле запустился, а значит, во многие современные игры поиграть не получится. С *Alien Arena* повезло больше: мы получили комфортные 60 кадров в секунду, но пришли к выводу, что для игр понадобится проприетарный драйвер. Первым GPU от Radeon, который мы протестировали, был встроенный HD 6550D с открытым драйвером Gallium версии 0.4. Производительность рабочего стола была хорошей, и ускоренный Unity в Ubuntu работал безо всяких проблем (как и на Intel). Ничуть не меньше нас впечатлило то, что тест «Небеса [Heaven]» дал лучшие результаты, чем у Sandy Bridge, и тем более – чем на нашей старой Nvidia 7600GTS, но рендеринг тем не менее не работал. Мы наблюдали, как силуэты, вместо цветных текстур, ползут по экрану с частотой 7 кадров в секунду. Поэтому для следующего теста мы воспользовались проприетарными драйверами Catalyst, которые установили вручную.

Нашим следующим тестом была *Alien Arena*, которая запускалась с на удивление низкой частотой в 25 кадров в секунду – более чем достаточно для счастья офисных приложений, но очень далеко от Sandy Bridge. Но в тесте «Небеса» проприетарные драйверы выполнили рендеринг кор-

## Тестирование GPU

Все устройства, кроме Intel, мы тестировали в системе с процессором AMD A8-3850 3,6 ГГц, 4 ГБ оперативной памяти и стандартной 64-битной версией Ubuntu 12.04. Чтобы избежать конфликтов с графическим ускорением, мы пользовались Unity 2D.

ректно, получив 10,3 очка. Может показаться, что это мало, но если учесть, что чипсет встроенный и сам тест не оптимизирован под игры, то результат неплох. Мы попробовали запустить один и тот же тест в Unity 3D и Unity 2D, чтобы посмотреть, есть ли разница, когда рабочий стол использует OpenGL, и не нашли ее – это доказывает, что в недавно выпущенном Unity 5.12 проблемы с производительностью OpenGL были исправлены. Производительность немного поднялась, когда мы тестировали Radeon HD 6670 1024 МБ. *Alien Arena* теперь запускалась с частотой 55 кадров в секунду, а тест «Небеса» дал нам 25,3 кадра в секунду с минимальным значением в 11 и макси-

**Intel Sandy Bridge i5-2500K, цена £145**  
**AMD Radeon HD 6550D, цена £80**  
 (с процессором A8-3850)  
**AMD Radeon HD 6670 1 ГБ, цена £50**  
**AMD Radeon HD6850 1 ГБ, цена £90**  
**Nvidia GTX570 1,25 ГБ, цена £180**

были установлены, и она мигмом продемонстрировала гораздо лучшие результаты с тестом «небеса», набрав 46,2 кадра в секунду с минимальными 15 и максимальными 78,8.

Восхищенные этим результатом, мы решили исследовать несколько других тестов, первым делом – с родной (и старой) версией *Darwinia*. Она запустилась с исключительными 160–250 кадрами в секунду, а значит, у этой карты не будет никаких проблем со старыми играми. Но когда мы попробовали приняться за Steam, появились проблемы. Например, чтобы заработал *Bioshock*, сначала пришлось выйти из Unity 3D. Но даже когда игра запустилась, графика отрисовывалась неправильно. С играми от Source все было лучше – стресс-тесты, и *Half Life 2*, и *Lost Coast*, показали хорошие результаты: последняя работала с частотой 47,91 кадров в секунду при впечатляющем качестве рендеринга.

И, наконец, самая дорогая видеокарта в нашей подборке – GTX570 от Nvidia с ОЗУ 1,25 ГБ. Сначала мы протестировали ее с открытыми драйверами nouveau, но игры *Darwinia* и Steam запустить не удалось, и мы поняли, что если вы намереваетесь потратить значительную сумму денег на видеокарту, вам понадобятся проприетарные драйверы, получше.

В использовании проприетарных драйверов Nvidia есть и другие преимущества. Например, вместе с драйверами можно установить очень мощную утилиту настройки. Для поддержки нескольких экранов можно включить TwinView, который всегда казался нам более стабильным, чем Xinerama, и переключаться между различными разрешениями для каждого экрана без перезапуска. Драйверы Catalyst тоже умеют это делать, но с Nvidia также можно разогнать свою видеокарту и отслеживать ее температуру. Утилита также довольно удобна при устранении неполадок, и с помощью утилиты Настройки [Settings] мы смогли загрузить с наших экранов данные EDID и заставить другие экраны тоже их использовать.

С проприетарными драйверами безусловным победителем стала GTX570. Она выдала серьезный результат на тесте «Небеса» (66,6 кадров/с) и безупречно запустила Stream под *Wine*, поэтому видеокарты от Nvidia потянут игры Steam с большим отрывом. Стоит ли это лишнего затрат, зависит от того, насколько игры для вас важны.

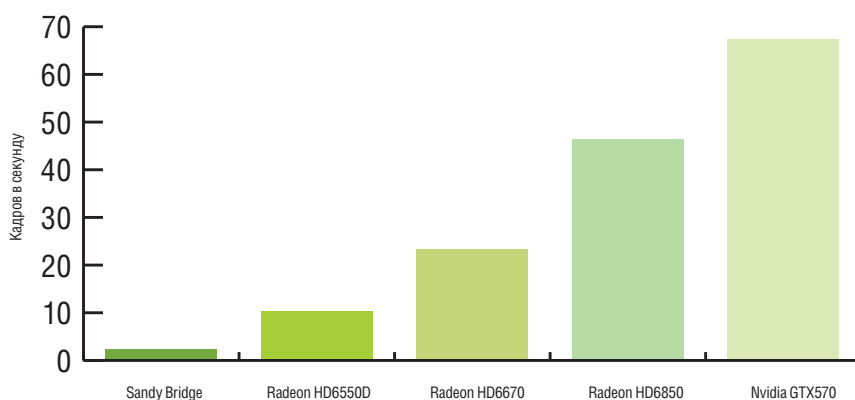
## «Идеальный вариант для Linux-машины под игры и фильмы.»

мальным в 46 кадров в секунду. Это прекрасный результат для бюджетной карты, и если вы предпочтете версию с пассивным охлаждением, она может стать идеальным вариантом для Linux-машины, используемого под игры и просмотр фильмов.

## Тест: мощные видеокарты

Нам остались две самых мощных видеокарты – Radeon HD6850 1024 МБ и Nvidia GTX570. Мы начали с Radeon, потому что драйверы для нее уже

## Тест «Небеса»: Сравнение производительности



## Твердотельные жесткие диски

Смените свой диск на такой, которым управляют только электроны.

**Х**отя процессоры, видеокарты, оперативная память и сетевые подключения с годами становятся быстрее, жесткие диски, кажется, ушли недалеко. В них все еще используются механические компоненты, и по сути они остаются одними из самых тяжелых, медленных, ненадежных и энергоемких компонентов компьютера. Но твердотельные жесткие диски (Solid State Drive – SSD) все меняют, и это одна из самых потрясающих разработок среди устройств для персонального компьютера за последние пять лет.

В данном разделе мы рассмотрим эти чудесные устройства. Мы сравним два диска, которые у нас есть, а также ответим на вечные вопросы пользователей об SSD: «Стоят ли они своих денег?», «Сколько они прослужат?» и «Как получить лучшие результаты от моего жесткого диска?»

### Стоят ли они затрат?

В обычных жестких дисках есть вращающийся диск, покрытый магнитным материалом. На этот магнитный материал воздействуют головки чтения/записи при перемещении над диском, и именно он хранит данные. А вот в SSD движущихся

### «Дисков такого же объема, как традиционные, купить нельзя.»

частей нет. Вместо этого они состоят из миллионов маленьких транзисторов (массива плавающих логических вентилялей), каждый из которых хранит один бит информации. Так как в них нет движущихся частей, они тише, легче, более энергоэффективны, обладают большим сроком службы и скоростью работы.

Это, безусловно, здорово, если вы хотите пользоваться ими на ноутбуке, где очень важны размеры, энергопотребление и шумность. Увеличенная

скорость также окажет огромное влияние на запуск приложений (и любые другие операции, требующие частых обращений к диску), и ваш компьютер станет работать заметно быстрее.

Все эти преимущества прекрасны, но у SSD есть и недостатки, и о них следует подумать, прежде чем покупать диск. Главный – дисков такого же объема, как традиционные, купить нельзя, и они гораздо дороже. Например, диск Crucial M4 128 ГБ, который у нас был, стоит £107,99, а за £79,99 можно купить обычный жесткий диск объемом 2 ТБ. Если вам нужно много дискового пространства или ваш бюджет ограничен, SSD, возможно, не для вас.

Таким образом, ответ на вопрос «Стоит ли они затрат?» – «В зависимости от того, как вы используете компьютер».

### Срок службы

Два общих сомнения, касающиеся SSD – в том, долго ли они служат и сохраняют ли они производительность в течение всего срока службы. Эти сомнения небезосновательны. Транзисторы в SSD прослужат около 10 лет или в течение 10 000 операций записи (что наступит быстрее), поэтому срок их службы ограничен. Более того, в некоторых ранних моделях плохие прошивки могли значительно снизить производительность с течением времени.

На современных дисках с современными операционной и файловой системами значимость этих проблем существенно снижается благодаря так называемой TRIM [команда интерфейса ATA, позволяющая операционной системе уведомить твердотельный накопитель о том, какие блоки данных больше не используются и могут применяться накопителем для подготовки к записи, – прим. пер.]. Она помогает прошивке диска управлять выделением блоков данных, гарантируя, что запись на каждый транзистор будет

производиться минимальное число раз без снижения производительности.

Насколько большой эффект оказывает TRIM? В одной из самых авторитетных статей по этой теме Ананд Лал Шимпи [Anand Lal Shimpi] обнаружил, что без TRIM производительность старого диска составляла всего 52% производительности нового; с TRIM этот показатель составлял 98%. Поэтому TRIM стоит включить.

Как же включить TRIM? Сначала нужно убедиться, что диск ее поддерживает. Если вы купили диск в последние пару лет, это почти наверняка так, но на дисках постарше это стоит проверить.

Сделать это можно командой `hdparm`:  
`hdparm -l /dev/<ssd> | grep "TRIM supported"`  
заменяв `<ssd>` на имя устройства своего SSD. Если команда что-то вернет, то вы готовы включить TRIM в операционной системе.

Для этого нужно отформатировать свои разделы с `ext4` или `btrfs`. TRIM поддерживают только эти две файловые системы. Здесь в редакции **LXF** мы пользуемся `ext4`, так как в `btrfs` все еще нет стабильной утилиты восстановления, что снижает шансы на восстановление после сбоя – и мы советуем вам тоже воспользоваться `ext4`.

### Меняем опции монтирования

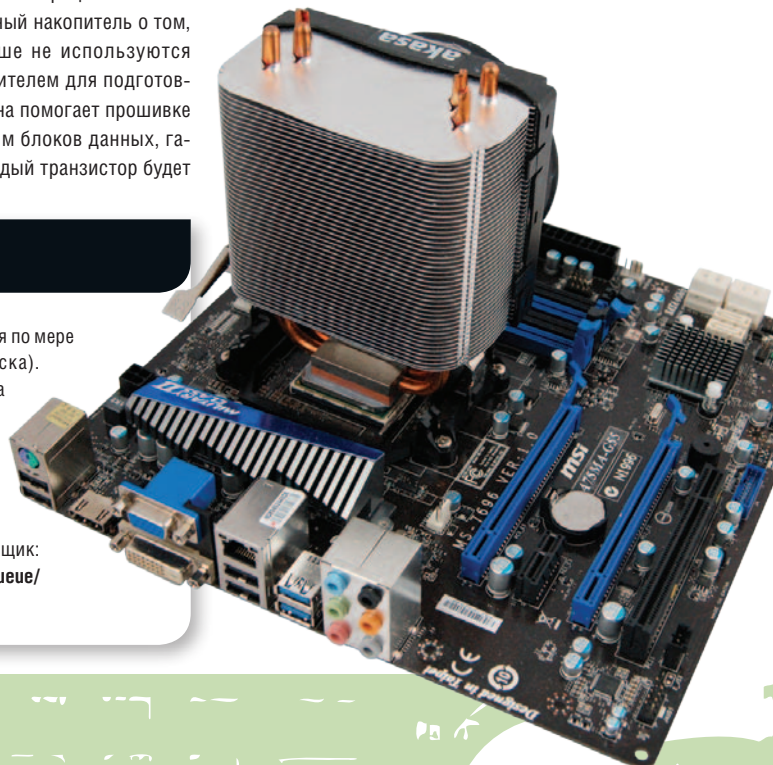
После этого измените опции монтирования файловой системы, так как в них TRIM не включена по умолчанию. Для этого нужно отредактировать файл `/etc/fstab`. Перед внесением любых изменений обязательно сохраните резервную копию, потому что если вы допустите ошибку, система может перестать загружаться.

### Настройки производительности

Для тех, кто помешан на скорости, есть пара настроек, которые можно применить для увеличения скорости диска. Одна из них включает правильное выравнивание разделов, но о ней уже рассказал Нейл Ботвик в **LXF160**, и здесь мы не будем углубляться в детали.

Другая – изменить настройки планировщика диска. Он определяет порядок выделения блоков диска: на вращающихся жестких дисках планировщик использовался для назначения приоритетов одним блокам диска по отношению к другим (вре-

мя доступа к блоку различается по мере его близости к центру диска). А в SSD времена доступа к блокам почти одинаковы, и такая дискриминация может значительно снизить производительность, поэтому стоит переключиться на «холостой [noop]» планировщик:  
`echo noop > /sys/block/<ssd>/queue/scheduler.`



```
cp /etc/fstab /etc/fstab.bk
```

Если что-то испортится, вы сможете загрузиться с Live CD, восстановить файл из копии, перезагрузиться, и система снова станет работоспособна.

Создав резервную копию, измените во всех строках файла, относящихся к разделам на SSD, часть со словом 'defaults'. Добавьте к ней 'discard', и теперь строки будут выглядеть так:

```
/dev/sda1 / ext4 defaults,discard 0 1
```

Готово. Сохраните файл, перезагрузитесь, и поддержка TRIM для диска будет включена. Это самая важная настройка для вашего твердотельного диска.

## Увеличение срока службы

Есть и другие настройки, способные увеличить срок службы диска.

Простейшая из них – добавить к опциям монтирования **noatime**, точно так же, как ранее мы добавили опцию **discard**. В файловых системах Linux обычно хранятся время последнего чтения и последнего изменения файла. С опцией **noatime** будет сохраняться только время последнего изменения, что снизит число операций записи и увеличит срок службы диска. Однако предупреждаем: старые программы, вроде *Mutt*, не будут работать правильно с этой опцией, поэтому сначала проверьте совместимость приложений.

Срок службы диска можно также увеличить, тщательно продумав, какие разделы на нем раз-

местить. Например, если наряду с SSD в системе есть обычный жесткий диск, стоит разместить на SSD файловые системы, доступ к которым осуществляется реже, такие как `/` и `/home`, а `/var`, `/tmp` и `swap` разместить на обычном диске. Если такой вариант вам не подходит, можно настроить и другие параметры, ради снижения частоты записи в эти каталоги. Например, можно увеличить уровень серьезности записываемых в лог сообщений, изменив файл `/etc/rsyslog.conf` (подробности см. через `man rsyslog.conf`), или уменьшить «активность подкачки» системы, чтобы она как можно реже пользовалась файлом подкачки. Для этого выполните следующую команду:

```
echo 1 > /proc/sys/vm/swappiness
```

## Наши тестовые диски

Большинство SSD используют примерно одинаковую технологию хранения данных. На производительность больше влияет контроллер и прошивка – устройство решает, когда и куда записать данные. Плохой контроллер может замедлить диск, особенно со временем, и привести к изменению производительности при операциях записи разного размера (например, 4К и 9К).

Два наших тестовых диска продемонстрировали два конкурирующих контроллера. В Crucial M4 используется контроллер Marvell, а в Intel 330 – Sandforce. Эти же контроллеры используются и на множестве других дисков, так что наши результаты помогут вам выбрать диск для покупки, даже если он будет другим.

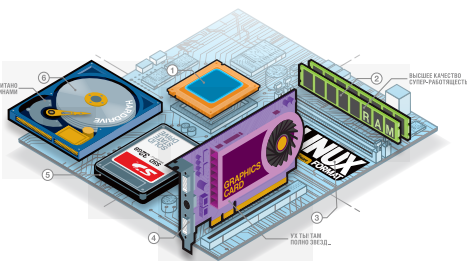


Мы протестировали диски с помощью тестов Postmark, Compile Bench и Kernel Unpacking из набора тестов Phoronix Test Suite, чтобы рассмотреть их производительность в реальных ситуациях. Все тесты проводились в Ubuntu 12.04 с файловой системой ext4 и опцией **discard** в `/etc/fstab`.

Пожалуй, интереснее всех тест Compile Bench, так как он пытается имитировать операции, которые старят файловую систему – самый верный сценарий для нагрузки контроллера. На этих тестах диск Intel с контроллером Sandforce показал гораздо лучшие результаты.

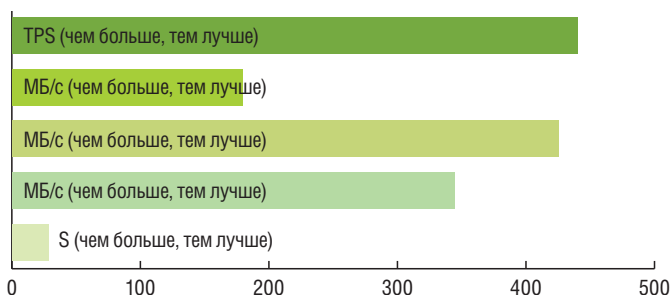
Однако диск Crucial был гораздо быстрее при работе со множеством маленьких файлов в тесте PostMark и немного быстрее при распаковке ядра.

Оба диска стоят одинаково; через Интернет их можно купить от £84 и выше. **LXF**



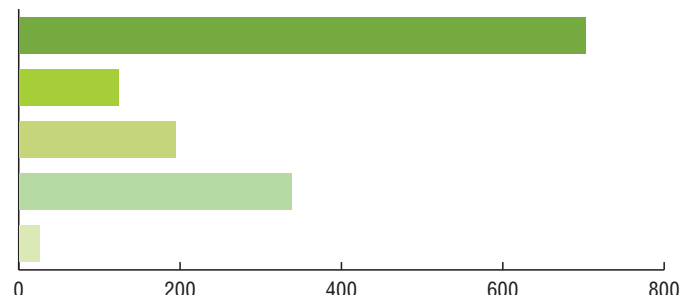
## Результаты

Intel 330



- Postmark
- Compile Bench Initial Create
- Compile Bench Compile
- Compile Bench Read Compiled Tree
- Unpacking the Linux Kernel

Crucial M4



- Postmark
- Compile Bench Initial Create
- Compile Bench Compile
- Compile Bench Read Compiled Tree
- Unpacking the Linux Kernel

# Большой босс

Джонатан Робертс встретился с Джейн Силбер, исполнительным директором Canonical, чтобы узнать о будущем Ubuntu.



Canonical активно продвигает Ubuntu во всех сферах. Для настольных ПК — пожалуйста, новый рабочий стол Unity; для серверов — поддержка процессоров ARM и облачные платформы; есть даже решения для мобильных телефонов и телевизоров. Следить за всем этим вместе и по отдельности, видеть перспективы развития — дело непростое... Мы встретились с Джейн Силбер (Jane Silber) — исполнительным директором Canonical, чтобы узнать, как ей это удается.

**LXF:** Давайте начнем с настольной Ubuntu.

Главное достижение здесь — это, конечно, Unity, и во многом благодаря тестированию пользователями. Не могли бы вы рассказать, что именно вас к этому подвигло и где вы нашли людей? Ведь для свободного ПО это опыт в общем-то уникальный.

**Джейн Силбер:** Да, вы правы, думаю, что мы в этой области действительно стали пионерами, открывшими новые горизонты в мире свободного ПО. Пользовательское тестирование изобрели не мы, но мы верим в концепцию дизайна, ориентированного на пользователя. Ведь мы поставили цель существенно поднять планку удобства использования в Ubuntu и свободном ПО в целом, сделав интересы конечных потребителей и результаты пользовательского тестирования правильным пунктом разработки дизайна.

Так что Unity с самого начала прошла множество таких тестов. От обсуждения ключевых концептов и их представления на бумаге

## ПРО ВКЛАД ПОЛЬЗОВАТЕЛЕЙ

«В целом, разработка дизайна — максимально открытый процесс.»

и вплоть до использования готового продукта — и в процессе разработки, и далее. Даже выпустив продукт, мы все равно продолжаем его тестировать и делать контрольные проверки, проводя некоторые тесты заново.

**LXF:** Ubuntu существует как свободный и открытый проект; а результаты тестирования можно найти в открытом доступе?

**ДжС:** Да, мы публикуем все результаты тестирований и будем это делать и впредь. Значительную их часть вы найдете, посетив блог [design.canonical.com](http://design.canonical.com), так что если вас это интересует, начинайте оттуда. У нас практически все публично. Есть некоторые вещи, которые мы не раскрываем на ранних этапах разработки, в интересах конкуренции или клиентов, но в целом разработка дизайна — максимально открытый процесс, равно как и пользовательские тесты. Также мы много делимся опытом с другими проектами, касательно



организации подобных работ. На UDS, Ubuntu Developer Summit, мы будем устраивать мастер-классы, куда человек может прийти и сказать: «Это мой любимый открытый проект, я его активно использую, но я не разработчик. Как я могу помочь его развитию?». И там же он сможет и чему-то научиться в плане дизайна, и принять участие в тестировании.

**LXF:** Еще один козырь настольной Ubuntu — это Software Centre, который действительно выделяет его среди других дистрибутивов. Как вы привлекаете сторонних производителей ПО, и пользуются ли спросом платные приложения?

**ДжС:** Все идет замечательно. У нас пока не набралось 100 000 приложений, как у некоторых других платформ, но магазин активно растет и действительно привлекает разработчиков. К примеру, такой гигант, как EA, выпускает свои игры для Ubuntu, и буквально на прошлой неделе у нас было совместное мероприятие с

Humble Bundle по запуску и раскрутке нового проекта.

Точные цифры продаж на текущий момент я назвать не берусь, но за первые 72 часа после релиза было зафиксировано 10 000 заказов из Software Centre. И это наш рекорд для нового релиза. Причем список релизов постоянно пополняется. Но нам предстоит большая работа. Почивать на лаврах пока еще рановато. Нужно предоставить людям более связный SDK, чтобы они могли писать под эту платформу.

**LXF:** Для Linux и свободного ПО это особая трудность, так как цели нередко смещаются.

**ДжС:** Да, и тут богатые тоже плачут. Вот у нас есть всяческие инструментари, и их гораздо больше, чем у других платформ. Это, конечно, здорово. Но иногда люди просто не знают, за что хвататься. И здесь мы провели кое-какую работу, не ограничивая их выбор, но задавая некое единое направление, чтобы раз-

работчикам было удобнее. Скажем, решение типа Quickly, или <http://developer.ubuntu.com>.

**LXF:** А что касается сервера, насколько я знаю, период поддержки был недавно увеличен до 5 лет...

**ДжС:** Наши долгосрочные (LTS) релизы поддерживались три года для настольных систем и пять для серверных, а сейчас мы ввели два изменения. Первое — продление поддержки настольных до 5 лет, в интересах корпоративных клиентов. А второе — обеспечение доступности LTS-версий на новом оборудовании, чтобы пользователи могли модернизировать свой парк, не опасаясь за стабильность корпоративной платформы. Поэтому ядро версии 12.10 будет совместимо с 12.04, а ядро 13.04 — с 12.04. И если вашему оборудованию нужна именно эта версия, она будет поддерживаться.

**LXF:** В Red Hat недавно объявили о продлении цикла поддержки серверных систем, вроде бы

**с 7 до 10 лет; не ощущает ли Ubuntu давления из необходимости этому соответствовать?**

**ДжС:** Пока что нет. Я полагаю, здесь все дело в разной пользовательской аудитории Red Hat и Ubuntu. И, что интересно, если говорить о давлении, то оно как раз противоположное.

Одной из тенденций облачного мира является быстротечность: невозможно представить, что через 10 лет вы будете использовать тот же инструментарий, что и сегодня. Очевидно, сейчас более актуальна задача обновления оборудования с поддержкой базовых ОС и обновление ПО для облачных технологий. То есть пользователи хотят иметь, например, новый *OpenStack* на 12.04 LTS, и это мы тоже обязаны ввести. Через полгода появятся новейший *OpenStack*. В Ubuntu 12.04 пока есть *OpenStack Essex*, но с выходом следующей версии *Folsom* все захотят, чтобы она была доступна не только на 12.10, но и на 12.04, у которой долговременная поддержка.

**LXF:** На этой неделе вы были на Computex, и самым громким заявлением стало то, что вышла демо-версия Ubuntu на платформе ARM. Есть ли уже примеры конкретных внедрений? И откуда такая потребность?

**ДжС:** Потребность сформирована появлением масштабируемых, энергоэффективных и недорогих устройств на базе этого процессора. Интерес пользователей, конечно, тоже есть, но пока об этом рано говорить, ведь такое оборудование пока не представлено на рынке. Готовых примеров еще не было. Но публика их жаждет, и уже появляются первые образцы. Люди из Calxeda представили кое-что на UDS неделю назад. Мы представили ARM-сервер совместно с тайваньской компанией MiTAC в Тайпее, и HP также объявил о новом проекте Moonshot, который станет их ARM-сервером. Так что такое оборудование уже существует, по большей части в стадии разработки и тестирования, где рассчитывается уровень нагрузки, необходимые оптимизации, но это более чем реально, хоть мы еще и в самом начале.

**LXF:** Можете сказать, сколько людей сейчас занято ARM?

**ДжС:** Точно сказать сложно, ведь в этом так или иначе задействована почти вся компания, это наше общее детище. Некоторые из разработчиков серверной версии Ubuntu, вот Робби Уильямсон [Robby Williamson], трудятся над тем, чтобы Ubuntu работала и на оборудовании Intel, и на ARM... то есть отдельной команды ARM нет, мы предпочитаем рассматривать это как еще одну архитектуру, поддержку которой необходимо обеспечить. Так что это и часть нашей команды QA, автоматизированные сборки, автоматизированные тесты и т. д.

**LXF:** Раз примеров готовых ARM еще нет, как вы сказали, может быть, вы расскажете нам о каких-нибудь других — серверной или

настольной Ubuntu, из тех, что вас особенно порадовали?

**ДжС:** О, сколько угодно! Некоторые из них уже в облаках. Например, результаты Mercadolibre, южноамериканского облачного провайдера, создавшего хранилище с 1000 узлов Ubuntu — то есть приличных размеров облако *Open Stack* на основе Ubuntu. По части настольной версии, мы недавно сделали довольно обширное внедрение в Южной

Африке, в сфере образования, установив 50000 систем в 1600 школах, что объединит 2 миллиона учащихся. Это общий ресурс. Он обеспечивает взаимосвязь и поддержку управляемой из облака настольной системы на территории всей провинции Гаутенг. Мы сделали это совместно с южноафриканским партнером, и на мой взгляд, это первый подобный проект такого масштаба в Южном полушарии.

**ДжС:** Sanopical тратит на это очень много усилий. И, к счастью, у нас уже есть хороший задел. Многие из того, что мы делаем на коммерческой основе, может быть полезно Ubuntu, и силами сообщества нельзя укрепить деловые партнерства и связи. Мы тесно сотрудничали в этом направлении с Тайванем и Китаем, и у нас сложились хорошие, прочные отношения со всеми главными игроками. Так что мы очень стараемся способствовать продвижению на рынок и Ubuntu TV, и Ubuntu для Android. Наш маркетинговый план развития рассчитан исключительно на продавцов оборудования. Мы не собираемся сами его производить и продавать. Поэтому мы надеемся попасть на рынок за счет партнерских связей; но поскольку сам продукт нам не принадлежит, о конкретных сроках говорить трудно.

**LXF:** Ну, в Sanopical уже делали подобные попытки — был опыт сотрудничества с Dell на рынке настольных компьютеров, и Ubuntu Netbook Remix для нетбуков; и нельзя сказать, что все вышло, как задумывалось. Вы учли этот опыт в работе с нынешними партнерами?

**ДжС:** Думаю, что учли. Мы научились многому. В самой индустрии многое изменилось, изменилась динамика. Я думаю, идея нетбука была неплохой, но вот оборудование подкачало, потому им и трудно было найти свою нишу на рынке. Что здесь главное? Дешевизна? Максимум компактности? И то, и другое требует совершенно разных технических решений и программного наполнения. Думаю, главное, что мы вынесли из опыта с Netbook Remix, это ценность удобства

## ПРО ПРОГРЕСС

### «Наблюдается активное развитие — и на сервере, и на настольных системах.»

Африке, в сфере образования, установив 50000 систем в 1600 школах, что объединит 2 миллиона учащихся. Это общий ресурс. Он обеспечивает взаимосвязь и поддержку управляемой из облака настольной системы на территории всей провинции Гаутенг. Мы сделали это совместно с южноафриканским партнером, и на мой взгляд, это первый подобный проект такого масштаба в Южном полушарии.

То есть по-прежнему наблюдается активное развитие, и на сервере — главным образом в облаке, операциях с большими объемами данных и масштабированием web-приложений — и в настольных системах: в корпоративной, управленческой и частной сфере.

**LXF:** Похоже, все идет к тому, что Ubuntu скоро будет повсюду: и в телевизорах, и в мобильных... Думаю, никто не сомневается в том, что вы справитесь с ПО, но, как и для большинства открытых проектов, самое сложное — договориться с изготовителями оборудования и довести ПО до поль-



использования программы; именно тогда мы формировали Unity и стали контролировать опыт пользователей... вернее, не столько контролировать, сколько определять свое видение на основе этого опыта, и мы обрели самоуверенность. И по-моему, это привело к большому удобству и целой серии продуктов Ubuntu.

**LXF:** Кстати о самоуверенности, насчет Unity и сообщества в целом — ведь случился раскол. Не кажется ли вам, что события с Gnome 3, Gnome Shell и Unity отчасти стали итогом вашего стремления подогнать интерфейс к своему вкусу и желания управлять пользователями? Не примите за критику, просто у вас была концепция, которую вы хотели воплотить...

**ДжС:** Я думаю, вы совершенно правы. Все действительно сложилось не в нашу пользу — в сообществе произошел раскол, и предотвратить его мы не сумели. По-моему, сообществу проще развиваться при четкой концепции развития, и именно так работает Ubuntu. Наше сообщество — эффективное, целостное и целеустремленное, но когда целей становится несколько, идти одной дорогой становится трудно, и однажды мы осознали, что на индивидуальном уровне цели разные, отсюда вся история с Gnome 3 и нашей концепцией Ubuntu.

**LXF:** А есть ли надежда, что через несколько лет другие дистрибутивы переймут опыт Ubuntu на настольных системах и напряжение спадет?

**ДжС:** Не знаю, последует ли какая-то отдача в плане кода — но для нас это всегда был не только код: это и документация, и опека новых пользователей, и прочее; так что если со временем другие проекты смогут дать нам что-то в плане дизайна и удобства — это уже будет большой вклад.

**LXF:** А в управлении компанией лично вы пользуетесь только Ubuntu и открытым ПО, или этого хватает не всегда?

**ДжС:** Вся компания полностью работает на Ubuntu. Может, скажу немного не в тему, но мы недавно переехали. В старом офисе у входа стоял ручной сканер для Windows, и это всех бесило — нужно было иметь лицензию Windows! А здесь его совсем нет, так что теперь у нас есть лицензионная Windows как подопытный образец, а не инородное тело. А наша внутренняя инфраструктура — это преимущественно открытый код. В бухгалтерии мы используем *OpenERP*. Мы применяем и некоторые проприетарные облачные сервисы: [Salesforce.com](http://Salesforce.com), GoogleApps для Calendar Sync и т.д. Мы не против коммерческого ПО.

Но сами предпочитаем открытый код. У нас есть свое внутреннее облако *OpenStack*, где каждый может получить доступ к любым ресурсам.

**LXF:** Ух ты! То есть, желая написать программу, можно просто взять оттуда все, что нужно?

**ДжС:** Да, и благодаря этому компания получила много инноваций, а сотрудники обрели подвижность. Допустим, вам пришла идея сделать какое-нибудь автоматизированное тестирование — и сразу можно ее опробовать, а если получилось, сразу же и внедрить.

**LXF:** А по финансовой деятельности Canonical, как идут дела с повышением прибыльности?

**ДжС:** Все достаточно неплохо. Прибыль стабильно растет. Официальных цифр мы не называем, мы же частная компания. Но развитие продолжается, растет интерес к Ubuntu со стороны корпораций, особенно в новых сферах управления большими нагрузками и в облаке, и в настольных системах, и на мобильных устройствах, в связи с увеличением их числа и интеграцией платформ. Грань между настольными и мобильными системами постепенно стирается, и Ubuntu как платформа может изящно лавировать между теми и другими. **LXF**

Что на самом деле творится в этом черном ящике с секретами внутри вашего ПК? И почему он так важен для нас, линуксоидов? Майк Сондерс раскроет тайну...

# Анатомия жесткого диска

**В**от вам мысль: какой компонент в вашем компьютере самый ценный? Ценный именно для вас, а не с точки зрения рыночной стоимости. Сам компьютер со всем оборудованием вряд ли является семейной реликвией, если только ваш дедушка не успел перед кончиной приобрести звуковую карту AdLib и передать вам по наследству (а вы сейчас от души наслаждаетесь, пытаетесь заставить ее работать в современной машине).

Но для большинства из нас самая ценная часть – определенно жесткий диск. Если прикажет долго жить CPU или нарочается видеокарта, мы просто купим другие. А вот если с нашим суровым миром распрощается жесткий диск, прихватив

с собой все ваши жизненно важные файлы (а свежих копий у вас нет), этого уже не исправить никакими деньгами. И тем не менее, несмотря на всю свою важность, скромняга жесткий диск недополучает заслуженного им внимания.

Мы все ведемся на яркие вещи – новые дистрибутивы и среды рабочего стола, но есть масса полезной информации, которую стоит узнать об этих устройствах. Например, существует много страте-

гий деления жесткого диска на разделы, влияющих на безопасность и производительность. Есть разные типы файловых систем, которые вы можете использовать, и разные хитрости, пригодные для восстановления данных в случае, если что-то пойдет не так.

Новые технологии вроде SSD меняют роль жестких дисков. Если вы случайно удалили файл, велика вероятность, что вы сможете его восстановить с помощью всяких хитроумных инстру-

ментов. Так что жесткий диск – это вовсе не прозаическая коробка с байтами, зарытая где-то в недрах вашего компьютера, а целый технологический мир с уймой возможностей индивидуальной

настройки. Цель нашей статьи – научить вас всему, что стоит узнать о жестких дисках, и даже немного большему. Мы также включили несколько советов, которые вы можете вырезать и повесить на стену рядом со своим компьютером на случай ЧП.

Исключительно безопасности ради (и нашей, и вашей!), небольшое предупреждение: наш урок посвящен изменениям, вносимым в структуру данных, которые хранятся на жестком диске. Настоятельно рекомендуем вам попробовать самостоятельно команды и опции, ведь это самый лучший способ обучения... но только на тестовой машине (или в VirtualBox). Не экспериментируйте на своем основном компьютере, если не хотите рисковать данными!

**«Новые технологии вроде SSD меняют роль жестких дисков.»**



# Что такое разделы?

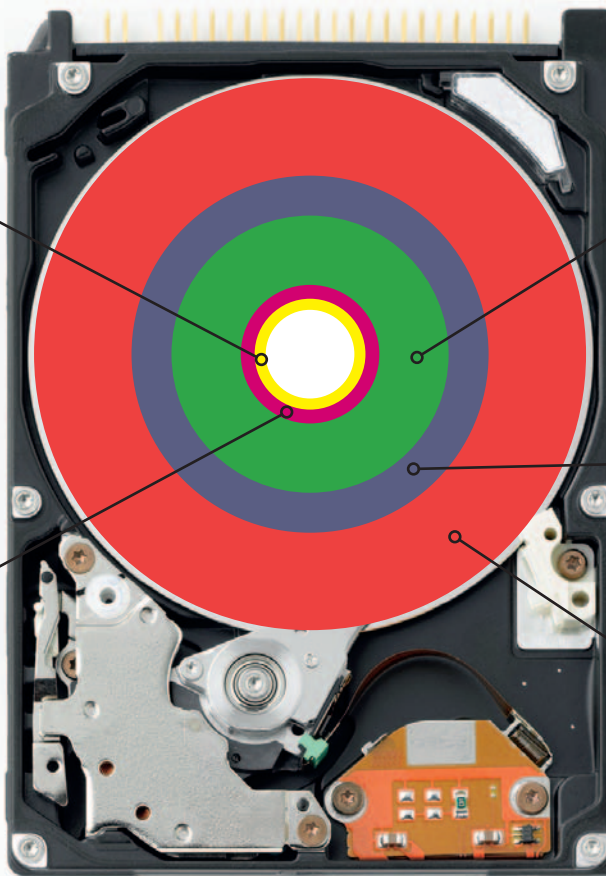
От пустого жесткого диска проку мало; прежде чем он сможет начать хранить данные, на нем должна появиться некая структура. На низшем уровне жесткие диски состоят из секторов, то есть очень маленьких блоков данных, хранимых в заданных участках диска. На диске могут находиться многие миллионы секторов, и они организованы в логические группы на разных уровнях.

На самом нижнем уровне это разделы (файловую систему мы рассмотрим позднее). Раздел

по сути являет собой собрание секторов, отведенных под разные задачи хранения данных.

Большинство компьютеров от наиболее известных изготовителей идут только с Windows (что печально), поэтому на их жестких дисках имеется только один большой раздел, занимающий практически весь диск. При загрузке Windows он именуется диском **C:**. На некоторых машинах имеется второй, «спасательный» раздел – он содержит резервную копию ОС на случай, если придется ее переустановить.

Цель разделов – хранить области данных отдельно друг от друга. Например, при установке Linux на компьютер с Windows программа установки Linux, как правило, сжимает раздел Windows, чтобы выделить место для разделов Linux. В конечном итоге вы получаете диск с несколькими разделами, как показано на диаграмме. Windows знает, что нельзя залезать в разделы Linux, и наоборот. Размеры этих разделов разные в разных системах, в зависимости от того, сколько вы выделите под каждую ОС.



### MBR – Master Boot Record

По сути, это не раздел, а крошечная часть диска, содержащая первые части загрузочного кода вместе с таблицей разделов, которая показывает ОС, какие разделы имеются на диске.

### Swap-пространство

Используемый для виртуальной памяти, раздел swap [подкачки] не содержит файлов, как обычный раздел. Вместо этого ядро Linux использует его в качестве более крупного и более медленного банка ОЗУ для простаивающих задач. Неплохо задать его вдвое больше вашего ОЗУ, но не более 4 ГБ, кроме экстремальных случаев.

### Раздел root (/) в Linux

Он содержит операционную систему и файлы приложений – т. е., полный дистрибутив Linux. Его часто объединяют со следующим разделом.

### Раздел /home в Linux

Здесь хранятся личные файлы пользователей. На следующей странице мы рассмотрим все «за» и «против» отдельного раздела /home.

### Windows

Раздел NTFS, содержащий системные файлы ОС Windows.

## Вырежьте и сохраните: Экстренное деление на разделы

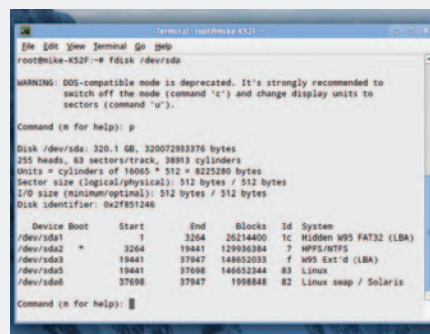
Программа *fdisk*, предназначенная для разбиения диска на разделы, во многом похожа на текстовый редактор *vi*: она простая, запускается из командной строки и имеется практически в каждом дистрибутиве. Запустите ее (от имени root), предоставив путь к диску (узлу устройства) таким образом:

```
fdisk /dev/sda
```

В обычной установке Linux */dev/sda* относится к первому жесткому диску, */dev/sdb* – ко второму, и т. д. Введите **p**, и вы увидите список разделов на диске, как на экранном снимке. Обратите внимание на столбцы Start и End, отображающие используемые секторы. Каждый раздел имеет свой номер, так что **sda1** – это первый раздел на первом диске, а **sdb3** – третий раздел на втором диске.

Чтобы удалить раздел, введите **D**, и у вас спросят номер раздела. Чтобы добавить новый раздел, введите **N**. У вас спросят, сделать ли его основным (не более 4-х) или расширенным; для простоты, выбирайте первый, если у вас есть такая возможность. Затем введите номер начального сектора (с учетом ранее упомянутого списка) и размер. В главном окне приглашения введите **P**, и вы увидите в списке новый раздел.

Но пока у него нет ID, поэтому введите **T** и номер раздела, а затем **Shift+L** для вывода списка имеющихся типов. Введите **83** для раздела Linux, **82** для раздела swap или **7** для раздела Windows (NTFS). Теперь введите **W**, чтобы записать изменения на диск, или **Q**, чтобы выйти, не сохранив изменений.



» Сессия *fdisk*, отображающая разделы Linux и Windows на жестком диске.

## Отдельный /home: да или нет?

Одним из самых сложных пунктов выбора, с которым вы столкнетесь при установке Linux и делении жесткого диска на разделы, таков: разместите ли вы директорию /home на отдельном разделе? Именно в ней проживают файлы пользователя – то есть личные документы и настройки учетной записи пользователя, а не файлы ОС, которые сидят в отдельных директориях. Некоторые дистрибутивы Linux рекомендуют учредить отдельный раздел, а некоторые по умолчанию все размещают в том же разделе. Как же поступить вам? Ответ зависит от того, как вы будете применять свой компьютер.

Если вы планируете попробовать много разных дистрибутивов и часто устанавливаете новые поверх старых, тогда имеет смысл создать отдельный раздел /home. Благодаря этому вы сможете делать с ОС все, что заблагорассудится – обновлять, понижать до более ранней версии или стереть и поставить случайно выбранный экзотический новый дистрибутив с Фарерских ост-



ровов. Каким бы дистрибутивом Linux вы ни пользовались, ваши личные файлы всегда будут там, в целостности и сохранности, на отдельной части диска. При известной осторожности можно даже иметь несколько дистрибутивов Linux на одном компьютере, и все они будут обращаться к одному и тому же разделу /home после загрузки.

Но почему мы говорим об осторожности? Вспомните о настройках и файлах конфигурации. Скомандовав, например, `ls -a` в своей домашней директории, вы увидите огромное количество скрытых файлов и директорий, имена которых начинаются с точек – там содержатся настройки программ. Если вы попытаетесь использовать одинаковые настройки в разных версиях программы, программа может запутаться.

Допустим, у вас на компьютере есть Дистрибутив А и Дистрибутив Б. Вы загружаете Дистрибутив А и запускаете *FooProgram 2.0* в первый раз, и она создает папку с настройками `.foooprogram/` в вашей домашней директории. Затем вы загружаете Дистрибутив Б при той же самой домашней директории и запускаете *FooProgram* – но на сей раз это будет версия 1.0. Она запутается из-за разницы в файлах настройки, что может привести к полной утрате или повреждению данных. Другая

потенциальная проблема с отдельной директорией /home – ограничение по размеру. Если вы поместите все в один раздел, то и ОС, и домашние директории будут иметь доступ к свободному месту. Если вы поместите /home на отдельный раздел и места не будет хватать, вы не сможете просто взять место из раздела ОС (но если вы используете *LVM*, Logical Volume Manager, как это предлагается на стадии установки во многих дистрибутивах, вы преодолеете эту проблему, поскольку он поддерживает изменение размера разделов).

Однако у подхода, поддерживающего отдельный раздел, есть и плюсы, особенно сейчас, когда все популярнее и доступнее становятся диски SSD (твердотельные накопители). Поскольку они невероятно быстры по сравнению с вращающимися жесткими дисками, вы можете поместить файлы ОС на SSD, чтобы обеспечить быструю работу системы и время запуска приложений, а /home – на традиционный жесткий диск (в конце концов, вас не слишком заботит, долго ли будут грузиться документы *LibreOffice* или фотографии).

Но для систем на обычных жестких дисках, если вы не собираетесь что ни день осваивать новый дистрибутив, мы рекомендуем подход «Размести все на одном разделе».

## Самые важные директории

Згляните в директорию `root (/)`, и вы увидите множество директорий, которые могут быть вам неизвестны. И хотя большинству пользователей крайне редко приходится обращаться к этим директориям, стоит все же знать, для чего они служат. Поясняем:

- » **/bin** Двоичные, или, более точно, исполняемые файлы, используемые базовой системой. Однако сюда не входят более крупные приложения рабочего стола, такие как *Firefox* (эти хранятся в `/usr`).
- » **/boot** Файлы, используемые для загрузки, например, ядро Linux.

- » **/dev** Узлы устройств [Device nodes]. Здесь находятся файлы, применяемые для доступа к оборудованию.
- » **/etc** Конфигурационные файлы системы (пользовательские настройки хранятся в директории `/home`).
- » **/media** Здесь часто монтируются съемные носители, например, USB-брелки.
- » **/mnt** Еще одно место для монтирования дисков (смотрите не запутайтесь), но обычно только жестких дисков или сетевых устройств.
- » **/opt** Опциональные пакеты приложений. В некоторых дистрибутивах именно здесь отводится место для программ-мастодонтов типа KDE или *LibreOffice*.
- » **/proc** Информация о процессах. Имеет реальную ценность только для администраторов, желающих отслеживать потребление ресурсов программой.
- » **/sbin** Крайне важные для работы системы исполняемые файлы, но запустить их может только суперпользователь (`root`).
- » **/usr** Содержит файлы, не имеющие критической важности, например, приложения. `/usr/lib`, внутри нее, содержит большинство библиотек, используемых приложениями.
- » **/var** Варьируемые файлы – т.е., данные, которые часто меняются, например, базы данных, почтовые ящики и журналы системы.

```
Terminal - mike@mike-K52F: ~
File Edit View Terminal Go Help
lrwxrwxrwx 1 root root 33 2011-09-17 19:26 initrd.img -> boot/initrd.img-2.6.38-11-generic
lrwxrwxrwx 1 root root 32 2011-09-17 18:46 initrd.img.old -> boot/initrd.img-2.6.38-8-generic
drwxr-xr-x 20 root root 4.0K 2011-12-11 15:23 lib
drwxr-xr-x 2 root root 4.0K 2011-09-24 16:34 loop
drwx----- 2 root root 16K 2011-09-17 18:37 lost+found
drwxr-xr-x 3 root root 4.0K 2012-06-12 21:50 media
drwxr-xr-x 2 root root 4.0K 2011-04-21 18:50 mnt
drwxr-xr-x 4 root root 4.0K 2012-01-04 17:16 opt
dr-xr-xr-x 158 root root 0 2012-06-12 23:50 proc
drwx----- 6 root root 4.0K 2012-03-19 22:11 root
drwxr-xr-x 2 root root 4.0K 2012-06-10 16:36 sbin
drwxr-xr-x 2 root root 4.0K 2011-03-21 09:26 selinux
drwxr-xr-x 2 root root 4.0K 2011-04-26 01:56 srv
drwxr-xr-x 12 root root 0 2012-06-12 23:50 sys
drwxrwxrwt 11 root root 4.0K 2012-06-12 21:58 tmp
drwxr-xr-x 12 root root 4.0K 2011-09-17 19:38 usr
drwxr-xr-x 15 root root 4.0K 2011-04-26 02:12 var
lrwxrwxrwx 1 root root 30 2011-09-17 19:26 vmlinuz -> boot/vmlinuz-2.6.38-11-generic
lrwxrwxrwx 1 root root 29 2011-09-17 18:46 vmlinuz.old -> boot/vmlinuz-2.6.38-8-generic
```

» Директория `root (/)` глазами новичка в Unix выглядит как хаотичная мешанина символов, но на самом деле здесь все полно смысла. У всего есть свое место.

## Вырежьте и сохраните: Синхронизация диска

Сейчас мы вас удивим: когда вы сохраняете файл в программе, он на самом деле не записывается прямо на диск. По крайней мере, это характерно для маленьких файлов (например, менее мегабайта). По соображениям производительности операционные системы не пишут данные на жесткий диск по каждому запросу, а ждут, когда накопится побольше данных с многочисленными запросами на запись.

Поэтому ОС держит все эти записи в буфере ОЗУ, а потом одним махом записывают их на диск. Если

вам когда-нибудь сильно не везло – настолько, например, что вы наблюдали отключение электричества через несколько секунд после нажатия в программе Ctrl+S – то вы увидите это в действии. К счастью, решение есть. В любой момент вы можете в окне терминала ввести `sync`, чтобы гарантированно выполнить все операции записи на жесткий диск.

И еще есть особое сочетание клавиш, которым можно воспользоваться, если повисла X Window System – т.е., графический уровень полностью

блокирован, но вы хотите все синхронизировать с дисками и безопасно перезагрузиться. Оно называется ключом Magic SysRq, и включается в большинстве дистрибутивов следующим образом:

Нажмите и удерживайте Alt+SysRq (обычно в верхней части клавиатуры справа), и затем нажмите следующие клавиши в таком порядке: **R** (возвращение управления клавиатурой), **E** (прекращение процессов), **I** (остановка процессов с ошибками), **S** (синхронизация данных с диском), **U** (размонтирование дисков), и **B** (перезагрузка).

## Что в файловой системе?

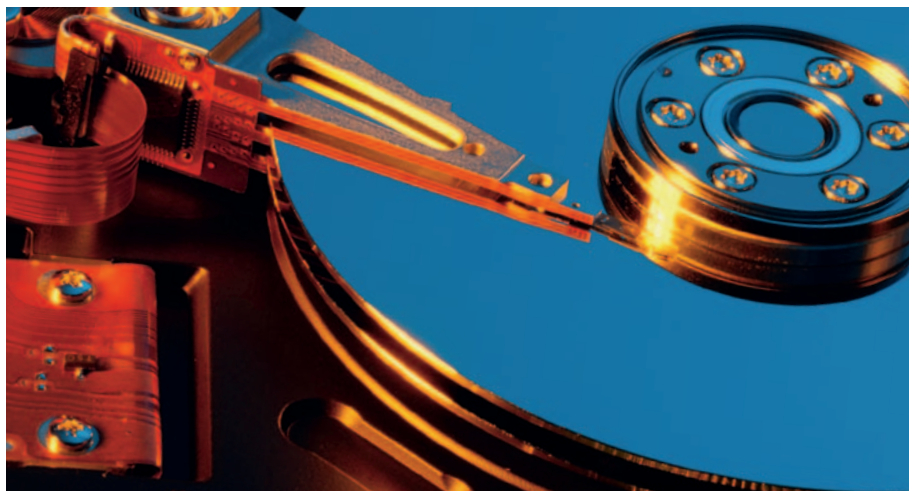
**Ж**есткий диск без файловой системы – это просто нагромождение байтов. Файловая система помогает ОС разобраться со смыслом диска – найти, где файлы начинаются, где заканчиваются, к какой директории относятся. В простых файловых системах, таких, как FAT в DOS, в первых нескольких секторах есть таблица, описывающая расположение фай-

лов. Каждый файл имеет свою запись в этой таблице (поэтому большинство файловых систем имеют ограничение на количество файлов), содержащую имя файла, время создания, размер в байтах, сектор, где он начинается, и т.п. Наиболее популярной файловой системой в мире Linux является ext4, отличная, надежная файловая система общего назначения для жестких дисков. Ка-

кое-то время в мире Linux сосуществовал ее конкурент, ReiserFS, инновационная файловая система, чье развитие оказалось отброшено назад после того, как ее главного разработчика обвинили и позднее осудили за убийство жены...

Есть и другие файловые системы, о существовании которых следует знать; они пригодны для более специфичных задач, чем обычная работа за ПК. Например, ZFS обеспечивает отличную производительность и надежность на нескольких дисках, о чем рассказывается в нашем руководстве по FreeNAS. Затем есть LogFS, разработанная для использования на флэшках (которые работают совершенно иначе, чем вращающиеся жесткие диски, и поэтому получают выигрш от специализированной файловой системы).

Интересно отметить, что при наличии всех продвинутых файловых систем, используемых в наши дни и в Linux, и в Windows, обычные флэшки USB идут с предварительным форматированием в FAT32 (и с его ограничениями). Использование такой отсталой технологии в наши дни кажется странным, но зато эти флэшки совместимы практически со всем. **LXF**



## Вырежьте и сохраните: Восстановление потерянных файлов

Создание резервных копий – единственная и самая важная вещь, которую вы можете сделать для своих данных. И вторая по значимости вещь – создание еще большего количества копий. Но несмотря на благие намерения, мы все периодически совершаем ошибки, и можем случайно удалить файлы. Из-за манеры современных файловых систем разбрасывать файлы по всему диску, во избежание проблем с фрагментацией, программы для восстановления файлов не всегда на 100% эффективны. Однако надежда есть.

Добудьте спасательный дистрибутив с *Photorec* в составе ([www.cgsecurity.org/wiki/PhotoRec](http://www.cgsecurity.org/wiki/PhotoRec)). Один из лучших – Recovery Is Possible, он же RIP – минидистрибутив, разработанный для восстановления поврежденных систем Linux и доступный на [http://](http://tinyurl.com/4bv55)

[tinyurl.com/4bv55](http://tinyurl.com/4bv55). Запишите его на CD-R и храните где-нибудь поблизости от своего ПК на случай аварии. Если потребуется восстановить удаленный файл, выключите ПК и загрузите RIP. В терминале введите 'photorec'; вам предложат выбрать диск и раздел, где содержится файл. Затем у вас спросят о типе файловой системы и о необходимости сканировать весь диск или только место, отмеченное как пустое [empty] (последнее быстрее), а также где сохранять восстановленные файлы.

Эти файлы будут размещены в папках с именами `recup_dir` с приспанным номером. Исходные имена файлов не сохранятся – например, файл `kitten.jpg` может стать `f0015362.jpg`. Если у вас много файлов, их придется смотреть и переименовывать вручную. Но, по крайней мере, вы вернете свои данные...

```
xterm
PhotoRec 6.14-WIP, Data Recovery Utility, January 2012
Christophe GRENIER <grenier@cgsecurity.org>
http://www.cgsecurity.org

Disk /dev/sda - 4294 MB / 4896 MiB (R0)  VBOX HARDDISK
Partition      Start      End      Size in
1 * Linux      0      1      522 42 32  8388

Pass 1 - Reading sector 385352/8388545, 2235 files found
Elapsed time 0h00m05s - Estimated time to completion 0h01m4
pvc: 1346 recovered
lxi: 488 recovered
elf: 299 recovered
lx7: 95 recovered
gz: 8 recovered
png: 2 recovered
zip: 2 recovered
aif: 1 recovered
au: 1 recovered
tar: 1 recovered
```

▶ В процессе восстановления *Photorec* показывает, какие типы файлов определились при поиске по исходным данным диска.

➤ Фил, наш мудрый гуру в области Red Hat.



# Сертифицируйтесь по Red Hat

Стоят ли дорогостоящие программы сертификации своих денег? Вправду ли они улучшают перспективы карьерного роста? **Джонатан Робертс** выясняет.

**Д**аже в разгар самого жестокого после 1930-х годов экономического кризиса (ну, по крайней мере нас так уверяют политики) Linux замечательно чувствует себя в бизнесе.

Например, в LXF160 мы рассказывали о том, как Red Hat стала первой компанией в мире открытого кода, заработавшей свыше миллиарда долларов; в LXF156 Док Браун [Doc Brown] сообщил о возросшем спросе на специалистов в Linux и других технологиях открытого кода. Более того, он рассказал нам, что средняя зарплата для вакансии специалиста LAMP (Linux, Apache, MySQL и PHP/Python) составляет £38500 – прилично больше среднего уровня в Великобритании. И мы не единственные, кто так думает. Linux

Foundation в сотрудничестве с Dice.com, опубликовал 2012 Linux Jobs Report [Отчет о рабочих местах для специалистов Linux за 2012 год] и выявил: 81 % нанимателей сообщили, что задача найма талантливого специалиста в сфере Linux является приоритетной, 85 % говорят о том, что найти талант в области Linux – задача не из легких, и почти треть компаний предлагает работникам со знанием Linux более высокую зарплату.

Принимая во внимание эти позитивные цифры, сложную экономическую ситуацию и ваш оче-

видный интерес к Linux (а иначе с чего вы сейчас читаете *Linux Format*?), похоже, настало время признать, что мечты в плане выбора карьеры сместились к сфере Linux и ИТ. Однако остается проблема: как проникнуть в эту дверь?

В нашей статье мы рассмотрим один из вариантов ответа на этот вопрос: программы обучения и сертификации Red Hat. Нам предоставили место в одном из таких курсов, чтобы мы могли изучить все изнутри и рассказать вам, что вы получите за свои кровные денежки.

Наша главная цель – выяснить, как эти курсы сравнимы с самообразованием и стоят ли они запрашиваемых средств, и помочь вам выяснить, как извлечь из них максимум пользы. Итак, вперед и с песнями!

**«Пора признать, что карьерные мечты сместились к Linux и ИТ.»**



## Самообразование

Прежде чем браться за курсы Red Hat, давайте рассмотрим самообразование как путь к карьере в области Linux, поскольку программы с открытым кодом по своей природе делают его жизненно важной опцией.

Тот факт, что лицензионных платежей на свободное ПО не существует, означает, что любой может скачать и запустить на своем ПК те же программы, что используются на предприятии. Сделав это, вы сможете управлять собственным сервером, собственным сайтом, собственной базой данных... и досконально ознакомиться со всеми тонкостями и подводными камнями системного администрирования. Имеется масса полезной и бесплатной онлайн-документации, а значит, вы не ограничены методом проб и ошибок и можете изучить все достаточно систематично.

Это эффективный способ обучения. Данный подход практиковали многие (в том числе и ваш покорный слуга), и сумели проложить дорогу к рабочим местам, связанным с Linux. Однако у такого подхода есть и проблемы.

Например, в ваших знаниях могут обнаружиться пробелы. Учиться на собственных потребностях вовсе не означает получить навыки, необходимые для успешной карьеры на предприятии. Когда вы проходите собеседование и работодатель просит вас пройти тест, чтобы продемонстрировать ваши умения и навыки, вы, возможно, сначала почувствуете себя уверенно, но потом обнаружите, что столкнулись с задачей, которую даже представить себе не могли, и не думали, что она может оказаться среди требований.

Может также оказаться трудным убедить работодателя пригласить вас на собеседование. Если до этого вы работали в совершенно иной сфере, вам придется выложиться по максимуму, чтобы продемонстрировать соответствующий опыт или предоставить полезные рекомендации.

Именно такие проблемы и призваны решать официальные программы сертификации. Обучающий курс устранит пробелы в ваших знаниях, предлагая вам программу, составленную с учетом требований работодателей, а серти-

фикация позволит вам продемонстрировать свой опыт нанимателям, помогая вам пройти собеседование.

## Необходимый опыт

Если вы задумали пройти обучающий курс, вам надо решить, какой именно курс выбрать. Однако это проще сказать, чем сделать, поскольку Red Hat предлагает целый ряд курсов, и каждый доводит пользователей с разными способностями до одинакового уровня сертификации.

Например, курс RH200, куда Red Hat и допустила нас, «разработан для опытных администрато-

## «Программа составлена с учетом потребности работодателей.»

ров систем Linux», которые хотят усовершенствовать свои знания перед сдачей экзамена Red Hat Certified System Administrator (RHCSA, Сертифицированный Системный Администратор Red Hat); для начинающих пользователей Linux есть курсы RH124 и RH135, которые совместно являют собой более медленный путь к достижению той же цели.

Трудно сказать, какой путь более соответствует вашим умениям и навыкам. Имеется онлайн-оценка уровня умений, которая вас ориентирует; однако многие пренебрегают ее содействием. Дейв Хорнби [Dave Hornby], обучавшийся на одном курсе с нами, сказал: «Я применил онлайн-оценку,

но не воспользовался ее результатами из финансовых соображений». И неудивительно, потому что курсы достаточно дорогостоящи. Ускоренный курс стоит £1955 – изрядная сумма, которую не вынешь

просто так из заднего кармана; а более медленный курс стоит в целом £3480, что недостижимо не только для отдельно взятых личностей, но и для многих компаний.

Дейв был бы идеальным кандидатом для более медленного курса обучения, поскольку у него за плечами опыт системного администрирования в Windows, а опыт работы в Linux невелик. Несмотря на это, он говорит: «Я понимал, что уровень этого курса был очень хорош, но предпочел бы воспользоваться знаниями, полученными непосредственно перед экзаменом... Если раньше вы никогда не работали [с Red Hat], попробуйте перед началом курса наработать опыт, самостоятельно осваивая несложные вещи».

Итак, если ваш опыт работы в Linux тоже не слишком велик, можете просто пройти быстрый курс и сэкономить себе кое-что в процессе. Просто сделайте так, чтобы перед курсом у вас было время освоить основы – возможно, с помощью таких пособий, как *Rute* или *The Linux Documentation Project's System Administration Guide*.

С другой стороны спектра, опытные пользователи Linux, включая меня и двоих инженеров в области телекоммуникаций, поняли, что мы достаточно узнали из учебного курса, и чувствовали себя вполне комфортно, сдавая потом экзамен.

## Творческий подход к созданию резюме

Что делать, если у вас нет необходимых средств, но вы все же хотите построить карьеру в области системного администрирования Linux? Как составить резюме, способное впечатлить потенциальных работодателей? Вот вам несколько идей, которые могут оказаться полезными.

» Прочитайте описание работы и настройте свой сервер, соответствующий перечисленным техническим требованиям. Вы сможете рассказать о нем в своем заявлении и предложить им самим взглянуть на него, чтобы убедиться в ваших умениях. Вы даже можете создать специально для этого интервью имя пользователя и пароль и предложить нанимателю поиграть с вашей системой.

» Fedora и другие проекты открытого кода имеют инфраструктуру, поддерживаемые сообществом.

Подумайте о том, чтобы присоединиться к сообществу ради оказания содействия, и тогда вы сможете приобрести тот самый очень ценный опыт и рекомендации для своего резюме, в то же время оказывая помощь своему любимому проекту открытого кода. В Fedora даже есть обучающая инфраструктура, разработанная для того, чтобы начинающие смогли освоить самые важные умения и навыки.

» Подходите к составлению своего резюме творчески, чтобы продемонстрировать свой профессионализм. Рассмотрите возможность создания резюме в виде map-страницы или даже превратите его в документ LaTeX, дополненный файлом *make*. Потенциального работодателя это непременно впечатлит... или избесит!



► Плата за курсы экономит время по сравнению с самообразованием, а время — деньги!

## Методы обучения

После того, как вы выбрали курс и обратились в центр обучения, чего вам стоит ожидать? Первое, что надо отметить — хотя это будет зависеть конкретно от того преподавателя, который будет вести у вас обучение — не ждите большого количества презентаций в PowerPoint. В нашем курсе мы увидели только один слайд за целую неделю, и это было сделано только для того, чтобы показать нам, что мы будем проходить ежедневно. Все остальное время обучения было посвящено

## «Для нас все уже было готово, чтобы мы освоили курс.»

тому, чтобы научить нас справляться с реальными задачами, при этом наше время распределялось 50:50 между лекционными и лабораторными занятиями.

Однако было бы неправильно называть это лекциями. Временами Фил, наш инструктор, стоял перед нами и рассказывал нам о новых концепциях,

но большую часть лекционного времени мы наблюдали его показ путей решения задач, с которыми мы должны будем справляться. Он останавливался в процессе, чтобы объяснить, что он делает, но справедливо будет сказать, что подавляющее большинство курса мы видели в терминале!

По сравнению с самостоятельным обучением, будь то чтение обучающих руководств или просмотр видео, самым главным преимуществом занятий в классе был колоссальный опыт Фила. Он несколько десятков лет работал с системами Unix и Linux в реальном мире и сравнительно долго занимался преподаванием.

Это означало, что вместо простого объяснения механики выполнения задачи — например, добавления новых пользователей — он мог предложить наилучший вариант, который оказался бы для всех заинтересованных лиц оптимальным. Это также означало, что он лучше знал, какой именно материал представляет наибольшую сложность для слушателей, и у него всегда наготове было несколько объяснений, чтобы материал понял каждый.

Помимо того преимущества, каким является наличие очень хорошего и опытного преподавателя, очное обучение также предоставляет отличный шанс воспользоваться классной лабораторией, чтобы отработать полученные знания на практике. У каждого слушателя был свой компьютер (16 ГБ ОЗУ) с установленным Red Hat Enterprise Linux 6 (RHEL), который выполнял роль хоста для виртуальной машины с другим RHEL 6, но настроенным как сервер. У нас был доступ root к обеим машинам, и мы могли делать с ними все что угодно, в том числе настраивать новые репозитории и новые сервисы и индивидуально работать с настройками Grub.

И физические, и виртуальные машины были объединены в сеть, которая предоставляла нам доступ к сервисам, запущенным как на машинах наших коллег-слушателей, так и на машине нашего преподавателя. Преподавательский компьютер выполнял функцию хоста для всех сервисов, которые нам случилось использовать в течение недели, в том числе NFS, LDAP, DNS и полного набора репозиториями Yum.

## Смешанная настройка

Эта настройка была идеальной, поскольку она означала, что мы можем легко сконфигурировать и использовать web-сервер или иной сервис, а затем у нас был другой компьютер, с которого мы должны были войти и протестировать его.

Это также означало, что когда дело дошло до LDAP или NFS (использование которых не было обязательным по нашей программе), мы могли просто настроить наши машины для работы в качестве клиентов преподавательской машины. Для нас все уже было готово, чтобы мы освоили требования курса. Все отлично работало и было хорошо организовано.

Еще один громадный плюс лабораторий в том, что, хотя занятия начинались не ранее девяти утра, мы могли прийти в любое время с 8 утра, поработать до обеда или воспользоваться свободным временем в конце дня, чтобы дополнительно попрактиковаться. Как ранее сказал Дейв, возможность такой практики очень важна для сдачи экзамена.

Некоторые другие кандидаты рассказывали мне, что на других курсах подготовки к сертификационным экзаменам есть также доступ к вир-

## Важные советы

► **Сначала практика!** Программы с открытым кодом находятся в свободном доступе, так что сначала скачайте Red Hat-подобный дистрибутив, скажем, CentOS или Scientific Linux, и попрактикуйтесь в навыках, которые перечислены в программе курса. Это позволит вам в полной мере воспользоваться преимуществами очного обучения, задавая вопросы, впитывая всю мудрость вашего преподавателя и усиленно практикуясь в лаборатории.

► **Делайте изменения постоянными** В конце экзамена первое, что делает преподаватель — пере-

гружает все компьютеры. Если вы не сделали внесенные изменения постоянными, например, забыли настроить сервис на запуск при загрузке, вы не получите за них никакой оценки.

► **Проверяйте всю свою работу** Весь экзамен основан на выполнении задач, так что вы можете быть на 100% уверены, что выполнили задания, проверив свою работу. Будет даже лучше, если у вас осталось время, перезагрузить компьютер до завершения экзамена и проверить все еще раз. Это спасет вас от провала.

► **Ведите конспект** Red Hat предоставит вам полный учебник курса, но ваш преподаватель будет давать вам дополнительную информацию, советы и подсказки, которые помогут вам во время экзамена. Не забывайте записывать их.

► **Повторяйте ежевечерне** Курсы длятся около недели, так что на неделю освободите себе вечера и каждый вечер еще раз просматривайте пройденный материал. Это очень полезно: тогда в конце недели вам будет проще вспомнить, что проходили в понедельник, если вы пару раз повторяли это.

туальным машинам с облачным размещением, чтобы можно было продолжать практиковаться по вечерам. Это было бы очень неплохим дополнением к тому, что предлагает Red Hat.

## Практические упражнения

Одними из лучших аспектов курса были практические упражнения и оценочные тесты. После каждого этапа предлагалось запустить скрипты, которые должны были либо разрушить, либо определенным образом настроить вашу систему. Далее вам предлагалось либо восстановить систему, либо изменить настройку, а потом запустить другой скрипт для оценки вашей работы.

Это была отличная идея, но в скриптах были кое-какие ошибки, которые не давали им работать так, как следовало бы. Например, один скрипт предложил нам удалить определенный пакет, но скрипт оценки искал другой пакет и сообщил, что мы все сделали неверно. Это было настолько общей проблемой, что Фил предупредил нас, чтобы мы просто сами оценили правильность своих действий. Причем сбои были в самых важных частях курса, поскольку имели тот же формат, что и вопросы на экзамене (хотя материал был другим). Как вам скажет любой недавно закончивший школу или университет, знание технологии экзамена, в частности, того, какие именно вопросы будут на экзамене, является не менее важным условием его сдачи, чем знание материала.

## Экзамен

Ладно, обучение и лаборатория были превосходны, хотя и оставляли поле для улучшений по части послеурочной практики и неверных оценочных скриптов, но каким же был сам экзамен?

Организация была той же, что и во время занятий всю неделю. У каждого из нас был настольный ПК и сервер на виртуальной машине, и мы все были соединены в одну сеть с доступом к сервисам, размещенным на компьютере преподавателя. Основное различие на этот раз было в том, что наш ПК был заблокирован – никакого доступа root, чтобы мы не могли повлиять на объективность оценки, взломать систему или найти ответ в Интернет.

И это правильно: во время экзамена разрешается пользоваться только справочником системы. Это одна из самых трудных частей экзамена. Имея постоянный доступ к Интернету, так просто каждый раз обращаться к Google вместо того, чтобы учить и запоминать материал. Возможно, это делает тест немного нереалистичным, но в то же время заставляет всерьез усовершенствовать свои навыки.

Еще более трудной была практическая, последовательная направленность экзамена. В отличие от других сертификаций, в экзамене Red Hat не было вопросов с вариантами ответа. Это было нечто вроде практического теста: вам дают некую информацию и в соответствии с ней предлагают настроить или исправить систему.

И ни один из вопросов не был отдельным от всего процесса. Например, в начале экзамена предлагалось исправить поврежденный файл



Обучающие компьютеры использовали RHEL 6 PC с установленными виртуальными машинами с RHEL 6.

/etc/fstab, который не позволял загрузиться системе. И пока вы этого не сделаете, не видать вам доступа к другим вопросам!

Дейв сказал, что сочетание всех этих трудностей делает курс «намного сложнее любого курса Microsoft, но в то же время и намного интереснее».

## И оно того стоит?

Остается только один вопрос: стоил ли курс всего этого? Предоставил ли он достаточную добавочную стоимость, чтобы оправдать трату £1955,

**«Даже опытные пользователи Linux узнали много полезного.»**

если самообразование может быть столь же эффективным и при этом не стоить ни копейки? Ответ – безоговорочное «да».

Курс был действительно превосходен и хорошо организован. Фил был отличным преподавателем,

о котором можно только мечтать, лаборатория была хорошо продумана и создавала идеальные условия для обучения и практики, а возможность заранее привыкнуть к стилю экзаменационных вопросов была просто бесценной. Более того, интенсивное обучение в течение целой недели означает, что, пройдя обучающий курс, вы наверняка сможете узнать и научиться большему и намного быстрее, нежели при посредстве самообразования. Даже более опытные пользователи Linux признали, что они узнали много полезного благодаря пройденному курсу.

Если вы сами или ваша компания можете себе это позволить, мы бы рекомендовали вам сделать некоторые вложения в обучение. Да, мы понимаем, что это немалые деньги; и для вас всегда остается вариант потратить некоторое время и некоторые усилия, чтобы самостоятельно овладеть некоторыми знаниями, заплатить за сертификацию и получить те же преимущества при приеме на работу. **LXF**

## Выполняйте домашнее задание

Приступив к обучению на курсе, вы получаете солидный учебник объемом более 300 страниц, где изложено все, что вы будете изучать на курсе. Там содержится основная информация, пошаговые инструкции по выполнению различных задач, практические упражнения и оценочные тесты. Учебник также включает дополнительные ресурсы для чтения и приложение с ответами на все тесты и задания, имеющиеся в нем.

Короче говоря, этот учебник – единственный необходимый вам источник для прохождения курса. Конечно, есть и книги, способные помочь вам подготовиться к успешному прохождению курса или сдаче экзамена.

Среди коммерчески доступных опций – *RHCSA/RHCE Red Hat Linux Certification Study Guide* [Руководство по подготовке к сертификационному экзамену RHCSA/RHCE Red Hat Linux] от Certification Press и *Hands-on Guide to the Red Hat Exams: RHCSA and RHCE Cert Guide and Lab Manual* [Практическое руководство по экзаменам Red Hat: Руководство по сертификации RHCSA и RHCE и Лабораторные инструкции].

Если вы не хотите тратить от £30 до £40 на Руководство, воспользуйтесь онлайн-документацией Red Hat, размещенной на <http://docs.redhat.com>, или System Administration Guide [Руководство сисадмина] от Linux Documentation Project.

# SUSE в истории и современности



**Алексей Федорчук не вытерпел и вернулся к истории – в связи с 20-летием SUSE.**

**В** прошлой исторической заметке (LXF159) я обещал, что она будет последней в цикле. Однако, будучи хозяином своему слову (захотел – дал, захотел – взял), обещание нарушаю. Правда, повод для этого вполне уважительный: если год назад мы праздновали двадцатилетие Linux'a, то теперь аналогичный юбилей наступил для семейства SUSE.

В статье «Лицом к пользователю» (LXF150) история SUSE уже затрагивалась, но лишь начальные ее моменты, да и то очень кратко. Ныне же, по случаю юбилея, есть смысл осветить ее подробно и целиком – от истоков до наших дней.

## Представление семейства

Сочетание символов SUSE в разное время появилось по-разному и имело разное содержание. Сначала оно в форме S.u.S. E. было просто аббревиатурой от названия фирмы, занимавшейся консалтингом и поддержкой UNIX-систем. Когда же эта фирма взялась за разработку собственного дистрибутива, на дистрибутив перешло ее имя. Имя это, утрачивая расшифровку, точки и меняя регистр символов, закрепилось за дистрибутивом надолго – вплоть до его расщепления на коммерческую и свободную линии.

В настоящий момент коммерческая линия представлена дистрибутивом SLE (SUSE Linux Enterprise), свободная же – openSUSE. С последним тесно связан ряд проектов, таких, как:

- » OBS (Open Building System, ранее openSUSE Building System) – автоматизированная система сборки пакетов не только для родного дистрибутива и соплеменного SLE, но и ряда других (Fedora, RHEL, CentOS, Mandriva);
- » SUSE Studio – система автоматической сборки на базе openSUSE и SLE в соответствии с потребностями и пожеланиями пользователя;
- » openQA – система автоматического тестирования созданных образов дистрибутивов;
- » openFATE – система управления возможностями и пожеланиями.

Все они неразрывно связаны с дистрибутивами openSUSE и SLE. А потому ныне SUSE можно рассматривать как общее имя для семейства проектов, охватывающих все стороны развития дистрибутивов – от разработки до распространения. И целью настоящей статьи будет описание того, как SUSE дошла до жизни такой. То есть – ее истории.

## Из предыстории

История SUSE уходит своими корнями в седую древность – в далекий 1992 год. И началась она в городе Нюрнберге или, точнее, в университете Эрлангена–Нюрнберга, когда его недавний студент – Томас Феер [Thomas Fehr] и трое студентов действующих – Бурхард Штайнбилд [Burchard Steinbild], Хуберт Мантель [Hubert Mantel] и Роланд Дюрюфф [Roland Dyroff] – собрались... нет, не выпить самого лучшего пива из Баварии, а учредить фирму по разработке программного обеспечения и оказанию консалтинговых услуг в области UNIX-систем.

Фирма эта получила название Gesellschaft für Software- und System-Entwicklung [Компания по разработке программ и систем]. И первые два года своего существования занималась распространением только что возникших в это время дистрибутивов Linux – сначала SLS от Питера Макдональда [Peter McDonalds], а затем, почуяв безвременную его кончину, Slackware от Патрика Фолькердинга [Patrick Volkerding] (подробности их истории описаны в предыдущих статьях цикла). В сферу деятельности компании входило также оказание технической поддержки пользователей, преимущественно корпоративных.

В 1994 году увидела свет локализованная, то есть немецкоязычная, версия Slackware, которая получила имя собственное – S.u.S. E. Linux, и номер версии – 1.0. Компанию-распространителя нельзя еще было назвать майнтайнером и тем более разработчиком. Но вклад ее в дистрибутив не ограничился германизацией – дистрибутивный комплект из сорока трехдюймовых дисков сопровождался весьма подробной печатной документацией. С тех пор качественная «бумажная» документация на многие годы стала визитной карточкой SUSE и служила образцом, к которому стремились многие другие разработчики дистрибутивов. В частности, на нее ориентировались сочинители документации для Mandrake Linux/RE (впоследствии AltLinux) и ASPLinux.

В 1996 году пути S.u.S. E. и прародительской Slackware расходятся навсегда. В качестве причины источники приводят то, что Патрик не принимал патчи с исправлениями ошибок в его системе, в результате чего немцам приходилось повторять править их в каждой новой версии.

Однако видится и другая причина: к этому времени популярность Linux'a вообще достигла того

критического уровня, когда аскетические средства установки, конфигурирования и управления пакетами Slackware, развивавшегося в качестве типичного «дистрибутива для себя», перестали устраивать потенциальных заказчиков компании S.u.S. E. Которые желали видеть «дистрибутив для всех», подобный набиравшему тогда популярность Red Hat'у – в статье «Linux: начало дистрибуции» я уже говорил, какой смысл тогда вкладывался в понятие «все».

## Начало самостоятельного плавания

Так или иначе, но в 1996 году дистрибутив S.u.S. E. Linux пустился в самостоятельное плавание. Это ознаменовалось:

- » появлением собственной инсталляционной программы по образу и подобию таковой из Red Hat, считавшейся тогда эталоном дружелюбия к пользователю;
- » изменением системы инициализации – с BSD-стиля, исконного для Slackware, на SysV, принятый как в первозданном Linux'e Торвальдса, так и в большинстве распространенных и тогда, и ныне дистрибутивов этой ОС;
- » разработкой первой в истории мироздания и дистрибуции сквозной системы конфигурирования дистрибутива – YaST (Yet another Setup Tool, то есть «Еще один установочный инструмент»), потомок которой, YaST2, используется в дистрибутивах семейства SUSE по сей день;



» Первый «коробочный» релиз S.u.S.E.

» сменой формата пакетов – со свойственных Slackware простых тарболлов на заимствованный из Red Hat'a RPM, быстро ставший наиболее популярным для распространения бинарников независимыми разработчиками.

Не ручаясь, что все эти изменения произошли одновременно – сам я свидетелем еще не был, а однозначных указаний в Сети (за исключением YaST'a) не нашел. Но могу определенно утверждать, что в 1997 году, когда я впервые увидел S.u.S. E., все они уже имели место быть в этом дистрибутиве.

Эта первая оригинальная разработка компании S.u.S. E. получила номер версии сразу 4.2, хотя логика подсказывала в лучшем случае лишь версию с цифрой 2. Почему – тайна сия велика есть. В Сети мне встречалось мнение, что номер версии был взят разработчиками прямо с потолка. Однако рискну высказать иное предположение: в 1996 году увидел свет дистрибутив Red Hat версии 4.0, а затем и 4.1. Разработчики же S.u.S. E. Linux сочли, что их продукт является более «продвинутым» – а учитывая систему YaST, некоторые основания к тому у них были. И потому присвоили ему «опережающий» номер версии: Red Hat 4.2 увидит свет лишь в следующем, 1997, году.

В скором времени SuSE, утратив в 1998 году точки в своем имени, стал дистрибутивом номер один не только в Германии, но и практически во всей Европе, оккупировав на этом континенте ту же нишу, что и Red Hat в Америке. И занял, вслед за последним, второе место по распространенности в корпоративном секторе в мировом масштабе.

Бизнес-модель SuSE строилась по несколько иной схеме, нежели у Red Hat. В частности, этот дистрибутив включал ряд собственных закрытых проприетарных компонентов, в первую очередь – ту же систему YaST и собственную графическую рабочую среду. Входили в него также коммерческие X-серверы – X-Accelerated и MetroX, которые тогда обеспечивали лучшую, по сравнению со свободной модификацией Иксов – XFree86, поддержку видеокарт.

Ни один из этих компонентов не был доступен в исходных текстах. Хотя практиковалось использование свободных X-серверов (в те далекие времена на каждую серию видеочипов приходился свой X-сервер) и оконных менеджеров (время интегрированных десктопов еще не настало).

В «полноразмерном» виде SuSE бесплатно не распространялась – для свободного скачивания была доступна evaluation-версия, по истечении 30 дней приобретающая функциональную ограниченность: утрачивали работоспособность YaST и графическая среда. Что, однако, не препятствовало дальнейшему использованию дистрибутива – ввиду наличия свободных альтернатив в его составе.

«Полноразмерный» дистрибутив в коробочном исполнении продавался за немалые по масштабам тех лет деньги – от 30 до 100 долларов, в зависимости от комплектации. А установочный комплект evaluation-версии распространялся первыми

системами онлайн-торговли по цене носителя и доставки. В том числе и в нашей стране – именно при посредстве такой, ныне забытой, онлайн-фирмы я в далеком 1997 и познакомился впервые с SUSE.

Справедливости ради надо отметить, что коробочные версии сопровождалась печатной документацией различного, в зависимости от цены, объема, но неизменно превосходного качества, которая сама по себе составляла львиную долю стоимости коробки. А «старшие» коробки включали и разнообразную атрибутику – в частности, значок с изображением хамелеона, исполненный почти как правительственная награда.

Видимо, непосредственная продажа дистрибутивов в коробочном исполнении и составляла основную статью дохода фирмы. Хотя и от технической поддержки в корпоративных масштабах она тоже не отказывалась.

## К технологическим высотам

Ближе к концу тысячелетия, одновременно с успехами в бизнесе, продолжается поступательное технологическое развитие дистрибутива SuSE.

Оно выражается, во-первых, в весомом вкладе в совершенствование графической системы, место которой в Linux'e и остальных UNIX-подобных ОСях к тому времени почти безраздельно заняла свободная инкарнация оконной системы X –

## «Продолжается поступательное технологическое развитие дистрибутива SuSE.»

XFree86. Тесные контакты как с разработчиками последней, так и с рядом производителей видеокарт, в том числе и профессиональных, наподобие германской фирмы Else, обеспечили дистрибутив поддержкой самых современных тогда решений.

Во-вторых, осенью 1998 года SuSE, сразу вслед за Mandrake, включила в свой состав KDE – первую (и тогда единственную работоспособную) интегрированную графическую среду. Что, с точки зрения пуристов свободного софта, в частности, Ричарда Столлмена [Richard Stallman], выглядело крамоллой, так как лежащая в ее основе библиотека Qt распространялась не под свободной лицензией. Тем не менее, судьбы SuSE и KDE оказались тесно связанными, и связь эта не разорвана и по сей день.

И Иксы с хорошей поддержкой «железа», и KDE как бы ориентировали SuSE в направлении десктопов – ведь к рубежу тысячелетий в прессе все чаще стали поговаривать о Linux-буме именно касаясь настольного его применения. Что получило свое выражение в вариантах основного дистрибутива – SuSE Linux Office Desktop и SuSE Linux Desktop, имена которых говорят сами за себя.

Однако не меньшее внимание в развитии SuSE уделялось и серверному направлению. Апофеозом чего стало появление в 2001 году SuSE Linux Enterprise Server (SLES), функционирующего не только на традиционных PC, но и на рабочих



» Грег Кроа-Хартман внес заметный вклад в разработку ядра Linux.

станциях IBM POWER и даже на майнфреймах серии s/390, как 32-, так и 64-битных. Со временем он стал по-настоящему кросс-платформенной системой, включив в себя поддержку архитектур Intel Itanium и x86\_64.

В общем, все было понятно и привычно: хороший дистрибутив развивался и становился еще лучше. И продолжалось это до осени 2003 года, когда мир открытого софта облетела весть о покупке фирмы-производителя SuSE компанией Novell, известной своей сетевой операционной системой NetWare. И считавшейся (обоснованно или нет – другой вопрос) одним столпов проприетаризма,

## Рождение openSUSE

Объявление о том, что Novell покупает компанию S.u.S.E. вместе с ее дистрибутивом, вызвало большое волнение в сообществе Open Source и опасения за будущее SuSE. Задолго до завершения сделки (как известно, такие дела с кондачка не решаются, требуя одобрения всяких антимонопольных контор) посыпались мрачные прогнозы.

Однако опасения оказались напрасными. Ибо первым деянием Novell в рамках проекта после покупки, кроме очередной коррекции имени дистрибутива (отныне и по сей день он величался SUSE или включал в свое имя этот компонент), стало открытие исходных текстов системы YaST (к тому времени к имени ее добавилась цифра 2) на условиях лицензии GPL 2.

Следующий шаг, сделанный компанией Novell в 2005 году, был еще более радикальным – единый проект был расщеплен на две ветки: коммерческую SUSE Linux Enterprise и свободную openSUSE. Первая осталась в ведении Novell; управление второй было целиком отдано в руки сообщества независимых разработчиков. Что, однако, не исключало тесного взаимодействия между ветвями, в том числе и участия одних и тех же лиц (например, Грега Кроа-Хартмана [Greg Kroah-

## Отработка кульбита

Интересно, как разработчики SLE будут выкручиваться из этой ситуации теперь, с появлением радикально «улучшенного» Gnome 3 с его резким изменением парадигмы и ориентацией на гаджеты, мало уместной в корпоративной среде. Это касается всех разработчиков Enterprise-систем,

в которых затраты на переучивание персонала корпоративных заказчиков являются существенным ограничителем на внедрение уж очень революционных инноваций. Впрочем, это заботы корпоративщиков и «энтерпрайзеров». Нас же больше интересует развитие свободной ветки.

Hartman], известного разработчика ядра Linux) в обоих проектах.

Дистрибутивы коммерческой ветки развивались на базе уже существовавшей SuSE Linux Enterprise Server – как в традиционно серверном (SLES), так и десктопном (NLD – Novell Linux Desktop, позднее преобразованный в SLED) направлениях. Кроме того, к ним присоединился дистрибутив OES (Open Enterprise Server), интегрирующий в себе Linux и NetWare и содержащий

KDE, а начиная с версии 10, приобрел в них статус рабочей среды по умолчанию.

Развитие свободной ветки проходило параллельно и опережающими темпами. Первой версией openSUSE, разработанной сообществом, стала 10-я, основанная еще на разработках прежней SuSE. Однако уже версия 10.1, под влиянием SLE, получила в качестве одного из основных десктопов, в дополнение к KDE, также и Gnome.

Правда, Gnome в качестве основного десктопа в openSUSE не прижился. И это – несмотря на выход в начале 2008 года версии KDE 4.0, принятой пользователями очень неоднозначно. Тем не менее,

уже в openSUSE релизе 11.2 (2009 год) KDE, достигший тогда вполне работоспособной версии 4.3, опять становится основным десктопом. Хотя сборки openSUSE с Gnome, как и другими интегрированными средами (XFce, LXDE) и оконными менеджерами (например, Enlightenment) продолжают регулярно выходить. Хотя только сборки с Gnome имеют официальный статус – забота об остальных находится в руках волонтеров.

### Соглашение с дьяволом

Однако я забежал несколько вперед. Возвращаясь к хронологии, надо рассказать об одном из тех событий, которые потрясли мир (Open Source, разумеется – многие обитатели этого мира вообще отличаются повышенной склонностью быть потрясенными или потрясаемыми).

«Случилось это в осень, в ноль шестой год» (а точнее, 2 ноября 2006 г.), когда... нет, в Питере с нагана никого не убили. Но компания Novell, к тому времени уже более двух лет владевшая SUSE Linux Enterprise, а также бывшая главной опорой openSUSE, и корпорация Microsoft объявили о начале сотрудничества в технической, маркетинговой и патентной сферах.

Заявление это вызвало обвинения Novell

в «делке с дьяволом» и грозные предупреждения о том, что такие сделки добром никогда не кончаются. Каюсь, и автор этих строк в посвященном сему событию очерке был не слишком оптимистичен.

Однако ничего страшного не произошло и на этот раз. Сотрудничество с Microsoft в технической сфере вылилось в облегчение интеграции Linux и Windows в гетерогенных сетях, в частности, увязку фирменных служб каталогов (eDirectory и ActiveDirectory, соответственно). Маркетинговое сотрудничество имело своим следствием рост продаж SUSE Linux Enterprise в несколько раз (в сетевых источниках фигурирует цифра до 250 %).

Что же до сотрудничества в патентной сфере, вызвавшее наибольшие попреки, вплоть до обвинений в государственной измене идеалам Свободы, то оно могло коснуться только корпоративных заказчиков, и только в одной слабообразованной по части патентного законодательства стране. Да и там никаких реальных последствий не имело. А косвенно это соглашение способствовало повышению внимания к патентной чистоте кода, используемого в открытом и свободном программном обеспечении.

### Парадоксы бизнеса

Правда, по прошествии некоторого времени после соглашения с Microsoft финансовые дела у Novell пошли не очень хорошо. Однако, как учит нас классическая логика, «после того» вовсе не значит «вследствие того». И причиной тут было не соглашение по поводу Linux'а – как раз Linux-бизнес компании развивался вполне успешно.

Правда, частыми версиями фирма не баловала – за период с 2004 по 2010 год мажорных релизов вышло всего три: 9, 10, и 11. Однако в течение каждого релиз-цикла регулярно выходили сервис-паки, которые по обилию и характеру обновлений часто вполне тянули на настоящие релизы.

Да и свободная ветка в лице openSUSE продолжала радовать пользователей своим развитием. Новые релизы ее выходили с регулярностью раз в 6–8 месяцев. Начиная с версии 11.2 (октябрь 2009 года), релизам стали присваиваться кодовые имена. В отличие от Ubuntu или Fedora, в качестве таковых выступали не звери, реальные или мифические, и не малопонятные словосочетания, созвучные именам великих ученых и населенных пунктов, а оттенки зеленого – исконного геральдического цвета SUSE. И дизайн каждого релиза оформлялся в соответствие с оттенком-эпонимом.

Первый «цветной» релиз был оформлен в изумрудных [Emerald] тонах, следующий, 11.3 – в бирюзовых [Cyan], релиз 11.4 переливался морской волной [Celadon], а текущий, 12.1, окрасился в цвет листьев спаржи [Asparagus]. Ныне разрабатываемый релиз, 12.2, по выходе приобретет расцветку богомола [Mantis].

Так что, раз у линии SLE хватало сил на выпуск очень серьезных сервис-паков, а у сообщества openSUSE оставалось еще время на подбор оттенков, гармонирующих с номером очередного

## «Правда, Gnome в качестве основного десктопа в openSUSE не прижился.»

общие для них сетевые службы (реализованные, к сожалению, в виде проприетарного кода).

Одним из существенных новшеств, принесенных в десктопную линию коммерческой ветки, стало включение в SLED рабочего стола Gnome. Причина была в том, что незадолго до покупки S.U.S.E. Novell начала активно участвовать в проекте Mono – свободном воплощении среды разработки .Net от фирмы Microsoft. Проект же этот основывался на библиотеке GTK и был тесно интегрирован с Gnome. В частности, и потому, что основоположником обоих проектов был один и тот же человек – Мигель де Икаса [Miguel de Icaza].

Так что казалось естественным, что сначала, в версии 9 обоих дистрибутивов SLE, Gnome оказался на равных правах с традиционным для SuSE



► Мигель де Икаса, основатель проекта Mono.

релиза, с Linux'ом в компании Novell был полный порядок. А главная причина финансового упадка фирмы крылась, видимо, в том, что время NetWare, на которой в значительной мере строился ее бизнес, прошло окончательно и бесповоротно. И никакой интеграцией с Linux'ом эту систему было уже не реанимировать.

## Продажа бессмертной души

Завершилась «упадочническая» тенденция в развитии Novell очередной покупкой, о которой было объявлено в ноябре 2010 года. Только на сей раз, как сказал бы философический принц Датский, ужинал не Полоний – ужинали Полония. То есть компания Novell впервые выступила не в роли покупателя.

Хотя помянутая роль была ею хорошо освоена еще в 90-х годах, когда, одержимая, подобно сиятельному Камильбеку из «Повести о Ходже Насреддине», хватательным рвением, она скупала все, до чего могла дотянуться:

- » операционные системы – DR DOS у фирмы Digital Research, ставшей Novell DOS;
- » настольные СУБД – *Paradox* у фирмы Borland;
- » офисные пакеты, скомпонованные перед этим той же фирмой Borland в виде сборной солянки из текстового процессора *WordPerfect*, электронной таблицы *QuattroPro*, презентационной программы *Presentation*.

Но теперь компания Novell оказалась в амплуа продавца – причем себя же. А покупателем ее товара выступила корпорация Attachmate, о которой до того времени в кругах Open Source, не интересующихся «большим бизнесом», мало кто знал.

Неизвестность всегда рождает опасения. И в который уже раз по миру Open Source поползли слухи, что за спиной этой сделки стоял все тот же ухмыляющийся дьявол из Microsoft (которой действительно досталось акций компании Novell примерно на пятую часть их суммарной стоимости). А сама сделка якобы имела своей целью нанести сокрушительный удар защитникам свободы, задушив одного из самых мощных и влиятельных игроков на этом поприще.

Как и в прошлых аналогичных случаях, слухи о близкой кончине SLE, а с ней и openSUSE, оказались несколько преувеличенными. По крайней мере, за прошедшее время (а оформление сделки было завершено в начале 2011 года) никаких тому подтверждений получено не было. Как и вмешательства дьявола Microsoft в посясторонние дела разработки коммерческой и свободной веток SUSE.

## Последствие

А вот корпорация Attachmate проявить себя успела – и есть мнение (и не только мое), что с хорошей стороны. Первым проявлением ее деятельности было разделение бывшей компании Novell на две независимые (друг от друга) группы:

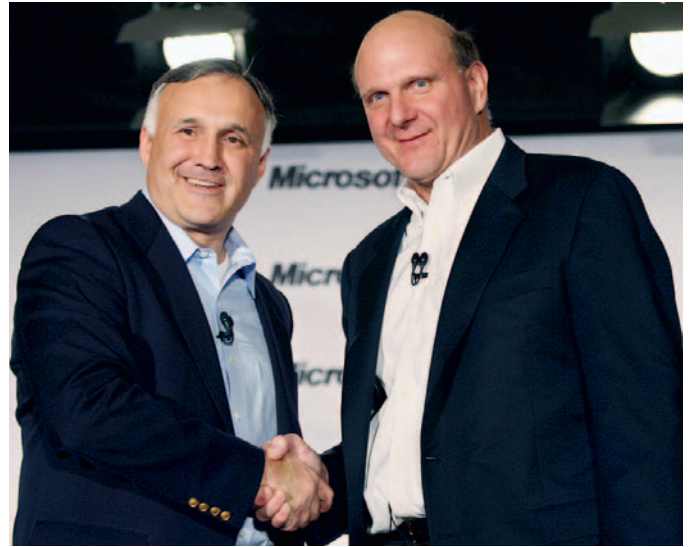
- » собственно Novell, которой достался весь груз NetWare'вского прошлого и в довесок к нему – проект ORS;
- » SUSE, унаследовавшей как SLE, так и контакты с разработчиками openSUSE.

Не знаю, как это скажется на Novell, но для SUSE, по моему (и опять же не только моему) мнению, это означает возможность сконцентрироваться на развитии собственно Linux-проектов, без отвлечения на посторонние «мелочи» вроде взаимодействия с NetWare.

Впрочем, поживем – увидим: времени прошло еще слишком мало. И для линии SLE оно поначалу казалось мертвым сезоном. Однако 28 февраля 2012 года прошло сообщение о выходе SUSE Linux Enterprise 11 SP 2 – как я уже говорил, сервис-паки SLE по серьезности обновлений тянут на полноценные релизы многих более иных дистрибутивов.

Этот же сервис-пак знаменателен еще и изменением метода обновления функциональности дистрибутива. На смену модели back-портирования (включения в старые версии ядра, Иксы и другие ключевые компоненты функциональности, достигнутой в новых их версиях, вышедших за время жизни текущего релиз-цикла) пришла модель forward-портирования. Отныне в дистрибутив будут включаться новые версии ядра и других базовых компонентов, которые дополняются специфичными для SUSE возможностями и модифицируются для сохранения совместимости с предыдущими версиями системы. Процесс этот будет непрерывным, следующим генеральной линии развития критически важных составляющих дистрибутива.

Ну, а про интенсификацию развития openSUSE под властью Attachmate и говорить нечего. Во-первых, новые релизы выходят регулярно и, до последнего времени, с прежней периодичностью, раз в 8 месяцев. Правда, с грядущим релизом 12.2 вышла задержка: вместо обещанного



» «Сделка с дьяволом»: президент Novell Рон Овсепян [Ron Hovsepian] (слева) и президент Microsoft Стив Балмер [Steve Ballmer] 2 ноября 2006 года.

июля он должен появиться в сентябре. Но это связано не столько с замедлением разработки, сколько со сменой версии компилятора и рядом других технических факторов. А может, разработчики хотели подгадать к юбилею своей системы?

Во-вторых, начиная с версии 11.4 (а она вышла уже по завершении сделки), и в свободной ветке, как и в коммерческой, изменилась модель обновления дистрибутива, причем еще более радикально, нежели в SLE. А именно, здесь появился репозиторий Tumbleweed, подключение которого превращает openSUSE в rolling-систему, то есть непрерывно обновляемую по скользящему графику.

Я далек от мысли приписывать все эти, безусловно полезные, новшества благотворному влиянию корпорации Attachmate. Однако их появление свидетельствует как минимум о том, что нынешний владелец не чинит препятствий творческой мысли разработчиков и коммерческой, и свободной ветки. Не видно с его стороны и навязывания заведомо непопулярных в народе инноваций.

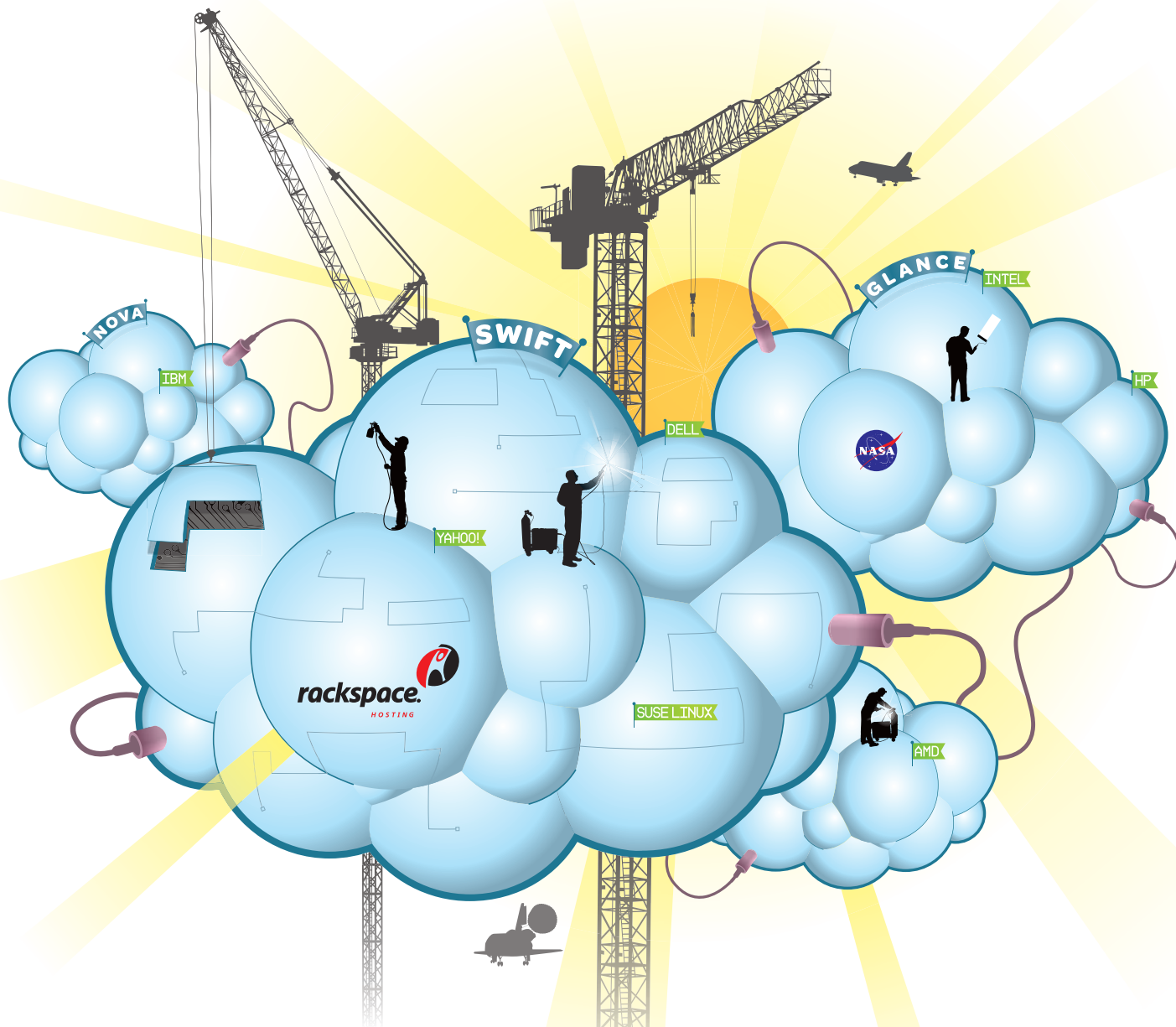
Так что современное положение openSUSE внушает оптимизм. Надеюсь, что он оправдается с выходом релиза 12.2. **LXF**

## SUSE на Руси

В 90-х годах прошлого века SUSE в нашей стране была менее известной, чем Red Hat, но пользовалась устойчивой симпатией в определенных, хотя и узких, кругах. И имела хороший шанс получить более широкое распространение. Ибо в самом конце 90-х IPLabs Linux Team (та самая, которая потом стала Altlinux'ом) предприняла попытку распространять SUSE в России примерно на тех же условиях, что и Mandrake/RE. То есть – с пакетами русификации, дополнительным софтом, актуальным для наших пользователей, и русскоязычной документацией. А Алексей Новодворский и Алексей

Смирнов даже съездили в Мюнхен на переговоры по этому поводу. После чего в отчете о поездке написали историческую фразу: «Самое лучшее из Баварии – это, конечно, не йогурт. Самое лучшее из Баварии – это, конечно, пиво».

Но, вероятно, вследствие известной любви немцев к *deutsche Ordnung*, нашему бизнесу категорически противопоказанному, высокие договаривающиеся стороны к консенсусу так и не пришли. А иначе имели бы мы нынче что-нибудь вроде openSUSE/RE. Но увы – история сослагательного наклонения не имеет...



Что за штука...

# OpenStack

**Грэм Моррисон** избегает облачных метафор, пытаясь удержать почву под ногами.

**В** OpenStack ведь не такая уж и новинка, не так ли? Что-то вы перестали успевать за событиями.

**О** Да, вы правы. Об OpenStack уже слышаны. Еще в 2010-м компания Rackspace, известная хостингом многих сайтов, начала сотрудничать с NASA – агентством, прославленным якобы отправкой астронавтов на Луну. Их совместный проект стартовал после публикации записи в блоге одного из сотрудников NASA. Заголовок был: «Вы-

шла NOVA – Облачная технология под лицензией Apache, на Python. Живая, кривая, бета. Проверяйте». И далее, NASA и Rackspace вместе творили своего рода фантастический сетевой мир, где хранение, ресурсы и производительность существуют как нечто само собой разумеющееся, и маленькие стартапы могут спокойно строить в облаках свои башни из слоновой кости, зная, что когда их день придет, они смасштабируют все вверх, быстро и эффективно, прежде чем продаться Facebook.

С OpenStack такое стало реальностью: ведь это открытый код, работающий подобно всему Linux. Да многие и называют это «Linux облачных вычислений». Можете оплатить кому-нибудь слежение за вашим OpenStack, а можете просто поэкспериментировать с ним, зная, что эту игрушку всегда можно превратить в мощное орудие. Как и Linux.

**В** Нет. Пожалуйста, только не про облака. Я ничего в этом не понимаю.



**О** Пожалуй, главная аналогия между облачными вычислениями и обычными облаками – то, что никто не может дать им точного определения: определения здесь переменчивы, как погода. Можно сказать, что это и Gmail, и Google Docs, но это и ваша учетная запись на Facebook, и календарь Exchange.

Если вы трудитесь в технической сфере, вполне возможно, что IT-отдел вашей компании уже меняет стойки серверов на арендуемое пространство где-нибудь на облаке; или, скажем, вам нужно передать большой объем научных данных на EC2 от Amazon и получить результаты вычислений за минуты, а не за недели, как это будет на обычном настольном ПК.

### **В** То есть более конкретно о функциях OpenStack сказать нельзя?

**О** Если говорить конкретнее, есть риск погрязнуть во всяких страшных сокращениях, например, IaaS – и ослиный клич здесь вовсе ни при чем, ведь это означает Infrastructure as a Service [Инфраструктура как Услуга]. В этом случае облако сродни IT-отделу из предыдущего примера, когда компания меняет собственную инфраструктуру на ту, что она получает в облачном объеме. Это та сфера бизнеса, которую Rackspace жаждет контролировать, и не без серьезной причины.

Когда одна компания управляет инфраструктурой, позволяя сэкономить любой компании из тех, что желает избавиться от своего IT-отдела, на этом можно сделать огромную прибыль. И это более чем привлекательная модель бизнеса. OpenStack работает на том же уровне. Его можно использовать как IaaS, ведь вы можете делать с ним, что хотите – как с виртуальной машиной, например.

На другом конце шкалы окажется SaaS, то есть Software as a Service [Программное обеспечение как услуга]. Это проще понять, поскольку вы управляете не целой инфраструктурой, а лишь одним конкретным приложением. Это может быть ваша внутренняя система управления работой с клиентами, или любая другая глобальная система, работающая на постоянный доступ.

Существенное различие между SaaS и прежними моделями серверов заключается в том, что сами приложения предоставляются владельцем облака. Вот простейший пример: вы платите Google за право корпоративного доступа к почте, календарю и документам, вместо того, чтобы создавать подобные программы самостоятельно или платить другой компании за хостинг ваших серверов.

С SaaS вам не нужно заботиться об установке, развертывании и поддержке (если, конечно, это не услуги, предоставляемые на вашем облаке). OpenStack становится сходным с SaaS, если вы настроите его так, чтобы на нем была некая услуга, которую вы будете продавать клиентам – к примеру, почта или редактирование документов.

### **В** Хорошо, но что же особенного именно в OpenStack?

**О** Есть ряд причин, делающих его особенным и достойным вашего внимания. И первая – это действительно открытый код. Все его компоненты выпущены под лицензией Apache, немного более либеральной, чем GPL.

Это значит, например, что любые изменения можно публиковать под другой лицензией, и эта особенность стала своего рода визитной карточкой команды OpenStack. Вот фрагмент из пятиминутного обзора проекта: «Мы твердо верим, что открытая модель разработки – единственный путь стимулировать так необходимые нам облачные стандарты, забыть об угрозе приватизации облачной сферы и создать огромную экосистему сосуществующих облачных провайдеров». При открытом API, открытых форматах и полностью свободных исходниках, вы поймете, почему OpenStack очень важен именно сейчас, когда на облачные технологии тратится столько денег. С большой вероятностью можно утверждать, что компании, пока что работающие со своими частными облаками во внутренних сетях, со временем

## «В OpenStack проприетарные компоненты отсутствуют в принципе.»

передадут их провайдерам OpenStack, таким как Rackspace, когда посчитают нужным.

Другая причина в том, что OpenStack уже стал своеобразным феноменом. Стоит только посмотреть на официальный список из 180 компаний, которые признали, что используют его. Помимо Rackspace, среди них вы найдете Dell, AMD, Cisco, HP, AT&T, Broadcom и Yahoo. NASA, увы, перебирается на Amazon.

Можете заглянуть и в список Linux-гигантов, все чаще обращающихся к OpenStack – здесь и RedHat, и SUSE, и Canonical. У RedHat прежде были свои облачные проекты, такие как Aeolus – довольно удачное решение для создания собственной виртуальной машины, как внутренней, так и развернутой на множестве несовместимых облаков. Но в апреле, перейдя на новую систему управления, они все же приняли OpenStack.

### **В** Но ведь Canonical вроде тоже активно развивала собственный облачный проект, Eucalyptus?

**О** Да, это так. Но около года назад владельцы Ubuntu объявили о переходе на OpenStack. По-видимому, такое решение было продиктовано тем, что Eucalyptus менее открытый проект, чем OpenStack. Это тоже открытый код, но релизы, к примеру, выходят раз в полгода. А в OpenStack проприетарные компоненты отсутствуют в принципе.

### **В** То есть в OpenStack несколько компонентов?

**О** Да, как таковых, их три. Первый называется OpenStack Compute. Это та самая ‘Nova’, о которой говорилось в первом посте, и это собственно инфраструктура. Среда обитания виртуальных машин. Compute предоставляет через API доступ для настройки и управления ими. Здесь разработчики могут не просто работать на виртуальном оборудовании, но также заниматься масштабированием и распараллеливанием.

Несмотря на низкоуровневую природу упомянутых операций, Compute написан на Python, и многие разработчики предпочитают использовать именно привязки на Python. Реализована поддержка различных гипервизоров (систем выполнения виртуальными машинами). В этом его отличие от Eucalyptus, который поддерживает только Xen и KVM, что стало возможно только после значительного содействия со стороны Canonical.

OpenStack же поддерживает не только KVM и Xen, но и VMWare, LXC (Linux Containers – ищите учебник в следующем номере), User Mode Linux

и даже Qemu, хотя в документации в отношении двух последних сказано, что их лучше использовать в разработке, нежели сами по себе.

### **В** А что вы можете сказать про другие компоненты?

**О** В то время как инфраструктурой управляет Nova, следующим обязательным требованием для облачной системы является хранилище. В отличие от компьютера, на котором вы это читаете, хранение в облаке невозможно без распределенных систем, процессов и динамической емкости.

Этот компонент в OpenStack называется Object Storage, или Swift, и был изначально разработан Rackspace. Как и Nova, он написан на Python и поддерживает и файлы (объекты), и блоки, что пригодно для долгосрочного хранения образов дисков виртуальных машин.

Есть также множество дополнительных функций, таких как возможность легко увеличивать емкость или автоматически исправлять повреждения, и масштабировать до размеров, которые большинство пользователей себе даже не представляет – множества петабайт и миллиардов объектов.

Наконец, Swift связан с Nova при помощи OpenStack Image Service, который отвечает за поиск, регистрацию и доставку образов виртуальных дисков. **LXF**



# По рецептам доктора Брауна

**Д-р Крис Браун**

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

## Вести из будущего

**М**ои попытки путешествий во времени в целом не были успешными. Однако мне удалось создать кратковременное возмущение в пространственно-временном континууме, и я смог загрузить любопытный отчет из Linux Format № 13316 от июня 3024 года. И вот он перед вами:

«Цифровым палеонтологам из Института Мира во всем мире в г. Шаттлворте недавно удалось загрузить древний компьютер, обнаруженный на чердаке в Пекхэме-на-море. Оказалось, что на нем установлена операционная система Windows 7. В археологических исследованиях в данной сфере есть значительная неопределенность (и белые пятна), так как в настоящее время отсутствуют материальные свидетельства о компьютерах с Windows 4, 5 или 6, однако, как это ни странно, науке известны компьютеры с Windows 95, 98 и даже 2000. Хроностратиграфическая последовательность не определена.

Но какова судьба этих компьютеров с Windows? Начиная примерно с 2020 года, сведений о них нет. Популярную теорию о том, что с облаков спустился Google Android и забрал их на планету Зог для подсчета, вряд ли стоит принимать всерьез. Вероятнее, имело место катастрофическое событие, которое привело к появлению смертоносного вируса – нечто вроде К-Т события, приведшего к вымиранию динозавров, но без участия астероида.

Как полагают ученые, уровень генетического разнообразия среди компьютеров Windows на тот момент был столь низок, что все они пали жертвой палеовируса, позволив гораздо менее распространенным системам Linux эволюционировать и занять освободившееся место в экологической нише».

Но сам я отношусь к этому отчету скептически. Не могу поверить, что кто-то в редакции *Linux Format* сможет выговорить слово «хроностратиграфический».

[chris.linuxformat@gmail.com](mailto:chris.linuxformat@gmail.com)

## Эзотерическое системное администрирование из причудливых заворотов кишок серверной



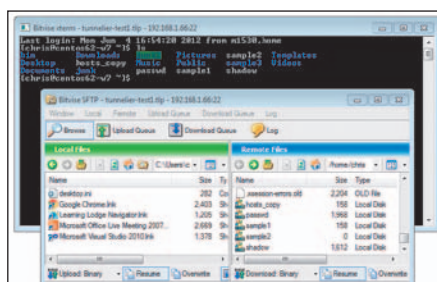
# Bitwise Tunnelier

Если между вами и вашей Linux-машиной затесался ПК с Windows, попробуйте этот клиент SSH и SFTP.

**Н**едavno мне на глаза попала программа *Tunnelier* от Bitwise ([www.bitwise.com](http://www.bitwise.com)). Это программа для Windows, и вам прости-тельно спросить: с какой стати я пишу о ней в журнале о Linux? А дело в том, что в ней есть клиенты SSH и SFTP, с помощью которых можно навести мост между мирами Windows и Linux. Загрузите установщик с сайта [www.bitwise.com/tunnelier-download](http://www.bitwise.com/tunnelier-download); установка проста, а программа бесплатна при количестве пользователей до четырех. Перед подключением к серверу SSH нужно создать профиль, введя имя хоста или его IP-адрес. Этот профиль также может включать имя пользователя и какую-либо аутентификацию – пароль, публичный ключ SSH или идентификационные данные GSSAPI. Включив их в профиль, можно просто нажать на кнопку Login, и вы сразу подключитесь к удаленному терминалу. У профилей есть имена, профили можно сохранять в файлы

и загружать их. Подключиться с *Tunnelier* можно прямо из командной строки Windows. По команде `tunnelier -profile=tunnelier-test.tlp -loginonstartup` вы попадете прямо в командную строку компьютера, указанного в профиле *Tunnelier-test*, если в профиле есть логин и данные аутентификации.

Наряду с клиентом SSH вы получаете xterm или эмулятор терминала VT100 и графический клиент SFTP. Это обычный двухпанельный клиент: на левой панели – локальные файлы, на правой – удаленные. Одна из понравившихся мне возможностей – журнал активности *Tunnelier* в главном окне, своего рода его поток сознания. Он дает мне теплое, пушистое ощущение, что все правильно, и полезную диагностическую информацию, когда что-то не ладится. В *Tunnelier* мало зрелищности – но это надежная, полнофункциональная программа, с которой приятно работать. Чего же вам боле?



➤ С помощью *Tunnelier* я получаю командную строку и доступ по SFTP с компьютера Windows на свои серверы Linux.

## Совершенство слов

Всем нам досаждают плохие технические тексты, будь то убалтывание маркетологов, невразумительные инструкции или просто дурная грамматика. Поэтому я рад сообщить, что сайт Bitwise – один из самых хорошо написанных сайтов, которые я встречал. Его стоит посетить, даже если вам не нужна программа: там есть прекрасные руководства по таким темам, как перенаправление портов SSH и туннелирование, и даже статья «Как работает Интернет».

# 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](http://www.linuxcenter.ru)

Linux  center

# Вы решили стать сисадмином?

Первая часть серии, которая превратит вас из зеленого новичка в звезду системного администрирования.

**Н**а мои курсы приходят желающие стать системными администраторами (но есть и те, что не желают, кричат и отбиваются, и из расслабляюще-комфортной среды Windows их надо вытаскивать за уши). Мне задают самые разные вопросы, но три самых частых выглядят примерно так:

**ВОПРОС 1: Я хочу стать системным администратором Linux и разбогатеть. Стоит ли специализироваться на одном конкретном дистрибутиве?**

**ОТВЕТ:** Мой инстинктивный ответ: «У универсала больше шансов получить работу, чем у специалиста». Но это зависит от обстоятельств. Например, если вы хотите получить работу в Canonical, пожалуй, стоит сосредоточиться на Ubuntu. Или вы можете устраиваться на работу в какую-то компанию, использующую SUSE. А если у вас нет четкой специализации, но вы хотите сфокусироваться на одном дистрибутиве, им должен стать Red Hat Enterprise Linux.

**ВОПРОС 2: Я несколько лет администрирую Windows и делаю все необходимое в графических утилитах. Стоит ли знакомиться с утилитами командной строки для администрирования Linux?**

**ОТВЕТ:** Коротко – да. В современных дистрибутивах Linux в графических утилитах можно выполнять достаточно много классических задач администратора, таких как создание учетных записей пользователя и установка новых программ. Но рано или поздно эти утилиты исчерпают себя, и командная строка останется единственным возможным вариантом. Например, я не знаю, как задать дисковые квоты не в командной строке. Кроме этой, есть еще по меньшей мере три причины освоить командную строку.

## Полюбите командную строку

Во-первых, навыки работы в командной строке чаще всего одинаковы для всех дистрибутивов, а графические утилиты в раз-

ных дистрибутивах отличаются. Во-вторых, написав небольшой скрипт с использованием нескольких утилит, можно сотворить чудеса с производительностью. Графическими утилитами управлять из скриптов нельзя. И третья, но не последняя по значимости, причина – есть все шансы, что на многих серверах просто не будет графических утилит и графических рабочих столов.

**ВОПРОС 3: С чего начать?**

**ОТВЕТ:** Практикуйтесь. ISO-образы большинства популярных дистрибутивов можно загрузить бесплатно. Можно установить их либо на настоящие компьютеры (если у вас есть свободные), либо на виртуальные машины *VMWare* или *VirtualBox* даже на компьютерах с Windows. *VirtualBox* бесплатен, как и *VMWare Player*. Либо можно воспользоваться бесплатным сервисом AWS и получить компьютер на облаке Amazon на один год. Лучше настроить два компьютера, чтобы поработать с сетью. Затем просто практикуйтесь: создавайте новые файловые системы, настраивайте веб-сервер и брандмауэр. Выведите систему из строя, чтобы она не загрожалась. Верните ее в рабочее состояние. Сделайте это снова.

Получайте удовольствие! Одно из удовольствий Linux в том, что с ним легко практиковаться. Вы учитесь не на хирурга – недостатка тел для практики нет.

Довольно наставлений. Пе-

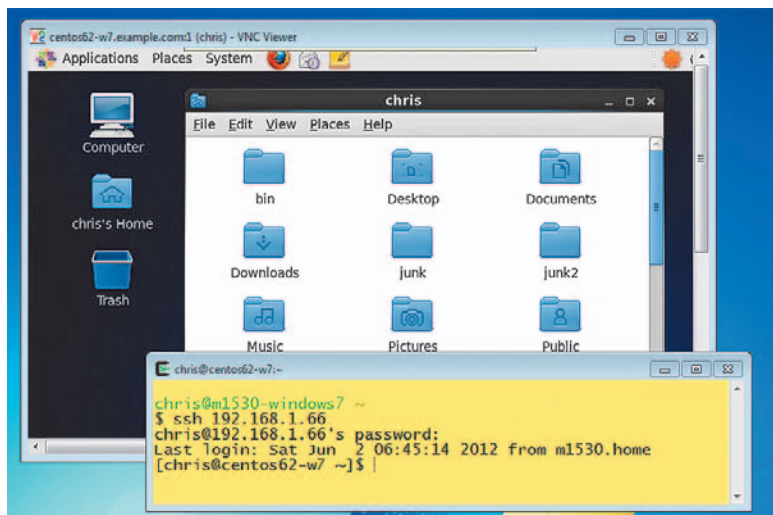
реждем к делу. В этой статье особой практики не будет – мы познакомимся с технологиями доступа к удаленным серверам и с несколькими утилитами, помогающими понять, что происходит. В следующих мы станем поактивнее и изучим классические приемы системного администрирования для управления пользователями, сервисами и сетями, а также для поддержания безопасности. Мы предполагаем, что у вас нет никакого опыта администрирования, и решительно призываем вас последовать за нами. Готовы? Первая проблема, с которой сталкивается большинство администраторов – как получить доступ к компьютеру, который нужно администрировать. У большинства студентов моих курсов физического доступа к ним не было. У многих из них были компьютеры с Windows и утилиты удаленного доступа.

Для доступа к командной строке серверов администраторы обычно пользуются безопасной оболочкой (SSH). В большинстве дистрибутивов Linux демон *sshd* установлен и активирован по умолчанию; часто это единственный сервис, который запускается сразу после установки сервера Linux. Если на клиентском компьютере (тем, за которым вы находитесь) установлен Linux, в нем почти наверняка установлен клиент SSH, поэтому у вас есть все необходимое.

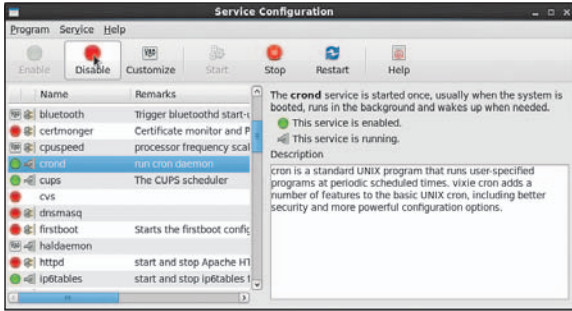
## Волшебство SSH

Если на клиентском компьютере – Windows, то по умолчанию клиент SSH скорее всего не установлен. В качестве клиента администраторы чаще всего пользуются *PuTTY* ([www.putty.org](http://www.putty.org)). Также есть *Cygwin*, прекрасная коллекция Linux-подобных утилит для Windows, в том числе и для SSH. Чтобы попробовать *Cygwin*, загрузите и запустите установщик (*setup.exe*) с сайта [www.cygwin.com](http://www.cygwin.com). При установке вы сможете выбрать, какие компоненты *Cygwin* установить – все сразу вам не нужны. *PuTTY* и *Cygwin* бесплат-

«Вы учитесь не на хирурга – недостатка тел для практики нет.»



Рабочий стол Windows 7 с двумя системами CentOS – в утилите просмотра *RealVNC* и клиенте SSH *Cygwin*.



➤ Типичная графическая утилита администрирования для управления запуском сервисов. В разных дистрибутивах они различаются.

ны. Я предпочитаю *Cygnin* потому что он больше похож на клиент SSH в Linux. Также есть *Bitvise Tunnelier* ([www.bitvise.com](http://www.bitvise.com)), который я только что описал.

## ...и графического рабочего стола

Если вы хотите подключаться к рабочему столу серверов, вариантов имеется несколько. Механизм управления окнами в Linux (окна X) – это клиент-серверная архитектура. Поэтому один из способов удаленного подключения к рабочему столу Linux – запустить графическую утилиту (клиент X) на компьютере Linux и подключиться к X-серверу, запущенному на настольной системе Windows. В течение многих лет популярным продуктом для Windows был сервер *Exceed X* от Hummingbird. Также есть *Xming* ([www.straightrunning.com/XmingNotes](http://www.straightrunning.com/XmingNotes)). Однако настроить все это в Linux не всегда просто.

## Настройка VNC

Альтернативный вариант – воспользоваться технологией подключения к удаленным рабочим столам, такой как *VNC*, что, как ни странно, означает «Виртуальные сетевые вычисления [Virtual Network Computing]». Для начала вкратце поясню, как она работает: обычно графические приложения взаимодействуют с X-сервером, который в свою очередь взаимодействует с монитором, мышью и клавиатурой компьютера (схему см. на рисунке внизу).

Чтобы воспользоваться *VNC*, нужно установить на удаленный компьютер сервер *VNC*. Для всех графических приложений, запущенных на этом компьютере, он выглядит как обычный X-сервер.

Но сервер *VNC* работает не с настоящими устройствами обработки графики, а с программным буфером кадра (на схеме – в середине). Любые изменения, производимые с этим буфером кадра, передаются по сети клиенту *VNC*, запущенному на локальном компьютере, за которым вы работаете (на схеме – справа). Данный процесс не слишком элегантен и требует компромисса между скоростью и качеством изображения, но это довольно сносное решение, к тому же оно кроссплатформенное.

Давайте настроим *VNC* на нашем компьютере с CentOS. Сперва посмотрим, установлен ли он:

```
# rpm -qa | grep vnc
```

Хм, никаких результатов; наверное, не установлен. Найдем нужный пакет:

```
# yum search vnc
```

```
gtk-vnc.i686 : A GTK widget for VNC clients
libvncserver.i686 : Library to make writing a vnc server easy
libvncserver-devel.i686 : Development files for libvncserver
tigervnc.i686 : A TigerVNC remote display system
tigervnc-server.i686 : A TigerVNC server
tigervnc-server-applet.noarch : Java TigerVNC viewer applet for TigerVNC server
tigervnc-server-module.i686 : TigerVNC module to Xorg
```

Я немного сократил вывод, но, похоже, недостает **tigervnc-server**. Быстрый поиск в Google по фразе “tigervnc centos 6” подтверждает это и даже дает нам некоторые указания по настройке. Давайте установим его:

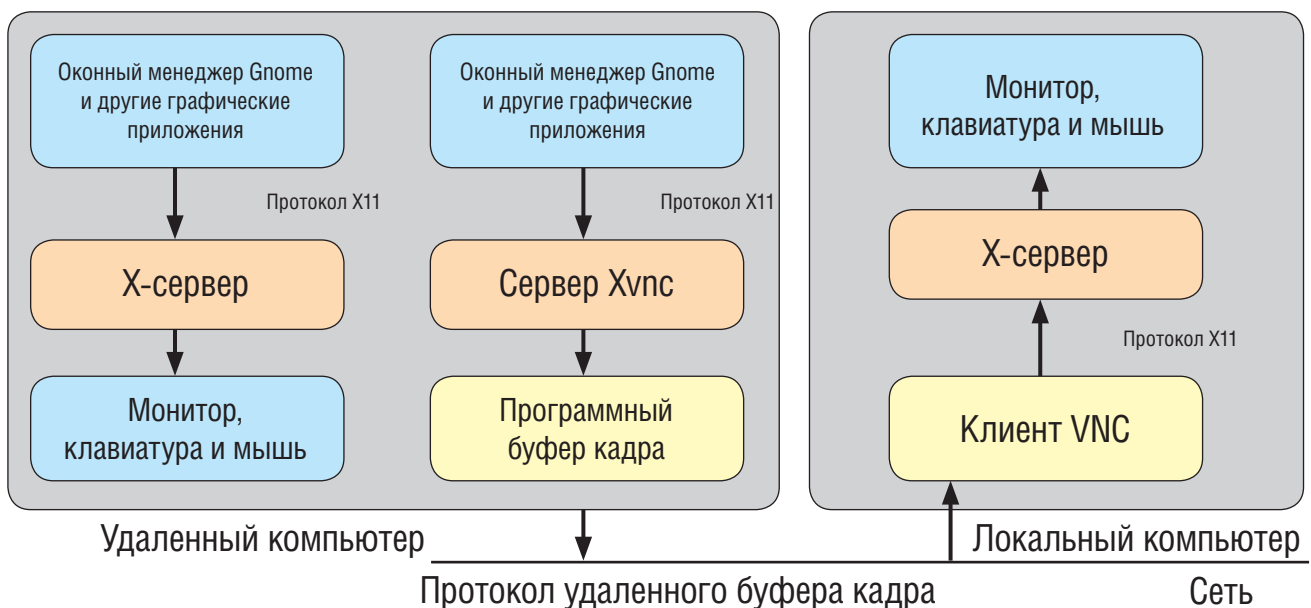
```
# yum install tigervnc-server
```

Теперь пакет установлен, и нужно сделать кое-какие настройки. Войдем в систему под обычным пользователем (не root) и зададим пароль для *VNC* командой **vncpasswd**. Пароль хранится в файле `~/vnc/passwd`. Обратите внимание, что он используется только для аутентификации сервером *VNC*. Он не имеет отношения к паролю для входа в Linux.

Теперь откройте файл `/etc/sysconfig/vncservers` и добавьте следующую строку, заменив “chris” на ваше имя пользователя:

➤ Сервер VNC на удаленном компьютере выглядит так же, как и обычный X-сервер, но передает содержимое буфера кадра обратно клиенту VNC.

## Как работает VNC



## Что вообще происходит?

Если вы хотите узнать...

... это можно сделать так

В каком каталоге я нахожусь?	Выполните команду <b>pwd</b> .
Под каким пользователем я работаю?	Выполните команду <b>id</b> . Она сообщит идентификатор пользователя, его членство в группах и (в зависимости от дистрибутива) ваш контекст безопасности SELinux.
Каков мой IP-адрес?	Выполните команду <b>ifconfig eth0</b> .
Где мои серверы DNS?	Загляните в файл <b>/etc/resolv.conf</b> .
Сколько места занято в моих файловых системах?	Команда <b>df</b> даст вам общую картину. Вам пригодится параметр <b>-h</b> для вывода цифр в удобопонятном виде. Для ответа на вопрос «Сколько места занимают файлы в <b>/var</b> ?» воспользуйтесь <b>du</b> .
Какие сервисы запускаются при загрузке системы?	Выполните команду <b>chkconfig --list</b> . Она выведет все сервисы системы и их уровни запуска.
Запущен ли сервис xxx?	Выполните команду <b>service xxx status</b> (где <b>xxx</b> – сервис, например, <b>cups</b> или <b>sshd</b> ).
Какие сервисы запущены в данный момент?	Есть несколько способов ответить на этот вопрос. Запустив команду <b>ps -e</b> , вы получите информацию обо всех запущенных процессах, но неясно, какие из них являются сервисами. Команда <b>lsdf -i</b> выведет те сервисы, которые слушают открытые порты TCP или UDP.
Какие сетевые порты открыты?	Как мы упомянули в предыдущем ответе, <b>lsdf -i</b> сообщит вам об открытых портах и приложениях, которые их слушают. <b>netstat -aut</b> отображает похожую информацию, но не сообщает, какие приложения используют порты.
Какие правила брандмауэра используются у меня?	Выполните команду <b>iptables -L</b> . (Интерпретировать ее вывод может быть не так-то просто.)
Какие пакеты у меня установлены?	Выполните команду <b>rpm -qa   less</b> (вывод команды будет объемным).*
Если пакет xxx установлен, как определить его версию?	Выполните либо команду <b>rpm -q xxx</b> (если знаете имя пакета), либо команду <b>rpm -qa   grep xxx</b> , если точное имя пакета неизвестно. Например, команда <b>rpm -q ssh</b> не сообщит ничего, потому что пакета <b>ssh</b> нет, зато команда <b>rpm -qa   grep ssh</b> выведет несколько похожих пакетов.*
Какую версию пакета xxx я установил?	Выполните команду <b>rpm -qi xxx</b> , где <b>xxx</b> – имя пакета.*

\* Только для Red Hat Enterprise Linux, SUSE и производных дистрибутивов

```
VNCSERVERS="1:chris"
```

Теперь запустите сервер VNC (от имени root):

```
# service vncserver start
```

Если вам нужна дополнительная справка, попробуйте прочитать статью по ссылке <http://wiki.centos.org/HowTos/VNC-Server> или посмотреть видеоролик [www.youtube.com/watch?v=rhEhHCeFnSo](http://www.youtube.com/watch?v=rhEhHCeFnSo).

Теперь сервер должен слушать порт 5901. Если позже вы будете добавлять новые серверы в **/etc/sysconfig/vncservers**, они будут слушать порты 5902, 5903 и т.д. Может возникнуть необходимость включить эти порты в брандмауэре или (если вы не знаете как это сделать) на время полностью отключить брандмауэр командой:

```
# iptables -F
```

Учтите, что с VNC нельзя получить удаленный контроль над существующим сеансом. Сеансы VNC полностью отделены от всего, что может происходить на «настоящем» рабочем столе сервера.

Теперь разберемся с клиентом VNC. Таковых существует множество. В Ubuntu попробуйте *Remmina*. В моей Ubuntu Precise Pangolin (12.04) он был установлен по умолчанию.

Если вам нужен клиент для Windows – их, кажется, писали все кому не лень. Некоторые из них бесплатны, некоторые нет. Попробуйте *RealVNC* с [www.realvnc.com](http://www.realvnc.com). Персональная версия (только клиента, не сервера) бесплатна.

Итак, теперь мы можем подключиться к серверу. Давайте оглядимся вокруг. В таблице выше приведены несколько классических утилит, с помощью которых администратор отвечает на вопросы из серии «Что происходит?». Подробности приведены на map-страницах этих команд.

## Через месяц

В следующем месяце мы узнаем, как задавать сетевые настройки наших компьютеров – IP-адреса, настройки маршрутизации, DNS, брандмауэров и др. **LXF**

Наши эксперты помогут вам с любым приложением Linux!



**ЕВГЕНИЙ БАЛДИН**  
Упомянул PostgreSQL на защите кандидатской диссертации по физике элементарных частиц.

## Отпустите музыку на волю

В том-то и прелесть музыки – ее не могут конфисковать. «Побег из Шоушенка»

И могут, и конфискуют. Спасение одно: проигрывать музыку только в своей голове, да и то пока некто не обучился телепатии. Я всемерно за достойное вознаграждение тяжкого труда авторов и исполнителей, но действия организаций, узурпировавших права по защите чести и достоинства авторов, мягко говоря, огорошивают. Кипучая жадность РАО заслоняет все разумные основания идеи защиты творцов. Нам следует отгородиться от этой клоаки и создать культурное окружение с нуля, без лазеек для этой заразы.

Кимико Исидзакэ [Kimiko Ishizaka] начала гастроль с целью подготовки к созданию свободной записи «Хорошо темперированного клавира» Баха. Этот ее проект, как и прошлый («Вариации Гольдберга», музыка и ноты, переданные в общественное достояние под лицензией Creative Commons Zero), частично финансирован через Kickstarter. В 2010 году на Kickstarter успешно отметился проект Musopen, собрав в 6 раз больше средств, чем изначально запросил основатель проекта Аарон Данн [Aaron Dunn]. Музыка, записанная в рамках этого проекта, выложена как общедоступная [Public Domain] на [archive.org](http://archive.org), без копирайт-ограничений. Инструментальную часть обеспечил Чешский национальный симфонический оркестр. А вот это всего лишь начало более светлого мира...

**P.S.** Связь с GNU/Linux тут есть: его окружение применимо для проигрывания музыки © Humble Music Bundle. [E.m.Baldin@inp.nsk.su](mailto:E.m.Baldin@inp.nsk.su)

## В этом месяце вы научитесь...



**Рулить VirtualBox ..... 62**  
**Шашанк Шарма** показывает, как протестировать новое программное обеспечение, не грохнув свою основную систему.



**Кодировать на RPi ..... 66**  
С этой целью **Джонатан Робертс** приглашает на ваш Raspberry Pi *Guido van Robot*. Гоняйте Гвидо с угла на угол, подбирая маячки.



**Шифровать секреты ..... 70**  
Позвольте **Нейлу Ботвику** отвести шпионские взгляды от ваших данных – он вам покажет, как использовать TrueCrypt.



**Учиться с Moodle ..... 72**  
**Бен Эверард** помогает всем учителям экономить деньги и поставлять качественные курсы обучения с Moodle.



**Делиться файлами ..... 76**  
Надо обобществить некоторые файлы? **Майк Сондерс** покажет, как тут помогает FreeNAS.



**Болтать онлайн ..... 80**  
Сговориться со своими коллегами по Arduino и контролировать их отовсюду – мечта **Ника Вейча**.



**Вникать в Erlang ..... 84**  
**Андрей Ушаков** задаётся сакраментальным вопросом: как пройти в библиотеку?



**Сканировать Сеть ..... 88**  
Но прислушайтесь к совету **Игоря Штомпеля**: работать только в рамках действующего законодательства!

## Пропустили номер?

Узнайте на с. 104, как получить его прямо сейчас!



# VirtualBox: ПК за

Находясь в творческом отпуске от Хогвартса, профессор **Шашанк Шарма** демонстрирует, как создавать компьютеры из воздуха.



**Наш эксперт**

**Шашанк Шарма** пишет о свободном ПО уже более четырех лет для различных изданий, включая [Linux.com](http://Linux.com). Он соавтор книги *Beginning Fedora*.

**З**десь, в Башнях **LXF**, *VirtualBox* занимает особый уголок в наших сердцах. Он позволяет нам запросто тестировать все: от альфа-версий программ до пробных версий дистрибутивов, не подвергая наши рабочие компьютеры опасностям, обычно исходящим от таких программ. Для этого он создает изолированные экземпляры виртуальных машин (VM) на наших физических устройствах.

И пусть термин «виртуализация» вас не пугает. Благодаря *VirtualBox* виртуализация доступна не только крупным коммерческим компаниям. Программа обладает приятным графическим интерфейсом, в котором можно создавать и запускать виртуальные машины в мгновение ока – ну, почти. В отличие от систем с двойной или мультизагрузкой, где каждая система устанавлива-

ется в отдельный раздел жесткого диска, с *VirtualBox* можно установить несколько дистрибутивов прямо в главной ОС. Это позволяет избежать разбиения диска на разделы и пробовать новые дистрибутивы и операционные системы в привычных условиях.

*VirtualBox* также умеет сохранять так называемые «снимки [snapshots]» виртуальных машин. Они очень удобны для восстановления виртуальных машин, поврежденных некорректным ПО. Это очень помогает нам в редакции **LXF**, но будет не менее полезно пользователям, желающим попробовать экспериментальное ПО. Прежде чем пробовать новую программу, просто сохраните снимок рабочей виртуальной машины. Если программа посеет хаос в виртуальной машине, вы сможете перейти к предыдущему состоянию всего за один щелчок.

А главное, *VirtualBox* поддерживает уйму различных устройств, и ими можно полноценно пользоваться в виртуальных машинах – например, запускать их с родным разрешением монитора и даже ускоренной 3D-графикой. Программа также может пользоваться командами виртуализации, встроенными в современные процессоры для поддержки производительности виртуальных машин.

## Установка VirtualBox

В большинстве популярных дистрибутивов *VirtualBox* можно найти в репозиториях, за исключением некоторых компонентов, доступных по проприетарной лицензии – например, поддержки устройств USB 2.0. Но лучше всего скачать последнюю версию *VirtualBox* с сайта [www.virtualbox.org/wiki/Downloads](http://www.virtualbox.org/wiki/Downloads). Там есть ссылки на двоичные файлы для различных поддерживаемых операционных систем.

Для Linux есть другие пакеты, формат которых зависит от дистрибутива. Чтобы найти нужный пакет, прокрутите список вниз и выберите установщик, который запускается на «Всех дистрибутивах [All distributions]» и соответствует вашей архитектуре. Так, текущий релиз для 32-битной системы – **VirtualBox-4.1.16-78094-Linux\_x86.run**.

На странице загрузок также есть ссылка на пакет расширений для текущего релиза. В отличие от самого релиза, есть только один пакет расширений, который работает на всех платформах.

Перед установкой *VirtualBox* убедитесь, что установлен пакет **dkms**, используемый для генерации модулей ядра, необходимых *VirtualBox*. Он есть в репозиториях большинства дистрибутивов. Пользователи Ubuntu могут установить его командой **sudo apt-get install dkms**, пользователи Fedora – командой **yum install dkms**, а пользователи Mageia – командой **urpmi dkms**.

Для установки *VirtualBox* откройте терминал и наберите

```
$ sudo ./VirtualBox-4.1.16-78094-Linux_x86.run
```

Скрипт создаст необходимые модули *VirtualBox* и установит программу в каталог **/opt/VirtualBox**. После установки *VirtualBox* установите пакет расширений. Откройте *VirtualBox* и выберите пункт меню File > Preferences [Файл > Настройки], затем переключитесь в раздел Extensions (Расширения). Нажмите на кнопку Add Package [Добавить пакет] справа и выберите загруженный пакет расширений.



» С помощью клонирования удобно перемещать виртуальные машины в другие основные ОС.



# НОЛЬ КОПЕЕК

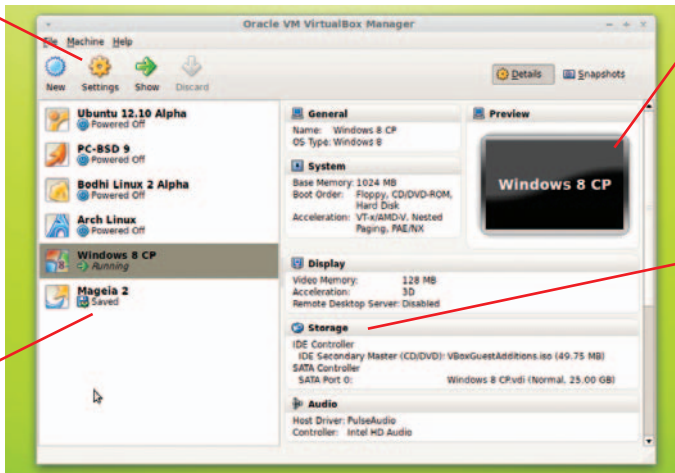
## Исследуем интерфейс главного окна VirtualBox

### Настройки

Нажмите на кнопку Settings [Настройки] для настройки выбранной виртуальной машины.

### Установленные VM

Список установленных виртуальных машин и их состояние.



### Предпросмотр

Текущий вид запущенной виртуальной машины в миниатюре.

### Хранилище

Щелкните здесь, а затем на иконке компакт-диска, чтобы загрузиться с привода оптических дисков главного компьютера или с ISO-образа.

### Скорая помощь

VirtualBox можно запускать и на компьютерах с более старыми процессорами, без поддержки аппаратной виртуализации.

## Настройка виртуальной машины

Установив и запустив *VirtualBox*, откройте мастер создания новой виртуальной машины и пройдите по нему. Как вы увидите, у виртуальной машины, как и у физического компьютера, есть несколько настраиваемых компонентов.

С помощью информации, указанной при создании виртуальной машины, *VirtualBox* сформирует готовую к запуску виртуальную машину с выделенными вами ресурсами. Но кроме этого, нужно задать еще несколько базовых настроек. Первое и самое главное – измените приоритет загрузки для виртуальной машины. По умолчанию виртуальные машины сначала ищут загрузочный диск в дисковом, затем в приводе оптических дисков, и только потом загружаются с жесткого диска.

Для изменения порядка загрузки в *VirtualBox* выберите виртуальную машину и нажмите на кнопку Settings [Настройки].

Зайдите в раздел System [Система] и переберитесь на вкладку Motherboard [Материнская плата]. Порядок загрузки будет указан в разделе Boot Order [Порядок загрузки].

Чтобы на несколько секунд ускорить загрузку, можно удалить ненужные загрузочные устройства, например, дисковод гибких дисков, сняв галочку рядом с ними. Если поставить эту галочку рядом с каким-то устройством, например, сеть, оно будет добавлено в список загрузочных устройств. С помощью стрелок вверх и вниз можно перемещать выбранное устройство в списке загрузочных устройств.

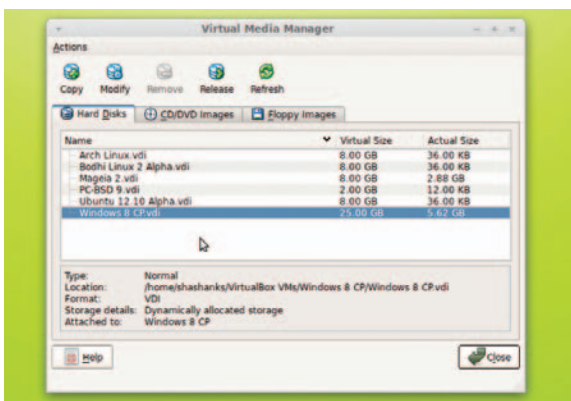
Если вы запускаете виртуальную машину на многоядерном компьютере с аппаратными расширениями, ей можно передать несколько ядер. Перейдите на вкладку Processor [Процессор] и выделите виртуальной машине несколько процессоров. Там также есть галочка PAE/NX, выбор которой позволит выдать виртуальной машине более 4 ГБ оперативной памяти.

По умолчанию, сетевой адаптер виртуальной машины подключается к реальной сети на главном компьютере через NAT. Важное следствие здесь таково, что виртуальные машины не будут напрямую доступны другим компьютерам сети. Но это легко исправить. Перейдите в раздел Network [Сеть], и на вкладке Adapter 1 [Адаптер 1] выберите в выпадающем списке Attached To [Подключен к] пункт Bridged Adapter [Режим моста]. На главном компьютере нужно выбрать физический сетевой адаптер.

Если вы хотите добавить несколько виртуальных адаптеров, перейдите на вкладку Adapter 2 [Адаптер 2] и установите галочку Enable Network Adapter [Включить сетевой адаптер]. Можно добавить до четырех сетевых адаптеров. После изменения настроек виртуальной машины для ее загрузки нажмите кнопку Start [Запуск] в главном окне *VirtualBox*. При первом запуске виртуальной машины *VirtualBox* запускает мастер первого запуска [First Run wizard], который помогает вам выбрать носитель, с которого

### Скорая помощь

Вообще говоря, чем больше памяти выделить виртуальной машине, тем быстрее она будет работать. Но оставьте не меньше половины физической памяти основной ОС.



» Virtual Media Manager управляет всеми жесткими дисками и образами CD/DVD, используемыми виртуальной машиной.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

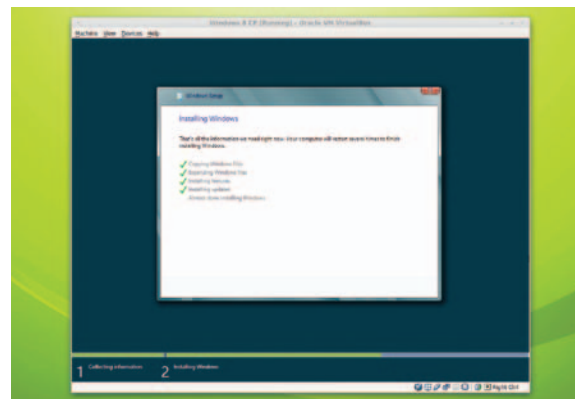
должна загружаться виртуальная машина, так как на подключенном виртуальном жестком диске еще нет операционной системы.

## Установка ОС в VM

Установка ОС на VM не отличается от установки на обычный компьютер. Вставьте установочный диск в привод CD/DVD и выберите его в мастере первого запуска. Однако одно из важнейших достоинств *VirtualBox* в том, что он может загружаться прямо с ISO-образов.

Дистрибутивы Linux поставляются в виде ISO-образов, и иногда мы записываем их в этом формате на **LXFDVD**. Загрузиться напрямую с них нельзя: чтобы создать загрузочный диск из образа, потребуется программа прожига дисков. Но в *VirtualBox* можно просто выбрать ISO-образ в мастере первого запуска, и он будет смонтирован в виртуальную машину как физический диск в виртуальном приводе.

После выбора загрузочного носителя при запуске виртуальной машины *VirtualBox* предупредит, что включена опция автоматического перехвата клавиатуры [Auto Capture Keyboard]. Это означает, что весь ввод с клавиатуры будет перенаправляться на запущенную виртуальную машину. Похожее сообщение появляется, когда вы первый раз щелкаете мышью в виртуальной машине – оно предупреждает, что теперь мышь будет работать только внутри виртуальной машины. Чтобы отключить эти устройства от виртуальной машины и воспользоваться ими в главной системе, нажмите клавишу Host – она отпускает эти устройства из управления виртуальной машины. По умолчанию клавиша Host – это правая клавиша Ctrl. Ее можно изменить, выбрав пункт меню File > Preferences > Input [Файл > Настройки > Ввод] в главном окне *VirtualBox*. После включения виртуальной машины она считает загрузочный диск, и все будет точно так же, как и на настоящем компью-



► Некоторые гостевые ОС, такие как Windows 8, запустятся только на процессорах с расширениями виртуализации.

тере. Отличие только в том, что вместо настоящих устройств установщик обнаружит виртуальные, например, жесткий диск **Ubuntu 12.10 alpha.vdi**.

Но, несмотря на свое название, он ведет себя как настоящий жесткий диск, и перед установкой системы его нужно отформатировать и разбить на разделы. Если вы не слишком уверенно владеете разбивкой на разделы в Linux, попрактикуйтесь в виртуальной машине, потому что физический диск тут ничем не рискует.

## Установка гостевых дополнений

Гостевые дополнения – специальные программные пакеты с драйверами устройств, призванные оптимизировать производительность гостевой ОС. Гостевые дополнения обладают множеством преимуществ, и хотя устанавливать их необязательно, советуем все же это сделать.

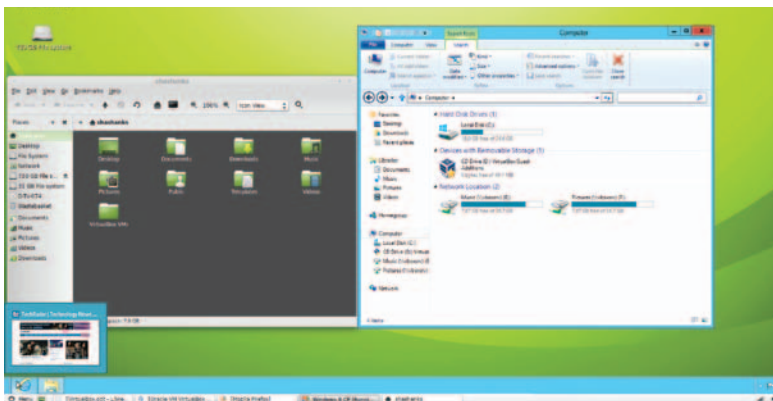
Для начинающих: вы сможете свободно перемещать мышь между гостевой и основной ОС без нажатия клавиши Host. Вы также получите лучшую поддержку видео с 3D-ускорением и более высокими разрешениями, а также возможность монтировать каталоги основной системы в гостевую как сетевые папки.

Гостевые дополнения поставляются в виде файла ISO вместе с *VirtualBox*, и устанавливать их нужно из виртуальной машины. Файл **VboxGuestAdditions.iso** содержит дополнения для всех гостевых ОС. Перед установкой гостевых дополнений нужно подготовить гостевую систему, установив пакет **dkms** из ее репозитория, точно так же как в главной системе. Затем в меню виртуальной машины выберите Devices > Install Guest Additions [Устройства > Установка гостевых дополнений]. При этом ISO-образ с гостевыми дополнениями автоматически смонтируется в виртуальный

### Скорая помощь

Для нажатия специальных сочетаний клавиш, таких как Ctrl+Alt+Del, в виртуальной машине воспользуйтесь меню Machine запущенной VM.

► С помощью функции **Seamless Windows** *VirtualBox* может отображать окна виртуальной машины в главной системе.



## Создание снимков VM

В *VirtualBox* можно делать моментальные снимки виртуальной машины на различных этапах и переключаться между ними. Это незаменимая возможность в нашей работе, куда входит тестирование программ, способных вызвать сбой в системе.

Сделать снимок довольно просто. Если виртуальная машина запущена, выберите пункт меню Machine > Take Snapshot [Машина > Сделать снимок]. Если же она выключена, перейдите на вкладку Snapshots [Снимки] в главном окне *VirtualBox* и кликните по иконке с камерой. В обоих случаях появится окно,

где нужно ввести имя снимка, а также необязательное описание. После сохранения снимка с виртуальной машиной можно работать как обычно. Состояние системы из снимка можно восстановить в любой момент, кликнув правой кнопкой на снимке и выбрав пункт меню Restore Snapshot [Восстановить снимок]. Помните, что если система восстанавливается к более раннему снимку, все изменения, произошедшие с момента его снятия, будут утеряны. Поэтому перед возвратом к более раннему снимку советуем сделать снимок текущей системы.



► Снимок сохраняет все настройки VM, а также полное содержимое виртуальных жестких дисков.

» Пропустили номер? Узнайте на с. 104, как получить его прямо сейчас.

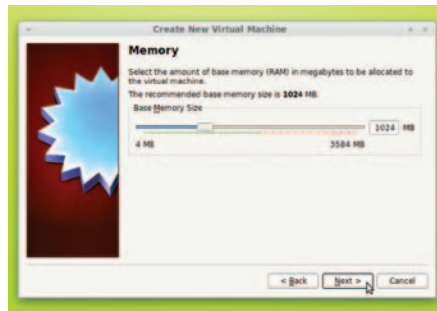


## Шаг за шагом: Создаем виртуальную машину



### 1 Имя и тип ОС

Нажмите на кнопку New [Новая] для запуска мастера создания новой виртуальной машины [Create New Virtual Machine wizard]. Указав имя виртуальной машины, выберите гостевую ОС, которую вы на нее установите, а также ее версию.



### 2 Основная память

Затем нужно указать объем оперативной памяти, выделяемой виртуальной машине. В зависимости от типа гостевой ОС *VirtualBox* укажет минимальный размер ОЗУ.



### 3 Виртуальный жесткий диск

Создайте виртуальный жесткий диск. Для этого есть собственный мастер, в котором нужно выбрать размер виртуального диска. Другие параметры можно оставить без изменений.

привод компакт-дисков. Если у вас виртуальная машина с Windows, то автоматически запустится установщик гостевых дополнений. В Linux придется запустить терминал, перейти в каталог, где смонтирован диск (обычно `/media`), и скопировать:

```
$ sudo ./VBoxLinuxAdditions.run
```

Когда библиотеки будут установлены, перезапустите гостевую ОС, чтобы включить гостевые дополнения.

## Изменение настроек монитора

После установки гостевых дополнений можно настроить виртуальные компоненты для наилучшего использования реальных устройств. Начнем с настроек монитора. По умолчанию большинство гостевых ОС будут работать с меньшим разрешением, чем поддерживает ваш монитор.

Чтобы запустить гостевую ОС в полноэкранном режиме, сначала нужно выделить ей больше видеопамати. Для этого зайдите в раздел Display [Монитор] в меню Settings [Настройки] виртуальной машины. Перетащите ползунок Video Memory [Видеопамать], чтобы увеличить размер памяти виртуальной видеокарты, доступный данной виртуальной машине.

Также можно попросить *VirtualBox* передать продвинутое возможности вашей видеокарты с 3D-ускорением виртуальной машине, просто установив галочку Enable 3D Acceleration [Включить 3D-ускорение]. В отличие от увеличения размера видеопамати, 3D-ускорение будет включено только для тех гостевых ОС, у которых установлены гостевые дополнения. Одно из лучших достоинств гостевых дополнений – возможность подключать каталоги

главной ОС в гостевую как сетевые папки, при этом никаких настроек сети ни в одной из ОС выполнять не нужно.

Для настройки разделяемых каталогов откройте окно Settings [Настройки] гостевой ОС и зайдите в раздел Shared Folders [Общие папки]. Здесь вы видите список разделяемых каталогов и их монтирование в гостевой ОС. Справа от этого списка есть несколько иконок. Для добавления общей папки кликните по иконке каталога с плюсом. Откроется окно Add Share [Добавить разделяемый каталог]. В нем нужно указать путь к каталогу главной ОС, к которому нужно предоставить общий доступ. Также можно изменить имя, под которым он будет отображаться в гостевой ОС.

По умолчанию, если не выбрать галочку Read-only [Только для чтения], у вас будет доступ на чтение и запись к каталогу из гостевой системы. При выборе галочки Automount [Автомонтирование] каталоги будут монтироваться автоматически при загрузке гостевой системы. Если сделать это в гостевой ОС Windows, то разделяемым каталогам будет автоматически назначена первая свободная буква диска, например, `G:\`. В гостевой ОС Linux такие папки монтируются в каталог `/media` и имеют префикс `sf_`, например, `/media/sf_pictures`.

Материала нашего урока достаточно для большинства применений *VirtualBox* – от тестирования программ до испытания новых операционных систем и дистрибутивов. Но это лишь малая часть способностей этой невероятно гибкой программы. Например, в *VirtualBox* есть расширенная командная строка под названием *VBoxManage*, которая позволяет полностью контролировать все аспекты работы программы. **LXF**

### Скорая помощь

Сменить съемный носитель в запущенной виртуальной машине можно с помощью меню Devices [Устройства].

### Скорая помощь

Для доступа к общим папкам позаботьтесь, чтобы пользователь гостевой ОС являлся членом группы vboxsf.

## Терминология виртуализации

» **Виртуальная машина** Автономная среда, состоящая из различных виртуальных устройств. Компоненты виртуальной машины можно настраивать точно таким же образом, как и на настоящем компьютере, но вместо отвертки вы будете употреблять мышь.

» **Главная ОС** Операционная система на физическом компьютере, где находится *VirtualBox*. Наряду с Linux, это может быть любая версия Windows или Mac OS X.

» **Гостевая ОС** Операционная система внутри виртуальной машины. *VirtualBox* официально поддерживает несколько популярных дистрибутивов Linux, Windows и Mac OS X, но в качестве гостевой можно установить практически любую ОС.

» **OVF** Open Virtualisation Format [Открытый формат виртуализации] используется для импорта и экспорта готовых к работе виртуальных машин. Он кроссплатформенный и поддерживает несколько продуктами виртуализации, в т.ч. *VirtualBox*.

» **VDI** Virtual Disk Image [образ виртуального диска] – родной формат файла для виртуальных дисков, создаваемых *VirtualBox*. Кроме VDI, поддерживаются форматы VMMDK от VMware, VHD от Microsoft и формат более старых жестких дисков Parallels.

» **Динамически выделяемые диски** Один или два типа файлов виртуальных дисков, поддерживаемых *VirtualBox*. Пространство на физическом диске используется только при его заполнении до максимальной емкости, указанной при создании диска.

# Raspberry Pi.

Raspberry Pi был создан, чтобы научить детей программированию. **Джонатан Робертс** представляет идеальную среду для начинающих: *Guido van Robot*.



**Наш эксперт**

Супер-скрытный **Джонатан Робертс** копается в Linux с детских лет. Иными словами, не так уж долго...

**В** двух наших первых руководствах по Raspberry Pi мы дали вам разгон. Мы объяснили, как настроить RPi и убедиться, что все работает нормально. Мы также познакомили вас с основными навыками работы в Linux, включая управление пакетами и удаленный доступ к SSH и VNC. Если у вас только появился Raspberry Pi и вам не терпится начать, вы сможете найти эти два руководства на диске этого месяца.

На этом уроке мы хотим сделать нечто иное. Мы рассмотрим Raspberry Pi в качестве платформы для обучения программированию, а для этого воспользуемся приложением под названием *Guido van Robot* (сокращенно – *GvR*).

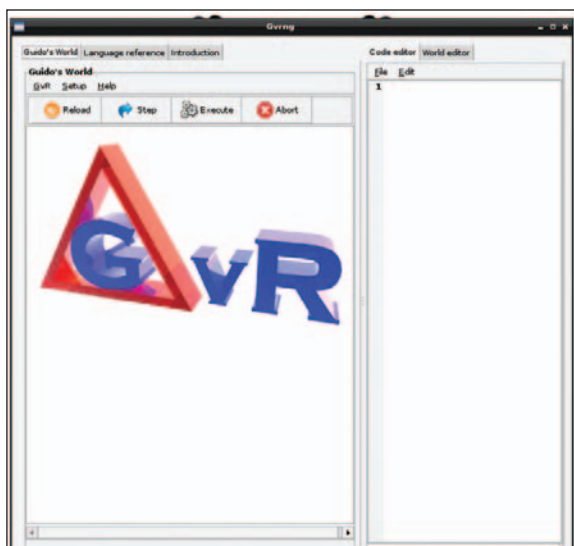
## Черепашки и роботы

Если вы выросли в Великобритании и учились в школе в компьютерную эпоху BBC, вы, возможно, помните черепашек и LOGO, несложный язык программирования, в котором было всего несколько команд. У черепашки была ручка для рисования разных фигур, и с помощью правильных команд в LOGO вы могли нарисовать любую фигуру (существовали даже физические версии, которые разгуживали по столам и полам).

Нарисовать квадрат в LOGO можно было командами

```
FORWARD 100
LEFT 90
FORWARD 100
LEFT 90
FORWARD 100
LEFT 90
FORWARD 100
LEFT 90
```

Этот язык программирования учил детей основам – как разбивать решение проблемы на шаги, необходимые для ее решения, то есть мыслить в виде процедур. Именно этому мы обучим вас на нашем уроке, только вместо LOGO у нас будет *Guido van Robot*.



➤ Это *GvR*, ваше окно в мир Guido. Слева вы увидите Guido, а справа сможете управлять его действиями.

Прежде чем приступать к чему бы то ни было, установите *GvR* на ваш Raspberry Pi, чтобы вы могли следовать инструкциям нашего руководства. Это делается запуском *LXTerminal* из меню приложений и вводом

```
sudo apt-get install gvrng pythongtksourceview2
```

Если, взглянув на эту команду, вы решили, что она бессмысленна, загляните во второе руководство по Raspberry Pi на диске: «Строим NAS».

## Знакомьтесь: Guido

Установив *GvR*, можете запускать его из меню приложений в нижней левой части экрана – оно находится ниже подменю Education. В появившемся окне вы увидите место для Guido и его мира в левой части, и место для инструкций, которые вы будете ему давать, и для создания его мира в правой части.

Давайте начнем со знакомства с миром Guido. Первое, что надо сделать – скопировать файлы в папке **GvR** на диске этого месяца в ваш Raspberry Pi. Можете использовать для этого любой удобный для вас способ, но самым простым будет USB-брелок, если вы используете для доступа SSH или VNC, поскольку в этом случае один из портов USB остается свободным. Затем переключитесь на вкладку World Editor в правой части окна, выберите File > Open [Файл > Открыть] и **GvR1.wld**. Левая часть мира заполнится изображениями, представляющими мир Guido, а сам Guido символизируется треугольником с буквой G.

Мир Guido похож на наш (ну, на американский город точно похоже): он состоит из улиц (протянувшихся с востока на запад) и авеню (идущих с севера на юг), и окружен крепкими внешними стенами, через которые Guido не может перебраться. На пересечении каждой улицы и авеню располагается угол.

В этом мире есть еще пара вещей. Веерер – это такой маленький маячок-пищалка, который можно поместить на углу. Он издает сигналы, которые Guido может слышать, только находясь на том же углу. Имеются также внутренние стены – они, как и внешние, для Guido непроходимы. В мире *GvR1*, помимо Guido, есть единственный маячок, представленный голубым кружком.

## Первые шаги

Guido – не самый продвинутый робот. Он умеет выполнять только пять действий:

- \* move [двигаться]
- \* turnleft [повернуть налево]
- \* pickbeeper [взять маячок]
- \* putbeeper [поставить маячок]
- \* turnoff [выключить]

Команда **move** заставляет Guido двигаться по единственной улице или авеню вперед, а команда **turnleft** заставляет его повернуться на 90 градусов влево. Если вы, например, хотите, чтобы Guido передвинулся вперед на три авеню, вам нужно позаботиться, чтобы он смотрел в нужном направлении (задав нужное число в командах **turnleft**) и затем задать три команды **move**.

В этом первом мире все, что нам от вас нужно, ограничивается перемещением Guido к маячку, возможностью его взять,

# Рудим Guido

переместить его на один угол дальше и затем выключить его. Это означает, что Guido остановится на углу 4-й авеню и 2-й улицы.

Чтобы сделать это, переключитесь со вкладки World Editor на вкладку Code Editor. Затем можете ввести любую комбинацию из вышеприведенных команд в текстовое поле справа, поставив каждую из них на собственную строку, чтобы Guido переместился к маячку и взял его. Когда у вас будет набор команд, который, по вашему мнению, должен будет сработать, нажмите на Execute и посмотрите, что произойдет.

Помните, что Guido может слышать маячок, только находясь на одном углу с ним – а команда `pickbeeper` срабатывает только тогда, когда маячок есть в наличии. Если вы попытаетесь использовать `pickbeeper` при отсутствии маячка, то получите сообщение об ошибке, и вам придется перезапустить программу, что делается посредством нажатия Reload.

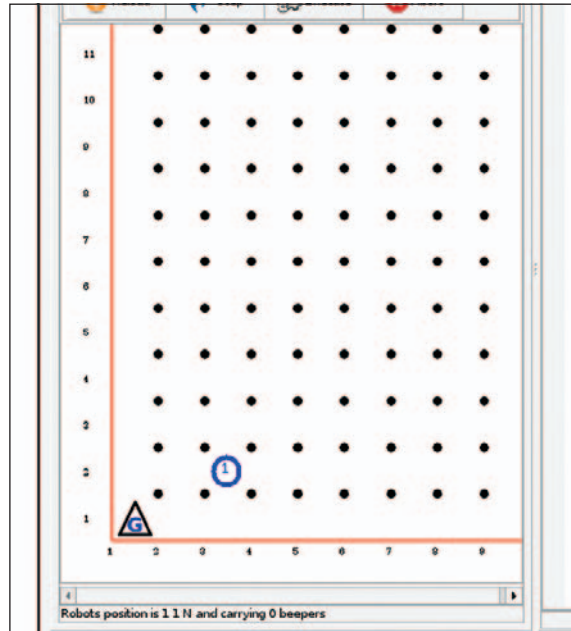
## Компактнее — удобнее

Как дела? Если все сработало, значит, у вас должен получиться примерно такой набор инструкций:

```
move
turnleft
turnleft
turnleft
move
move
pickbeeper
move
turnoff
```

Самая хитрая часть этой проблемы – повернуть направо. Поскольку Guido не знает, как это сделать, вам придется осознать, что один поворот направо равен трем поворотам налево.

Это довольно громоздко и затрудняет чтение нашей программы: если бы у нашей программы была проблема – например, Guido неправильно повернулся и пришел к стене, от которой он и начал – было бы трудно понять, что подразумевалось под всеми этими `turnleft` и что один был явно лишним. Это также означает, что нам придется писать одно и то же много раз, а поскольку мы – ленивые программисты, нам это совсем не нравится!



» Наша первая проблема: можете ли вы заставить Guido переместиться к маячку и забрать его?

К счастью, Guido не так глуп, как кажется на первый взгляд. Хотя на данный момент он знает только пять команд, он быстро учится, и он готов, чтобы мы научили его чему-то новому на основе того, что ему уже известно. Иными словами, мы можем сделать подборку инструкций, таких, как три `turnlefts`, дать ей название, и Guido будет исполнять ее каждый раз, когда мы введем это название.

## Схватывает на лету

Чтобы это продемонстрировать, и чтобы сделать наш код более легким для чтения, менее ошибочным и более быстрым для ввода, давайте научим Guido поворачивать направо. Для этого нам надо будет использовать новый вид инструкции:

```
define turnright:
turnleft
```

»

## Сотворение мира

Непростую задачку мы поставили перед вами в этой статье, а? Или вы хотите создать собственные гадкие миры для своих друзей по программированию *GvR*? Тогда вам повезло, поскольку *GvR* как раз и позволяет создавать индивидуальные миры, и для этого у вас есть целых два способа.

Первый – интерактивный метод `point-and-click` [наведи-и-щелкни], который запустится, если выбрать *GvR* > *Open World Builder* на левой панели. Появится пустой мир, окрашенный в голубой цвет.

Можете добавлять стены, щелкнув левой кнопкой там, где вы хотите их разместить, или маячки, щелкнув правой кнопкой по пустому углу и введя желаемое количество. Если хотите отрегулировать начальные параметры Guido, например, его расположение, направление или количество маячков, которые он несет, щелкните средней кнопкой по любому месту в мире и введите нужную информацию.

Вся эта информация хранится в текстовом файле и подобна инструкциям, которые вы даете Guido

в этом руководстве. Вы можете просмотреть содержание этого файла для любого мира, который вы открываете, щелкнув по вкладке World Editor в правой стороне. Описание робота задает начальные настройки Guido: улицу, авеню, направление и количество маячков. Каждое описание стены определяет угол, где появится стена, и направление, которое она блокирует, а каждое определение маячка выделяет угол, где будет располагаться некое количество маячков.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

```
turnleft
turnleft
```

Первая строка сообщает Guido, что мы определили новую инструкцию под названием **turnright**. Строки с отступом, следующие за двоеточием – это уже существующие команды в том порядке, в котором они должны выполняться, когда мы хотим, чтобы Guido повернул направо.

Отступ играет важную роль. Все строки с отступом считаются частью определения **define**. Как только вы введете строку без отступа, GvR будет считать, что введенное больше не является частью определения.

Чтобы использовать это в программе, вам нужно поставить определение наверх, а все остальные команды разместить внизу, таким образом:

```
define turnright:
turnleft
turnleft
turnleft
move
turnright
move
```

Попробуйте переписать нашу первую программу, но используйте инструкцию **turnright** вместо трех **turnleft**. Разве не проще стало читать?

## Повторяемся

К этому моменту вы освоили первый мир, так что давайте перейдем к чему-нибудь поинтереснее. Переключитесь снова на вкладку World Editor на правой стороне и выберите File > Open, на сей раз открыв файл **GvR2.wld**. Это похожий мир, за исключением того, что здесь нет стены позади маячка, и все расстояния увеличены

(возможно, вам придется расширить окно и перетащить линию, разделяющую панели мира и редактора, чтобы увидеть маячок).

Цель та же: Guido должен переместиться к маячку, взять его, переместить на один угол дальше и выключить. Мы можем сделать это точно так же, как раньше, отдельно указывая каждое движение и каждый поворот, но, поскольку сейчас все находится на большем расстоянии, это будет выглядеть так же путано, как раньше все эти многочисленные **turnleft**.

К счастью, Guido знает еще несколько команд, которые несколько облегчат нашу задачу. Первая из тех, которые мы собираемся рассмотреть – команда **do**. Выглядит она так:

```
do <number>:
<instructions>
```

Как видите, по формату она подобна команде **define**. Сначала мы сообщим Guido, что это команда **do**, созданная для повторения любых **<instructions>** [инструкций] указанное **<number>** [количество] раз. В **GvR2** мы можем использовать ее, чтобы начать нашу программу так:

```
do 4:
move
```

Это намного быстрее, чем вводить четыре отдельных команды **move**. Более того, если нам нужно выполнить 100 перемещений, на это потребуется ровно такое же количество инструкций. Попробуйте написать оставшуюся часть программы для **GvR2** с помощью отдельных команд **move**, а потом с помощью команд **do**, и обратите внимание, насколько последняя короче.

## Время принятия решений

Еще одна способность Guido – принимать решения. Например, Guido может спросить: «На этом углу есть маячок?» и решить исполнять **pickbeepers** только при его наличии. Или он может спросить, есть ли перед ним стена, и двинуться вперед, только если путь свободен. Формат для этого типа команды таков:

```
if <condition>:
<instructions>
```

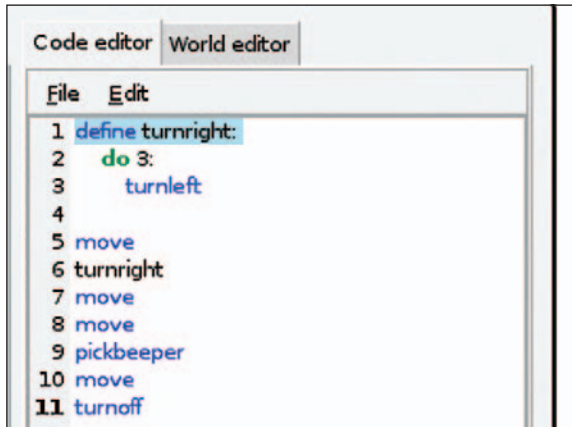
Здесь вам все знакомо, за исключением условия. Это – вопрос, который задает Guido, и есть подборка вопросов, которые он умеет задавать. Их 18, и все они перечислены во вкладке Language Reference интерфейса **GvR**.

Вы увидите, что все они являются вопросами типа Да/Нет, с ответами **true** [истинно] или **false** [ложно] – это единственный тип вопросов, с которыми Guido (и большинство компьютеров) могут работать.

Вот пример того, как можно написать код, который обеспечит движение вперед только при отсутствии стены перед Guido.

```
if front_is_clear:
move
```

➤ Программируя Guido, каждую инструкцию размещайте на новой строке. Следите, чтобы у каждой части сложной инструкции был отступ.



## Программирование для удовольствия

*Guido van Robot* – не единственная имеющаяся возможность для подающих надежды программистов, которым не терпится сделать свои первые шаги в Raspberry Pi. Образ Debian на SD-card, на котором мы основываем наши руководства, также идет со Scratch, графическим языком программирования, который работает, соединяя разные компоненты программы вместе.

Он использует многие из тех же основных процессов управления, о которых мы рассказали

в этом руководстве, в том числе условия и циклы, но вводит также и множество новых функций. Это позволит вам выполнять самые разные действия, в т.ч. заставить графику отвечать на щелчки мышью и нажатия клавиш клавиатуры, и многое другое.

Вместо того, чтобы управлять примитивным роботом, компоненты программы Scratch управляют различными спрайтами и изображениями и могут быть использованы для создания прекрасных интерактивных историй, игр, музыки и изоб-

разительного искусства. Однако среди главных достоинств Scratch – огромное сообщество тех, кто делится своими произведениями на <http://scratch.mit.edu>. Здесь вы можете не только поиграть с произведениями других, но и узнать, как они работают, и интегрировать их идеи в свои проекты.

Желающим приступить к делу со Scratch рекомендуем ознакомиться с изображениями и аудио-эффектами их сайта, прежде чем перейти к видеороликам на [http://info.scratch.mit.edu/Video\\_Tutorials](http://info.scratch.mit.edu/Video_Tutorials).

➤ Пропустили номер? Узнайте на с. 104, как получить его прямо сейчас.

Иногда вы, возможно, решите проверить, является ли условие истинным, и выполнить действие, если оно таково, или другое действие, если условие ложно. Это можно сделать с помощью команды `if ... else ...`, которая используется так:

```
if front_is_clear:
    move
else:
    turnoff
```

Можно даже проверять множественные условия, с помощью команды `if ... elif ...`

```
else:
if front_is_clear:
    move
elif right_is_clear:
    turnright
    move
elif left_is_clear:
    turnleft
    move
else:
    turnoff
```

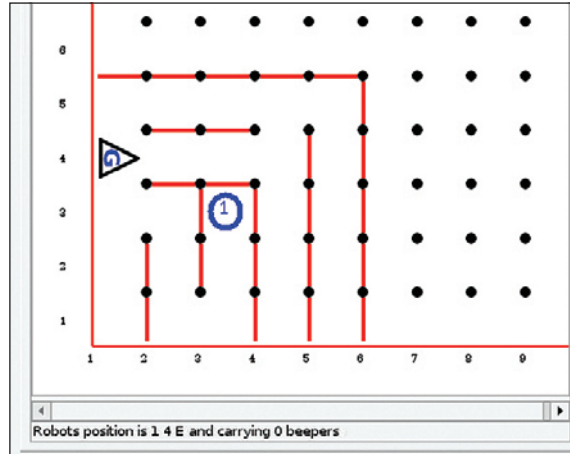
Конечное `else` является опциональным. Оно обеспечивает ответ на ситуацию по умолчанию: если все остальное ложно, сделайте это – так что обычно его стоит использовать.

Помимо простого предоставления Guido возможности принимать самостоятельные решения, вы можете увидеть, насколько это может быть удобно. Где-то в начале этого руководства вы увидели, что, если вы попытаетесь выполнить команду `pickbeeper` при отсутствии маячка на углу, где находится Guido, вы получите ошибку, и программа остановится. Если бы вы сделали ошибки в других частях программы, вы, возможно, увидели бы то же самое, попытавшись провести Guido через стену. Условные инструкции позволяют нам добавлять безопасную проверку нашего кода, чтобы программа не завершала свою работу из-за такой ошибки.

## Еще раз повторимся

Можно также использовать условные инструкции с помощью другой команды повторения: `while`. Эта инструкция проверяет, истинно ли условие, исполняет команду – или последовательность команд – в своем теле (части с отступом), а затем еще раз проверяет истинность условия. Если да, она снова исполнит «тело», и будет повторять это до тех пор, пока условие не окажется ложным. Мы говорим, что она будет зациклена на теле команды.

С командой `while` стоит быть поосторожнее. Если вы укажете условие, которое никогда не станет ложным, есть риск застрять в бесконечном цикле. Как и большинства других бесконечных вещей, бесконечных циклов лучше избегать



► Если вы ищете более сложную задачу, почему бы вам не попробовать справиться с этим лабиринтом? Он по-прежнему является как часть *GvR!*

С помощью цикла `while` мы можем далее усовершенствовать нашу программу *GvR2*, сделав ее безопаснее и дав больше управления Guido. После первой партии команд `move` для перемещения Guido на нужную улицу и разворота его на восток, мы можем использовать команду `while`, чтобы перемещать его вперед, пока он не окажется на том же углу, что и маячок:

```
while not_next_to_a_beeper:
    move
pickbeepers
move
```

Собрав воедино все, чему мы научились, мы можем завершить программу *GvR2* всего за 10 строк. Используем мы только основные команды, у нас это заняло бы 20 строк. Наша программа стала не только короче – она стала более ясной, и ее можно с легкостью расширить для более крупных и сложных миров.

Мы рассказали вам все, что нужно знать о Guido. То, чему вы научились на нашем уроке, поможет вам заставить его делать более умные вещи, например, находить выход из лабиринта, чтобы найти в конце маячок в качестве приза.

## Другие проекты

Для решения проблемы лабиринта у вас должен быть файл `maze.wld`, поставляемый с установкой *GvR* по умолчанию. Как и в предыдущих проблемах, вариантов ее решения множество: например, указать каждый шаг по отдельности, или поступить более разумно и дать Guido шанс самостоятельно принимать соответствующие решения. Советуем попробовать оба способа.

Если лабиринт покажется вам нудным, и вы захотите создать собственные миры, загляните в наш раздел «Создание Миров [World building]» – там есть инструкции для этого. **LXF**

## Что дальше?

Guido van Robot познакомил вас с самыми основами программирования: как разбить проблему на отдельные шаги, создать новые инструкции и управлять потоком программы (с помощью команд `if`, `while` и `do`). Но кое-чему вы в *GvR* научиться не можете, хотя для большинства компьютерных программ это общее место.

К счастью, все в *Guido van Robot* основано на языке программирования Python, который используют многие профессиональные разработчики.

Python может делать все то же, что и *GvR*, и многое другое. Он может запросить у пользователя ввод информации и дать на нее соответствующий ответ или запомнить то, что происходило раньше,

он знает все о данных самых разных типов (таких как строковые, числовые, булевские и списки), а также позволяет вам определять собственный тип данных – можно даже воспользоваться им для создания собственного усовершенствованного Guido.

Поскольку *GvR* основан на Python, переход от управления Guido к собственно программированию на Python не должен представлять собой особой сложности, поскольку синтаксис, то есть структурирование и расположение текста и структура программ, тот же самый. Более того, есть превосходные свободные ресурсы, которые помогут вам начать. С вашего разрешения, мы бы рекомендовали начать с *Think Python: How to think like a com-*

*puter Scientist* [Думайте на Python: учимся думать, как компьютерный специалист] ([www.greenteapress.com/thinkpython](http://www.greenteapress.com/thinkpython)). По мере вашего обучения вы найдете немало даже воспользоваться проектами и идеями, которые сумеют вас захватить.

А если вы решите, что Python не для вас, есть и другие отличные языки программирования, которые могут освоить начинающие. Одним из популярных является Scheme, и мы настоятельно рекомендуем вам книги *Little Schemer* [здесь игра слов: можно перевести как «Юный программист на языке Scheme»] или «Маленький интриган», *прим. ред.*] и *How To Design Programs* [Как создавать программы].



# Безопасность:

Нейл Ботвик покажет, как обезопасить ваши данные с помощью *TrueCrypt*, независимо от того, на скольких ПК или ОС вы собираетесь их использовать.



## Наш эксперт

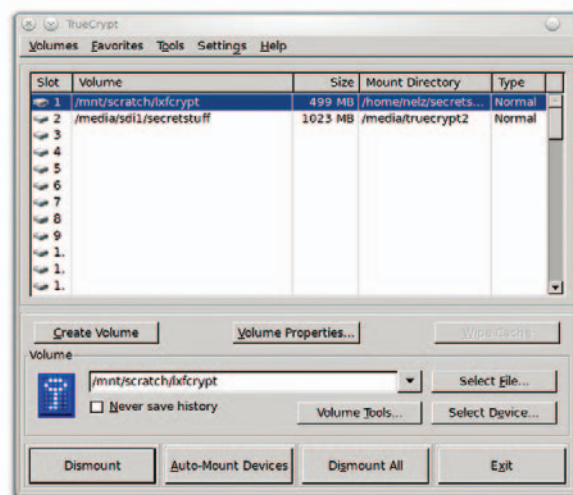
У **Нейла Ботвика** по компьютеру в каждой комнате, но по соображениям безопасности он ни за что не скажет, где находится центральный сервер.

**Ш**ифрование данных бывает жизненно важным – как для личной жизни, так и для работы. Есть разные возможности для хранения зашифрованных данных: так, Ubuntu применяет виртуальную файловую систему *ecryptfs*, включенную в ядро, для шифрования домашних директорий. Ядро также включает *dmccrypt* для шифрования целых разделов – именно так я предпочитаю защищать свой ноутбук от посторонних глаз. Однако иногда нужен более мобильный подход: иногда важные данные приходится хранить на USB-брелке, чтобы посмотреть их на другом компьютере или в другой ОС. Здесь-то и выходит на первый план *TrueCrypt*. *TrueCrypt* ([www.truecrypt.org](http://www.truecrypt.org)) – это инструмент, который шифрует все устройство или создает зашифрованный файл, рассматриваемый как устройство, где хранятся файлы. Программа мультиплатформенная, и данные, зашифрованные ею, можно прочитать в системах Mac или Windows; и в ней имеются статичные портативные бинарники, так что вам даже не нужно устанавливать никаких программ на компьютер, в котором вы собираетесь читать свои файлы. Также, в отличие от опций ядра, *TrueCrypt* имеет GUI, и вам нужно лишь помнить свой пароль, чтобы получить доступ к своим данным.

## Создание тома

Установив *TrueCrypt* обычным для вашего дистрибутива способом – практически все дистрибутивы имеют этот пакет в своих репозиториях, так что используйте стандартный менеджер пакетов – подключите носитель USB с достаточным объемом свободного места, запустите *TrueCrypt* и нажмите на Create Volume [Создать том]. У вас есть выбор: создать том в файле или зашифровать весь диск или раздел; на данный момент выбирайте файл. Это не только проще, но также позволит вам оставить место на носителе для незашифрованных файлов, что очень важно, если вы хотите включить программу *TrueCrypt*. Создайте стандартный том и выберите имя файла. При попытке использовать существующий файл тот будет перезаписан – отдельные файлы *TrueCrypt* не шифрует. Если вы создаете контейнер на устройстве, отформатированном в FAT, например, USB-брелке, размер тома не должен превышать 4 Гб, потому что FAT не работает с файлами такого размера.

Далее вам предложат выбрать применяемый алгоритм шифрования; для большинства случаев прекрасно подойдет выбор по умолчанию. По кнопке Test можно убедиться, что выбранный вами алгоритм поддерживается вашей системой – нажмите Test, затем Auto-Test All. Кнопка Benchmark демонстрирует производительность алгоритмов. Если вы намерены зашифровать большие файлы в медленной системе, вам, возможно, стоит выбрать ал-



➤ Главное окно *TrueCrypt* со списком подмонтированных томов.

горитм побыстрее, но если вы сохраняете данные на USB-брелке, основным ограничивающим фактором будет, скорее всего, скорость записи устройства. Установите размер тома и выберите, использовать ли пароль или файл ключа; здесь мы выберем пароль. Выберите файловую систему. Максимальную портативность обеспечит опция FAT; для использования только в Linux более логичным будет выбрать одну из файловых систем ext. Нажмите Next, дайте мышке поработать, чтобы сгенерировать случайные данные, используемые для шифрования, и нажмите Format.

Монтирование вашего зашифрованного тома также делается из GUI. Нажмите на Select File и выберите только что созданный файл, затем дважды щелкните по одному из номеров слотов вверху дисплея. Вам предложат ввести пароль, а затем на дисплее отобразится информация о томе, в том числе и точка монтирования – по умолчанию это `/media/truecryptN`, где N – номер слота. Кнопка Options [Параметры] в окне запроса пароля позволит изменить точку монтирования и другие настройки. Размонтирование не менее просто: выберите слот и нажмите Dismount. Можно сделать это и командной строкой, вместо того, чтобы загружать GUI – например, при монтировании из скрипта:

```
truecrypt /path/to/file
```

Вам предложат ввести пароль и затем монтировать том в первом имеющемся слоте, используя точку монтирования `/media/truecryptN`. Можете монтировать ее где угодно, указав точку монтирования в командной строке:

```
truecrypt /mnt/scratch/lxrcrypt ~/secretstuff
```

При этом должна существовать точка монтирования. Размонтирование производится добавлением опции `-d` или `--dismount`:

```
truecrypt --dismount /path/to/file
```

Вы также можете указать пароль в командной строке с помощью опции `-p` или `--password`:

```
truecrypt --password "some r3ally 10ng pa55word" /path/to/file
```

Однако использовать в данном случае скрипт – не особенно умная мысль. Кавычки нужны, если пароль содержит пробелы

## Играйте честно

Мы никоим образом не предлагаем применять *TrueCrypt* или любую иную технологию шифрования с незаконной целью, но есть множество вполне законных причин желать защитить свои данные от любопыт-

ных глаз. По сути, во многих случаях как раз не делать этого будет незаконным: тот, кто владеет важной информацией о других, несет ответственность за надежное и безопасное хранение этой информации.



# Шифруем данные

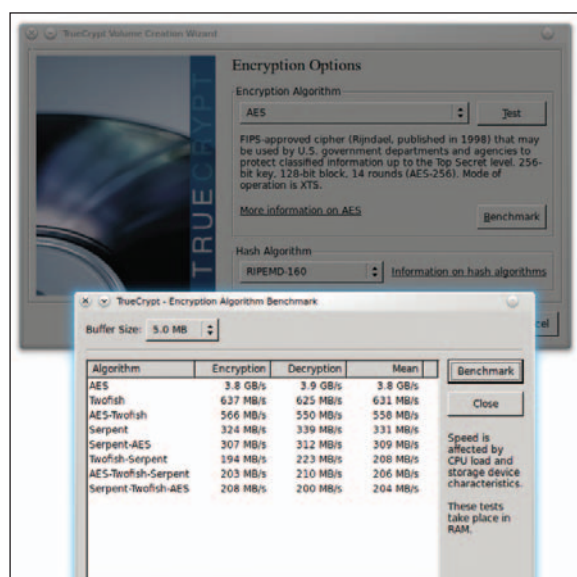
или любые другие символы, которые могут запутать оболочку. Из GUI создавать тома *TrueCrypt* проще, но можно это делать и из командной строки – вот пример создания тома:

```
truecrypt --text --non-interactive --password=notmyrealone --
volume-type=normal --size=1073741824 --encryption=AES
--hash=RIPMD-160 --filesystem=FAT --keyfiles="" --create somefile
```

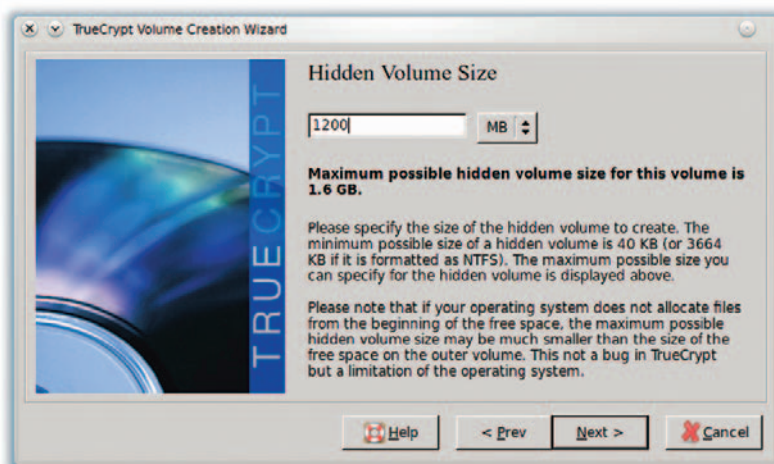
Но нужно прочесть документацию, чтобы разобраться с опциями.

Если вы пропустите какую-либо настройку, *TrueCrypt* сообщит об этом, что обезценивает использование командной строки как альтернативы (если только вы не создаете том в системе без рабочего стола). Опция **--non-interactive** нужна, чтобы *TrueCrypt* перестал просить вас использовать клавиатуру для генерации случайных данных для шифрования; но тогда данные будут менее случайны, а значит, менее безопасны. Вообще-то стоит использовать GUI для создания томов при любом удобном случае. *TrueCrypt* снабжен большим руководством в формате PDF, но там затронуты только версии опций командной строки для Windows. Более подробную и полезную информацию вы получите, запустив `truecrypt --help`

Истинной портативности ради, вам нужно быть в состоянии расшифровать ваши тома *TrueCrypt*, где бы вы ни находились. Есть отдельные программы для Windows и Linux, доступные на [truecrypt.org](http://truecrypt.org). Скачайте 32-битные пакеты для Windows и Linux и запустите каждый из них в соответствующей ОС. Пакет для Linux нужно распаковать до того, как вы сможете запустить оттуда хоть один файл. Можете запустить программу установки Windows с помощью *Wine*, если у вас нет под рукой Windows. В любом случае, выберите опцию Extract. Программа установки Windows разместит файлы в *TrueCrypt* в текущей директории, а программа установки Linux создаст tar-архив в `/tmp`, который вы должны распаковать. Скопируйте Windows **TrueCrypt.exe** и Linux **Truecrypt** на устройство, которая содержит ваш том, и можете запускать их непосредственно с него, без необходимости установки.



➤ *TrueCrypt* дает широкие возможности для освоения азов шифрования, а его настройки по умолчанию подойдут практически всем.



## Прячьте свои секреты

Одним из недостатков шифрованных файловых систем является то, что, скрывая содержимое файловой системы, они в то же время не скрывают того факта, что вы что-то прячете. *TrueCrypt* заполняет новое устройство случайными данными, и у него нет подписи в начале файла, в отличие от разделов `dmccrypt`; поэтому раздел *TrueCrypt* выглядит, как неиспользованный раздел, заполненный случайными данными, но там нет скрытого многогигабайтного файла, содержащего случайные данные. Также могут возникнуть ситуации, когда вам придется передать кому-то ключи шифрования. В *TrueCrypt* имеется возможность найти обходной путь – скрытые тома, о которых мы говорили ранее.

*TrueCrypt* создает два зашифрованных тома, один внутри другого. Внешний файл – это стандартный том *TrueCrypt*, где вы можете сохранять файлы, как обычно. Внутри него находится еще один том, который не появляется ни в одном списке директории, и место, занятое им, видится всеми стандартными инструментами как свободное. По мере того, как *TrueCrypt* заполняет любой новый том случайными данными, тот факт, что все «свободное» место во внешнем пространстве содержит данные, не является подозрительным. Каждый том имеет свой пароль; используйте пароль к внешнему тому – и вы увидите только содержащиеся в нем данные; но если использовать пароль к скрытому тому, монтируется именно он. Это означает, что вы можете передать пароль к вашему скрытому тому, зная, что остальные данные, которые вам нужно сохранить в безопасности, спрятаны. Имеет смысл использовать для скрытого тома более длинные и более безопасные пароли, чтобы взломщики паролей сначала взломали пароль внешнего тома.

Процесс создания скрытого тома начинается во многом так же, как и для стандартного тома; затем вас просят скопировать некоторые данные в том перед тем, как будет создана скрытая часть. Это – повторение шагов для создания стандартного тома. Теперь вы можете открыть любой том, введя пароль. Однако, если вы открываете внешний том и запишете что-то в него, вы рискуете повредить скрытый том, потому что инструменты файловой системы не могут распознать, где он находится. Справочник расскажет, как избежать подобной ситуации. **LXF**

➤ Создавая скрытый том, оставьте место для добавления файлов во внешний том – пустой контейнер *TrueCrypt* довольно очевиден.

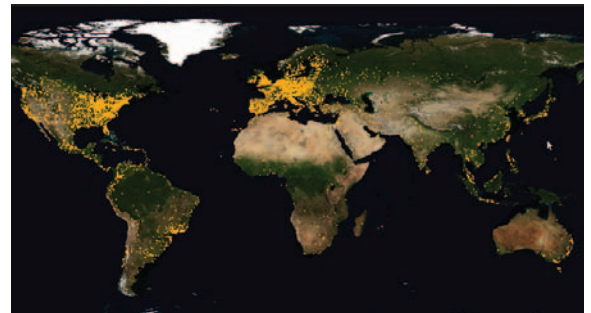
# Moodle: Обучите

Роджер Уотерс и Pink Floyd в песне “Another Brick in the Wall” ошиблись – образование нам нужно! **Бен Эверард** создает онлайн-университет с *Moodle*.



## Наш эксперт

**Бен Эверард** бросил работу IT-консультанта, чтобы отправиться на два года в Танзанию устанавливать системы на базе Ubuntu в школах. Теперь его знания находят применение в бурлящем котле открытий – редакции *Linux Format*.



Мы раскрасим целый свет... оранжевые точки на карте – это сервера *Moodle*.

простой способ настроить сервер *Moodle* – использовать TurnKey Linux. Это дистрибутив Linux, предоставляющий готовые к работе виртуальные машины (VM) с разной серверной настройкой. Потребуется установленная *VirtualBox* – вы найдете ее в репозиториях своего дистрибутива. Для сборки вашей системы *Moodle* следуйте пошаговому руководству. Нужно, чтобы ваш компьютер был соединен с сетью через DHCP. Тестирование системы на VM – отличный способ узнать что-то новое. Если вы умудрились запороть систему, просто начните сначала в свежей VM. В порядке альтернативы можете использовать функцию моментального снимка системы [snapshot] в *VirtualBox*, которая позволит вам вернуться к предыдущему состоянию.

Особо радует в *Moodle* ориентация этой системы на пользователей без технической подготовки: можно начать с нуля, построить сервер и администрировать его, не затрагивая командной строки (почти). А значит, преподаватели могут управлять онлайн-содержанием своих курсов, имея лишь минимум знаний о компьютере. Как и полагается для образовательного ПО, контент разбит по курсам. Они, в свою очередь, делятся на разделы, приблизительно соответствующие урокам. К этим разделам можно добавлять пункты, которые могут или дополнить их до очного урока, или стать самостоятельным онлайн-уроком.

Однако мы опережаем события. Прежде чем создавать курсы, нужно вернуться немного назад и заняться более приземленной

**L**inux отлично подходит для образования. Гибкость, поддержка маломощных устройств и дружелюбное сообщество – все это помогает школам и университетам по всему миру извлечь максимум из своих ресурсов. И Linux помогает образованию не только на клиентской стороне: это также основной выбор для серверов. Причины прекрасной работы Linux в данной сфере те же, что обусловили и его популярность на коммерческих серверах. Но есть одно выдающееся приложение специально для обучения: *Moodle*. В мире более 66 000 сайтов, применяющих это приложение для обслуживания более 6 000 000 курсов для почти 60 000 000 пользователей. Это бездна образования, и все это работает на программе с открытым кодом.

*Moodle* – система управления курсами, то есть система управления контентом [Content Management System, CMS], разработанная именно для учебных курсов на базе традиционного набора LAMP для обеспечения работы в сети. Это позволяет преподавателям и студентам получать ко всему доступ и обновлять все через свои браузеры, не устанавливая какие-то специальные программы. Однако не стоит считать *Moodle* простым ПО для школ и колледжей. Мы учимся на протяжении всей нашей жизни, и каждой организации стоит задуматься о культивации знаний; а хорошая система управления курсами – важная часть этого процесса. Вы можете быстро оценить способности *Moodle*, нацелив свой браузер на их демо-сайт – <http://school.demo.moodle.net>. Имеются пакеты для большинства дистрибутивов, но самый

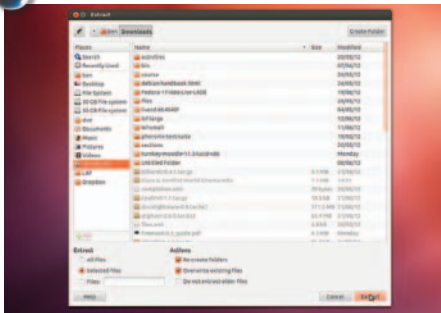
## Развертывание системы

Протестировав настройку и работу своей системы, вы, возможно, решите переместить ее на другой сервер, чтобы обновить ее. Начать с самого начала и выполнить те же самые шаги может оказаться напрасной тратой сил, да и легко просто упустить какие-то детали по забывчивости. Лучшей возможностью будет перенести свои настройки на новую машину. Это потребует некоторого знания командной строки, но не должно быть чрезмерно сложным. Загляните на [docs.moodle.org/en/Moodle\\_migration](http://docs.moodle.org/en/Moodle_migration), там вы найдете пошаговые инструкции.

# ВЕСЬ МИР!

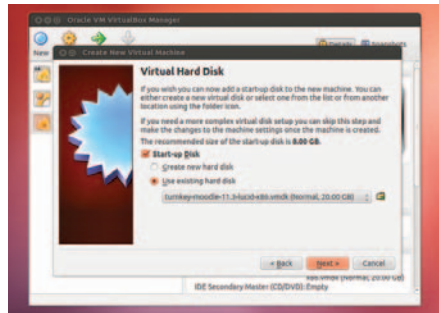


## Шаг за шагом: Установка и настройка Moodle



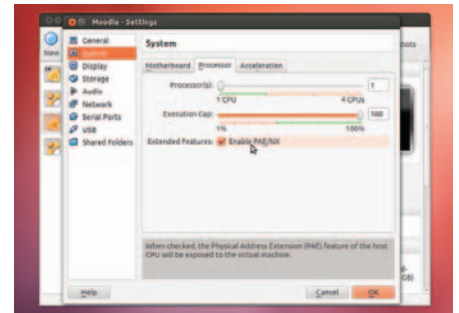
### 1 Скачайте файл VM

Загрузите файл VM с [www.turnkeylinux.org/moodle](http://www.turnkeylinux.org/moodle) (шаг, когда вам предлагают указать свой адрес электронной почты, можно пропустить) и распакуйте его.



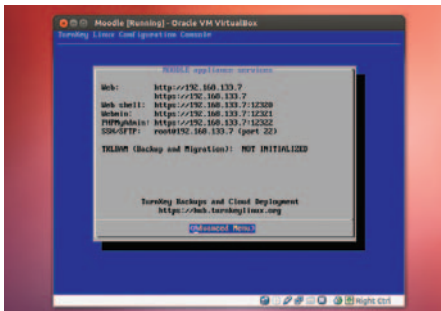
### 2 Создайте VM

Откройте *VirtualBox* и нажмите на New [Новая], чтобы создать VM. Дайте ей название, выделите ей некий объем памяти и выберите свой файл VMDK.



### 3 Настройте VM

Перейдите в Settings > System > Processor [... > Система > Процессор], проверьте Enable PAE/NX, перейдите в Networking [Сеть] и смените Adapter 1 на Bridged Adapter.



### 4 Запуск

Нажмите на Start [Пуск], чтобы запустить VM. Введите пароли для учетных записей администраторов и пропустите раздел инициализации сервисов хаба.



### 5 Соединение

Теперь можно открыть браузер на машине-хосте и соединиться с IP-адресом, указанным в виртуальной машине.



### 6 Ввод данных о пользователе

При первом входе в систему под именем пользователя admin и указанным вами паролем надо будет указать более подробную информацию для пользователя.

задачей создания пользователей. В целях нашего урока мы создадим одного преподавателя и одного студента. В реальности обычно требуется больше. Чтобы не создавать все это вручную, вы можете либо доверить пользователям создание собственных учетных записей, либо загрузить пользователей из файла CSV.

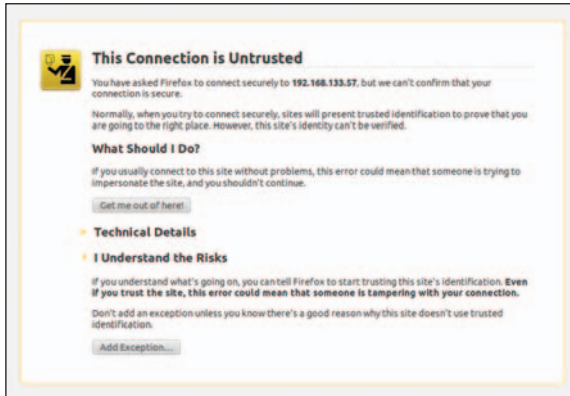
В меню Settings, перейдите в Site Administration > Users > Accounts > Add A New User [Администрирование сайта > Пользователи > Учетные записи > Добавить нового пользователя]. Введите информацию по каждому пользователю по очереди (помните, что обязательны только поля, помеченные красным), и нажмите на Update Profile [Обновить профиль]. При желании можно изменить требования паролей в Site Administration > Security > Site Policies [... > Безопасность > Политики сайта]. Теперь предоставим преподавателю разрешения для создания курсов. Сделайте это, перейдя в Site Administration > Users > Permissions > Assign System Roles [... > ... > Разрешения > Задать роли в системе] и на-

жав на Course Creator [Создатель курса]. Перед вами откроется страница, где вы можете присвоить любому пользователю право создавать курсы. Выделите своего вновь созданного преподавателя в Potential Users [Потенциальные пользователи] и нажмите на Add [Добавить]. Он должен появиться в колонке Existing Users [Имеющиеся пользователи]. Сделав это, выйдите из системы как пользователь Admin и войдите уже как преподаватель. Статусы преподавателей и студентов не являются жесткими: преподаватель на одном курсе может быть студентом на другом.

На домашней странице вы увидите кнопку Add A New Course [Добавить новый курс]. Нажмите на нее, чтобы перейти на страницу Edit Course Settings [Редактировать параметры курса] и ввести подробную информацию по курсу. И снова, обязательны только поля, помеченные красным. Мы назвали наш курс Linux Certification, сокращенно LXC; ну, а вы можете дать своему любое название. Если вы измените настройки по умолчанию, он будет

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

► Moodle использует SSL, но по умолчанию сам подписывает сертификат. Ваш браузер может отобразить предупреждение, но вы не волнуйтесь, просто добавьте исключение в системе безопасности.



разбит на десять еженедельных сессий. Они будут отображаться в Weekly Outline [Недельная картина] после создания курса. В Moodle два стандартных вида страницы: один – для просмотра информации на ней, а другой – для редактирования. Переключаются между ними кнопкой в правом верхнем углу под названием Turn Editing On [Включить редактирование]. Нажмите на нее, и мы добавим некий контент вашему курсу. Есть два разных вида контента, добавляемого к курсам: ресурсы [resources] и действия [activities]. Ресурсы – статические объекты, которыми студенты могут воспользоваться для обучения, а действия – то, что студенты делают. Щелчок по выпадающему окну выдаст полный список видов контента, поддерживаемого каждым.

Для начала создадим статическую страницу с информацией для студентов. В первой неделе откройте выпадающее меню Add A Resource [Добавить ресурс] и выберите Page. Откроется Adding A New Page [Добавление новой страницы]. Здесь вводятся имя, описание и контент. Покончив с этим, нажмите на Save And Display [Сохранить и показать], чтобы взглянуть на свое творение. Оно должно появиться в виде обычной страницы Moodle, и через нее вы можете снабжать своих студентов информацией. Но курс из одних статических web-страниц, вероятно, быстро надоеет участникам. В идеале надо бы добавить нечто интерактивное.

## Обучение в среде равных

Вместо того, чтобы преподаватель диктовал все студентам, сами студенты могут помогать друг другу. Возможно, это звучит противоречиво, но ряд исследований показывает, что студенты гораздо лучше усваивают информацию, полученную от равного, нежели от своего преподавателя. Памятуя об этом, давайте создадим форум, где студенты будут обсуждать свои идеи.

Щелкните по выпадающему меню Add An Activity [Добавить действие] и выберите Forum. На новой странице можно указывать подробную информацию о создаваемом форуме. Таковых предлагается несколько типов, функционирующих немного по-разному. Форум по умолчанию – Default Forum For Standard Use [Форум по умолчанию для стандартного использования] – обычный форум, примерно такой же, как на [www.linuxformat.com](http://www.linuxformat.com)

Вероятно, интересное всех форумы Q-and-A, где преподаватель размещает вопросы студентам. Но студенты не могут видеть других ответов, пока не дадут своего. Дав ответ, студент может размещать комментарии к другим ответам, так же, как и на других форумах. Это способствует тому, чтобы студенты сами обдумывали свои действия, заодно предоставляя им возможность обсудить свои идеи с другими обучающимися на курсе. Moodle поддерживает конструктивистский подход к обучению. Проще говоря, это означает, что студенты обучаются, создавая нечто самостоятельно. В контексте Moodle, они могут «создавать контент».

Помимо форумов, преподаватели могут добавлять wiki, где студенты могут совместно трудиться над созданием сайтов по изучаемым темам. Это отлично работает с конструктивистскими методами и помогает собрать материал по курсу, который смогут использовать будущие студенты. Как и форумы, wiki можно добавлять несколькими щелчками мыши. На экране Weekly Outline со включенным редактированием, щелкните по Add An Activity и выберите Wiki. Дайте ей имя и описание, так же, как вы делали с форумом. Изменение формата на NWiki придаст ему внешний вид, хорошо знакомый тем, кто постоянно использует wiki.

Но дело не только в новомодных форумах и wiki. Moodle также поддерживает и более традиционные инструменты преподавания – например, проверочные работы. Перед тем, как создать саму проверочную работу, следует создать для нее вопросы.

► Перейдите в Course Administration > Question Bank > Categories [... > Банк вопросов > Категории] и введите новое имя. Затем щелкните по Add Category [Добавить категорию].

► В Course Administration > Question Bank > Questions [... > ... > Вопросы] вы теперь можете выбрать свою новую категорию и нажать на Create A New Question [Создать новый вопрос].

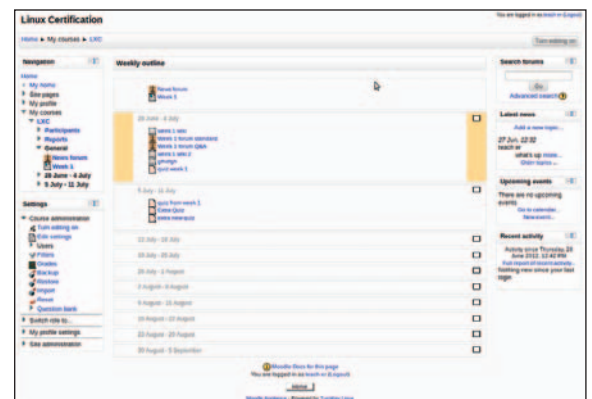
► Выберите тип вопроса.

► Введите детали вопроса. Вы можете ввести текст, который будет показан пользователю в зависимости от его ответов в различных окошках Feedback [с обратной связью]. Можете также выбрать допуски для числовых вопросов и прочих подобных моментов, в зависимости от типа выбранного вами вопроса.

► Щелкните по Save Changes. Теперь можете создавать новый вопрос.

Скомпонуйте свой банк вопросов в контрольный опросник.

На экране Weekly Outline откройте выпадающее меню Create An Activity [Создать действие] и выберите Quiz [Опросник]. Дайте ему имя и переключите Question Order [Порядок вопросов] на Shuffle Randomly [Случайный]. Можно ввести описание и настроить обратную связь на основе оценочных критериев проверочной работы, но это опционально. Произведя все изменения, щелкните по Save And Display, чтобы создать проверочную работу. На новом экране щелкните по Add A Random Question [Добавить произвольный вопрос], выберите новую категорию в выпадающем меню и нажмите на Add Random Question [Добавляем произвольный вопрос]. Повторяйте этот процесс, чтобы добавить к проверочной работе столько вопросов, сколько вам надо. Учтите: во время тестирования мы обнаружили, что отчеты по проверочным работам некорректно работают с одним вопросом. Есть ряд других действий и ресурсов, которые можно добавлять к курсам. Мы не будем рассматривать их здесь все – они работают так же, как и те, которые



► Можно добавить к своему курсу ленту новостей, чтобы все были информированы о последних событиях.

**Скорая помощь**

Навигация: раскладка меню может показаться немного загадочной, но понижее меню располагается окно поиска, которое пригодится для поиска скрытых опций.

► Пропустили номер? Узнайте на с. 104, как получить его прямо сейчас.

## Mooch

В школах всего мира есть серверы *Moodle* с курсами самой разнообразной тематики. Многие из них предназначены только для использования людьми, зачисленными в определенный колледж, но есть и открытые для публики. Как было бы здорово иметь возможность поиска общественно доступных курсов! И такая возможность есть. Mooch – Узел открытого сообщества *Moodle.Org* – это сайт, который соединяет несколько серверов *Moodle* и позволяет искать курсы. Есть два типа курсов, которые можно

искать: те, на которые можно записаться онлайн, и те, которые можно скачать. Скачиваемые курсы затем можно загрузить на другой экземпляр *Moodle*, чтобы использовать самостоятельно, или дать возможность пользования и другим людям тоже. Чтобы импортировать курс, перейдите в Course Administration > Restore [...] > Восстановить, затем выделите загруженный курс и следуйте инструкциям. Если исходный сервер использует другую версию *Moodle*, с импортом могут быть проблемы.



➤ Ищите имеющиеся курсы *Moodle* с помощью Mooch.

мы рассмотрели. Среди самых удобных – Assignment [Задание]. Если возникнут проблемы, на каждой странице внизу есть ссылка под названием Moodle Docs For This Page [Документы Moodle для данной страницы]. Вы удивитесь, но ссылка приведет на соответствующую страницу справочной документации *Moodle*.

## Записывайтесь на курсы

Настроив курс (хотя бы частично), запишем на него студентов. Перейдите в Course Administration > Users > Enrolment Settings [Администрирование курса > Пользователи > Параметры регистрации] и нажмите на значок, похожий на два лица, чтобы добавить ранее созданного студента. Сделав это, выйдите из системы как преподаватель и войдите как студент, чтобы увидеть отображение добавленного вами контента для слушателей курсов. Студенты видят практически ту же раскладку, что и преподаватель с выключенным редактированием, и могут просматривать ресурсы и участвовать в действиях. Разместите пару ответов в форуме, добавьте содержания в wiki, выполните проверочную работу... затем выйдите и снова войдите как преподаватель.

Конструктивистский подход основан на поощрении студентов к участию во всяческих действиях. Однако не все студенты работают охотно. Иногда преподавателю нужно подтолкнуть студента к большей активности, а для этого активность требуется оценить. На маленьком курсе чья-то леность будет заметна; но если у вас обучаются сотни людей, она вовсе не бросится в глаза.

К счастью, *Moodle* позволяет преподавателю отслеживать активность каждого студента. Войдите как преподаватель и перейдите в My Courses > Course Name > Reports > Participation Report [Мои курсы > Название курса > Отчеты > Отчет об участии]. Выберите здесь модуль деятельности, переключите Show Only на Students и нажмите на Go. Появятся данные по всем записанным студентам, и вы узнаете, завершили ли они All Actions [Все действия] (то есть просмотр и размещение материалов в деятельности). Если нужно заставить некоторых студентов быть поактивнее, щелкните по Select All No [Выбрать всем Нет], убедитесь, что With Selected Users [С выбранными пользователями] настроено на Send A Message [Отправить сообщение], и нажмите OK. Затем введите сообщение студентам, что им нужно активнее участвовать в деятельности. Нажмите на Preview [Предпросмотр] – а если сообщение вас устроит, то на Send [Отправить]. Сообщение отправится на их учетную запись *Moodle*.

Естественно, нам надо знать результаты проверочных работ студентов, а также их онлайн-активность. Перейдите в My Courses > Course Name, затем в Course Administration > Grades [...] > Оценки] в нижнем меню. Появится обзор результатов работы студентов по каждому виду деятельности, в виде сетки (способ оценки – буквы, проценты, и т.д. – можно настроить в Grade Administration > Course Grade Settings [Администрирование оценок > Парамет-

ры оценок курса]). Более подробную информацию по результатам работы студентов в определенных видах деятельности, щелкните по заглавию деятельности в верхней части колонки. Ее можно развернуть и узнать, как студенты оценили вашу проверочную работу, и при желании скачать эту информацию для анализа во внешнем приложении, например, в электронной таблице.

Настроив сервер и разместив на нем курсы, в принципе можно просто оставить его работать дальше. Однако любой системный администратор знает: в реальном мире все идет не так. Или из-за отказа жесткого диска, или из-за сбоя обновления ваш сервер рано или поздно упадет. А если это случится, вы намного быстрее вернете все к рабочему состоянию, если у вас будут своевременные резервные копии. *Moodle* позволяет их делать в зависимости от имеющихся у вас разрешений. Преподаватели могут копировать свои курсы в особо важные моменты, а системные администраторы – настроить скрипты автоматического копирования, чтобы данные копировались регулярно. Преподаватели могут найти соответствующие опции в Course Administration, а Administrators [Администраторы] должны заглянуть в Site Administration [Администрирование сайта]. Однако создание резервной копии всей системы полностью немного более трудозатратно. Более подробную информацию вы найдете в Moodle Docs на [http://docs.moodle.org/23/en/Site\\_backup](http://docs.moodle.org/23/en/Site_backup)

*Moodle* – фантастически мощная система для обслуживания материалов курсов онлайн, и здесь мы коснулись ее лишь поверхностно. Почти ко всему имеется доступ через web-интерфейс, и стоит потратить некоторое время, щелкая по разным функциям и узнавая, что можно сделать. Например, к большинству действий можно ограничить доступ, предоставляя его только в определенные дни, а модули расширения позволяют сделать ссылки на медиа из других систем, например, Wikimedia или YouTube. LXF



## Скорая помощь

Внешний вид: Войдя в систему как admin, вы можете изменить внешний вид сайта, перейдя в Site Administration > Appearance [...] > Вид].

➤ Можете активировать мобильные web-сервисы, чтобы люди получали доступ к контенту *Moodle* с мобильных телефонов.

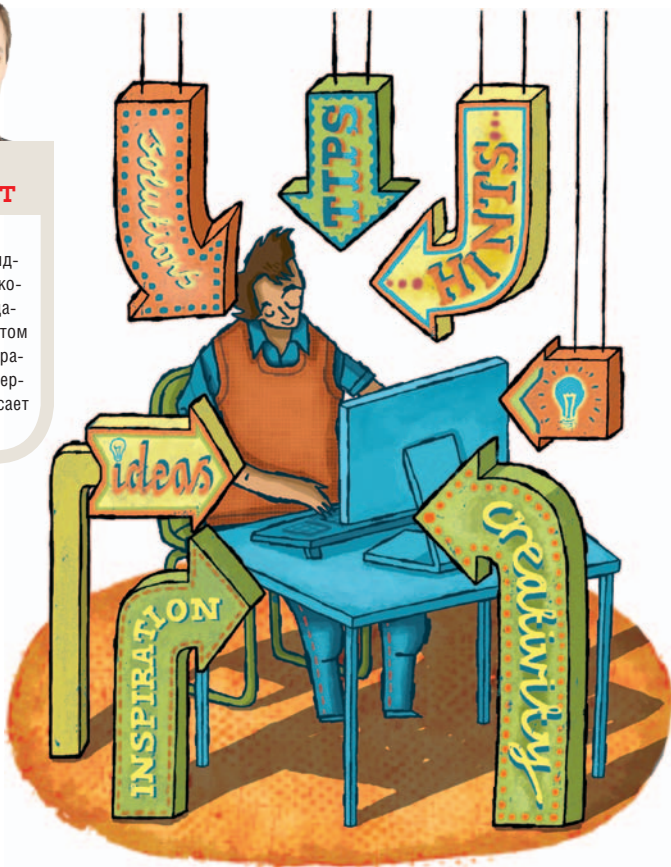
# FreeNAS: Храни

Майк Сондерс проводит 98% своего времени в Linux, но случайные прогулки в страну BSD ради смены декораций тоже доставляют ему удовольствие.



## Наш эксперт

Майк Сондерс сжился с командной строкой, и когда его вынуждают работать в этом новомодном «графическом» интерфейсе, его бросает в дрожь.



Хранить все яйца в одной корзине довольно рискованно. Когда среди этих яиц – любимые семейные фотографии, музыка и важные документы, риск еще отягчается. Но долой аллегории – скажем прямо: храните данные на отдельном устройстве, вдали от того компьютера, с которым работаете ежедневно. В конце концов, на основном компьютере вы можете пробовать новые программы и дистрибутивы, посещать множество сайтов и вставлять разнообразные устройства в USB-порт. Это здорово – потому-то и приятно быть линуксоидом; но это означает, что ваш повседневный компьютер – не самый защищенный и надежный компьютер в мире. Хранить на нем сотни гигабайт данных – не лучшая идея.

А вдобавок, как получить доступ к этим файлам с других компьютеров у вас дома или в офисе? Можно установить на главном компьютере *Samba* или *NFS*, но и здесь есть риски и ограничения. Если главный компьютер выйдет из строя из-за сбоя при обновлении дистрибутива, с других компьютеров нельзя будет получить доступ к его данным. А может статься, что ваша главная машина – ноутбук, и вы хотите время от времени выносить его в большой зал с голубым потолком (т.е. на воздух).

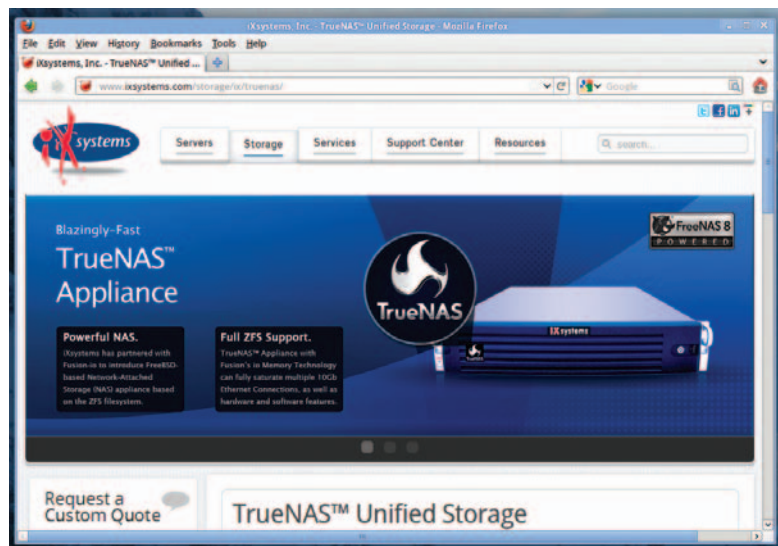
Нам поможет NAS (network attached storage – сетевое хранилище данных). В этом случае в сети есть специальный компьютер, задача которого – хранить файлы. Ему не нужен 32-дюймовый монитор и самый современный процессор; были бы лишь хорошая сетевая карта, достаточное количество оперативной памяти и большой жесткий диск (или несколько). Этот компьютер будет пылиться себе в уголке, предоставляя доступ к файлам и не требуя постоянного внимания к себе, в отличие от настольного компьютера.

## Сделай сам

Конечно, всегда можно купить готовый компьютер с NAS (за кучу денег), но это и затратно, и неспортивно. Здесь, в мире Linux, мы любим делать все по-своему; поэтому, вооружившись обычным компьютером и экземпляром *FreeNAS*, создадим-ка мы свой собственный сервер NAS с минимальными издержками. Скорее всего, у вас есть старый компьютер, который пылится без дела, и это прекрасный способ дать ему новую жизнь.

*FreeNAS* – прекрасная программа: бесплатная, с открытым исходным кодом, высоконадежная и зрелая. Но она работает не на платформе Linux, а на FreeBSD, о которой вы, возможно, слышали. Вкратце, FreeBSD – UNIX-подобная операционная система с открытым исходным кодом, во многом похожая на Linux, но с другим (более консервативным) процессом разработки. Хотя это не такая классная и яркая система, как Linux, и она держится в тени, но весьма надежна и прекрасно подходит для серверных задач.

На нашем уроке мы настроим компьютер с NAS и покажем, как им пользоваться, и в итоге все ваши файлы окажутся в надежном месте и вы сможете начать устанавливать Linux From Scratch на главном компьютере (не содрогаясь при мысли, что она сотрет жесткий диск).



» Можно купить спецкомпьютер с готовой *FreeNAS* (например, на сайте [www.ixsystems.com](http://www.ixsystems.com)), но намного веселее и интереснее сделать такой самим.

# И ПОЛЬЗУЙСЯ

## Выбираем «железо»

На нашем уроке мы будем пользоваться *FreeNAS 8*, основанной на FreeBSD 8.2. Аппаратные требования последней можно найти на сайте [www.freebsd.org/releases/8.2R/hardware.html](http://www.freebsd.org/releases/8.2R/hardware.html), но в целом сгодится любой достаточно современный компьютер со стандартными устройствами. FreeBSD не поддерживает столько экзотических устройств, как Linux, но для NAS достаточно лишь нескольких основных компонентов.

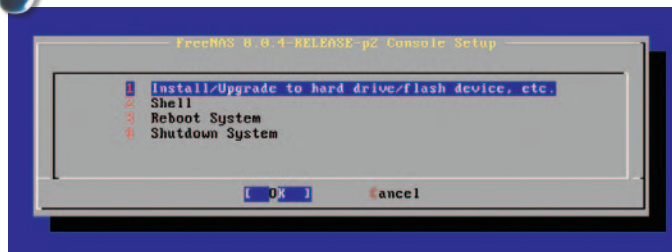
Если сервер NAS предназначается для серьезных дел и хранит многие терабайты данных на нескольких жестких дисках, вам понадобится 64-битный процессор и приличная оперативная память – не менее 4 Гб, по 1 Гб на каждый терабайт жесткого диска. Назначим такому серверу файловую систему ZFS: она прекрасно подходит для работы с большими объемами данных. Для менее

масштабного сервера, например, домашнего медиа-сервера объемом в несколько сотен Гб – от силы 1 Тб на одном диске, таких мощностей не нужно. Вполне достаточно 32-битного процессора с 2 Гб оперативной памяти и стандартной файловой системой, принятой в FreeBSD – UFS.

В любом случае вам понадобится быстрая сетевая карта для подключения к маршрутизатору (не вздумайте делать все это через Wi-Fi!) и флэшка объемом не менее 2 Гб. На ней будут размещены файлы *FreeNAS* и операционной системы, которые нельзя хранить на том же диске, что и файлы (если у вас два жестких диска, можно установить *FreeNAS* на первый диск и использовать другой диск для хранения файлов, но это будет зряшний перевод дискового пространства).



## Шаг за шагом: Установка FreeNAS



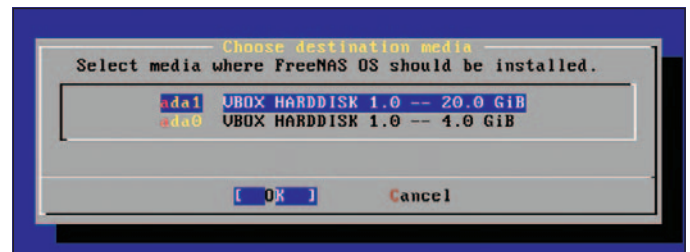
### 1 Запись диска и загрузка

Возьмите 32- или 64-битный ISO-образ с LXF DVD или загрузите его с [www.freenas.org](http://www.freenas.org). Запишите его на CD-R (именно как ISO-образ – не копируйте его на диск напрямую) и загрузите с него ваш будущий компьютер NAS. Появится меню установки. Выберите пункт 1 (Установка/Обновление).



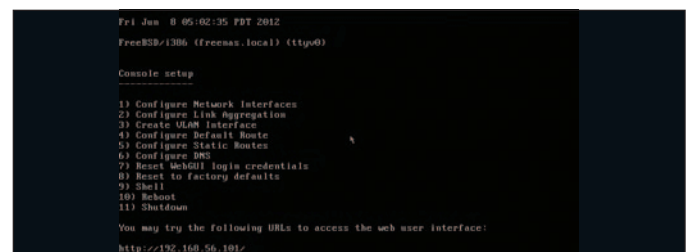
### 3 Установка ОС

Затем на него будут скопированы файлы, что на быстром компьютере займет всего несколько минут. В нижней части окна вы видите индикатор хода процесса. По окончании вам будет предложено вынуть диск и перезагрузить компьютер.



### 2 Выбор диска

Теперь выберите диск, где вы хотите установить *FreeNAS*. Важно отметить, что этот диск будет полностью затерт и будет использоваться только для файлов *FreeNAS*, поэтому для него достаточно 2 Гб. Советуем установить *FreeNAS* на USB-флэшку и приберечь диск(и) для хранения данных.



### 4 Загрузка

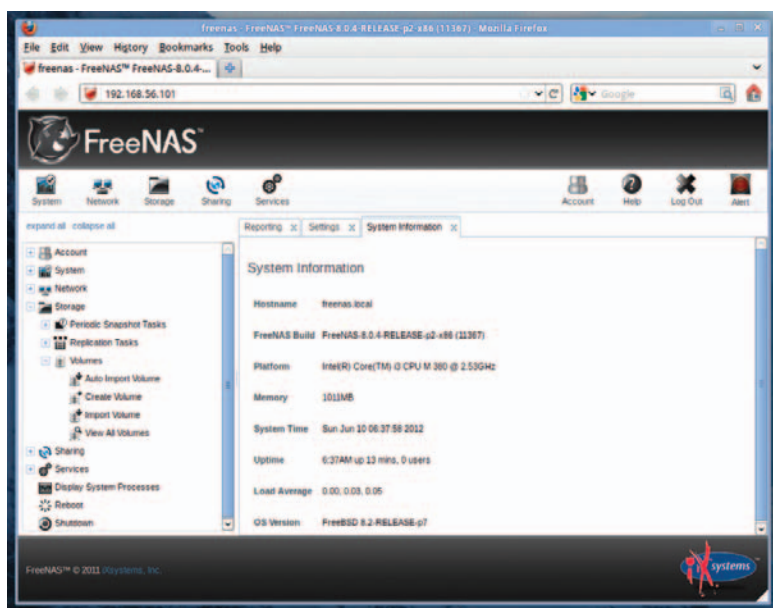
FreeBSD начнет загружаться – загрузочных сообщений будет немного больше, чем в графических дистрибутивах Linux, где они прикрыты экраном загрузки. Через пару минут вы увидите меню; там можно задать сетевые настройки. Вы также увидите IP-адрес для доступа к web-интерфейсу.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

## Настройка сервера

Прежде чем приступать к обслуживанию файлов нашим свежесобранным компьютером с *FreeNAS*, нужно кое-что настроить. На другом компьютере сети введите IP-адрес, показанный в окне *FreeNAS* (т.е. на последнем шаге инструкции с предыдущей страницы). В нашем случае web-интерфейс доступен по адресу <http://192.168.56.101>, поэтому мы вводим его в браузер на другом компьютере и видим интерфейс, показанный на первом экранном снимке.

Если вы видите нечто похожее, поздравляем – теперь можно отключить от этого компьютера монитор и клавиатуру, поставить его в уголок и забыть о нем (с точки зрения устройств). Отныне его можно настраивать через web-интерфейс. А если вы не можете подключиться к web-интерфейсу, проверьте сетевые настройки компьютера *FreeNAS* (компьютер попытается получить адрес по DHCP, но если это не сработает, вам придется ввести его вручную). Введите '1' для настройки сетевого интерфейса и следуйте указаниям. Если застряли, загляните в документацию на **LXFDVD** ([freenas8.0.3\\_guide.pdf](#)) или на сайте <http://doc.freenas.org>.



➤ Web-интерфейс *FreeNAS* позволяет настроить сервер, не копаясь в командной строке.

### «FreeNAS позволяет представить несколько дисков как один том.»

Вернемся к web-интерфейсу. Проницательный зритель увидит мигающий красный свет в правом нижнем углу. Щелкните, и вы увидите напоминание о том, что нужно задать пароль для учетной записи администратора. Если в сети есть недоверенные пользователи, сделайте это обязательно – вы вряд ли захотите, чтобы какой-то подлый тип дорвался до управления сервером. В раскрываемся меню слева выберите Account [Учетная запись], My Account [Моя учетная запись], затем Change Password [Изменить пароль].

### Назначаем жесткие диски

Следующий этап – выбрать жесткий(е) диск(и) для хранения данных. *FreeNAS* очень гибок и позволяет представить несколько жестких дисков как один том – то есть, хотя файлы будут находиться на разных физических дисках, для других пользователей сети они будут находиться в одном месте. В панели слева выберите Storage > Volumes > Create Volume [Хранилище > Тома > Создать том], и откроется окно создания тома.

Задайте имя тома и выберите тип файловой системы, которой хотите воспользоваться. Как мы уже упоминали, UFS – стандартная файловая система FreeBSD без лишних наворотов, и она прекрасно подходит для небольших серверов. Если у вас один жесткий диск и 4 Гб оперативной памяти или менее, это будет наилучшим выбором.

### Для честолюбцев...

Если у вас более мощный сервер с несколькими дисками и более 4 Гб оперативной памяти, выбирайте ZFS. Эта файловая система обладает гораздо большей гибкостью за счет более высоких системных требований, и мы подробно рассмотрели ее во врезке «Знакомимся с ZFS» внизу.

Пока мы остановимся на UFS. Выберите диски, которые нужно связать с этим томом, на панели Member Disks (Диски тома). Схема именования дисков во FreeBSD немного отличается от схемы Linux, поясним вкратце: диски IDE начинаются с 'ad', а диски SCSI – с 'da'; число в конце – номер диска. Например, первый диск в системе с двумя IDE-дисками – ada0.

После выбора диска нажмите Add Volume [Добавить том], и появится индикатор создания раздела. Когда оно будет закончено, слева в разделе Storage [Хранилища] появится новый том с именем /mnt/myshare (или с тем именем, которое вы задали). Этот путь пригодится нам позже. Мы связали с ним несколько

## Знакомимся с ZFS

ZFS отлично подходит для мощных серверов. Начавшая свою жизнь в операционной системе Solaris, эта файловая система теперь есть в различных версиях Unix (кстати, и в Linux – в сторонних разработках).

Ее возможности можно увидеть непосредственно во *FreeNAS*, если выбрать ее файловой системой тома при нескольких дисках. Например, в Extra options [дополнительных опциях] ZFS рядом с дисками вы найдете кэш. Тогда можно выбрать

традиционный жесткий диск большого размера в качестве главного хранилища и SSD (небольшой, но гораздо более быстрый) в качестве кэша. Это значительно ускоряет работу сервера, если у вас есть ограниченный набор файлов, обращение к которым производится очень часто; хотя если у вас не очень быстрая сетевая плата, большой разницы не будет.

Если отключить Extra options ZFS и выбрать все диски, вы увидите несколько очень полезных опций:

Mirror [зеркало] и Stripe [полосы] для двух дисков, а также RAID-Z (для трех или более дисков).

Желая объединить все диски в один большой том, как обычно и делается в NAS, выберите опцию Stripe. Если вам чрезвычайно важна надежность, выберите RAID-Z – свободного места будет меньше, зато при выходе из строя одного диска другие диски придут на помощь, и вы не потеряете никаких данных.

➤ Пропустили номер? Узнайте на с. 104, как получить его прямо сейчас.



жестких дисков для хранения данных, и теперь нужно точно определить, как хранить данные.

Нажмите на кнопку Change Permissions [Изменить права доступа] под именем папки слева, выберите опции Write [Запись] для группы и остальных пользователей (Group и Other) и нажмите Change [Изменить]. Тогда все смогут читать и записывать на наш том, и это прекрасно подходит для домашнего сервера; а если вам нужен более тонкий контроль, с учетными записями пользователей и правами доступа, загляните в третий раздел руководства.

## Делиться — это хорошо

Теперь можно предоставлять общий доступ к данным через сеть по различным протоколам: NFS (характерен для UNIX-подобных систем), AFP (используется в OS X) и CIFS (также известен как Samba и широко применяется в мире Windows). Для нашего урока мы возьмем CIFS — нет, не убивайте нас, хотя от него пахнет Microsoft. Дело в том, что почти все устройства, имеющие сетевой порт, могут работать с общими ресурсами CIFS, поэтому ради простоты и совместимости есть смысл выбрать его.

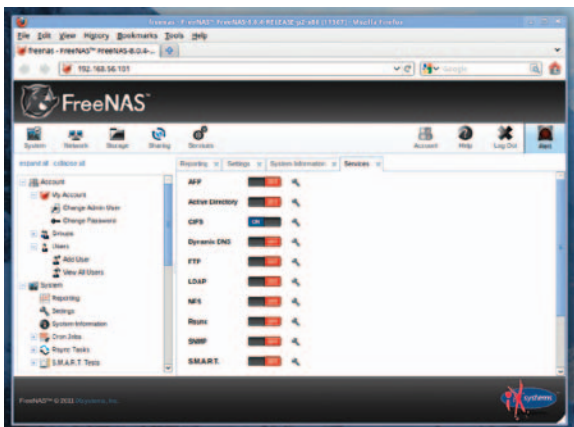
Для начала щелкните по иконке Services [Сервисы] в верхней части окна web-интерфейса, затем нажмите на переключатель рядом с CIFS, чтобы включить его. Затем щелкните по иконке с гаечным ключом — откроется окно Advanced Settings [Дополнительные настройки] — и установите галочку Allow Guest Access [Разрешить гостевой доступ]. Прокрутите окно вниз и нажмите OK.

Затем откройте подменю Sharing [Общий доступ] на панели слева, кликните на CIFS Shares [Общие папки CIFS], затем нажмите Add CIFS Share [Добавить общий каталог CIFS]. Задайте его имя, под которым он будет виден в сети (не используйте веселые символы, сбивающие с толку клиентское ПО). В поле Path [Путь] введите путь до тома, заданный ранее (/mnt/myshare) и установите галочку Allow Guest Access [Разрешить гостевой доступ]. Нажмите OK. Все готово!

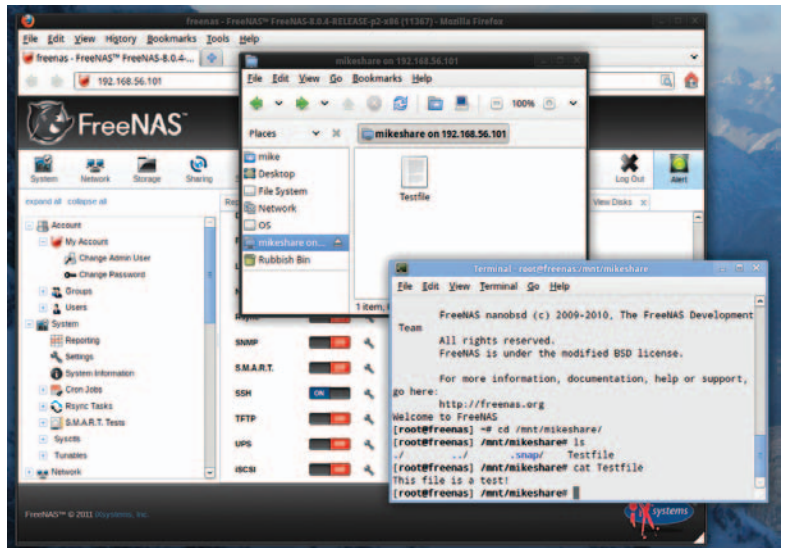
## Активация других сервисов

На нашем уроке мы сосредоточились на CIFS, но FreeNAS может общаться с внешним миром и по множеству других протоколов. Для начала, особенно если вы разбираетесь во FreeBSD, стоит включить OpenSSH, чтобы при необходимости заходить в систему на компьютере с NAS (не подключая к нему клавиатуру и монитор).

Нажмите кнопку Services [Сервис] на верхней панели, затем включите SSH соответствующей кнопкой. Если вы хотите разрешить непосредственный вход в систему от имени root (т.е. не как обычный пользователь с su), нажмите на гаечный ключ и выберите Login as Root [Входить в систему от имени root]. Теперь можно подключаться к нему по SSH с любого компьютера сети.



➤ Включать дополнительные сервисы во FreeNAS очень легко — чаще всего вы обойдетесь одним щелчком.



➤ Настройка удалась: общая папка создана. Мы смонтировали ее в Nautilus, подключились к ПК с FreeNAS по SSH и убедились, что загруженный нами файл там есть.

Если вы захотите передать несколько файлов с одного компьютера на компьютер с NAS, в списке сервисов можно мигмом включить FTP-сервер. В меню Spanner Settings [Настройки] можно разрешить вход в систему пользователю root (и вам не придется создавать учетные записи обыкновенных пользователей) или анонимный вход, если вам нечего беспокоиться о безопасности.

Если вы планируете открывать web-интерфейс FreeNAS через Интернет, настоятельно рекомендуем включить поддержку HTTPS для повышения безопасности. На левой панели выберите System > Settings [Система > Настройки] и выберите HTTPS в выпадающем списке Protocol [Протокол]. А если вы опасаетесь, что кто-то подключит клавиатуру к серверу NAS и сделает нечто гадкое, зайдите на вкладку Advanced [Дополнительно] и снимите галочку Enable Console Menu [Включить меню консоли]. После этого настройки можно будет менять только через web-интерфейс.

## Проверка сервера

Теперь, если все хорошо, вы сможете открыть общую папку CIFS/Samba с любого компьютера сети. Смонтируйте ее в любимом



В web-интерфейсе FreeNAS можно настроить задания Cron, т.е. команды, выполняемые каждую минуту, каждый час или раз в день. Выберите пункт меню System > Cron Jobs > Add Cron Job [Система > Задания Cron > Добавить задание Cron] и введите команду, которую нужно запустить. Затем задайте интервал запуска.

«Ваш ПК с FreeNAS будет тихо работать хранилищем и точкой доступа для музыки и фильмов.»

файловым менеджере Linux или попробуйте выполнить следующую команду (сперва нужно установить пакет smbfs):

```
smbmount //192.168.56.101/mikeshare /mnt/netshare
```

Измените IP-адрес и имя папки на свои и убедитесь, что каталог, в который вы монтируете, уже существует. Если у вас запросят пароль, просто нажмите Enter. Теперь можно в свое удовольствие загружать файлы в общую папку и удалять их.

Ваш ПК с FreeNAS будет тихо и послушно работать хранилищем и точкой доступа для музыки, фильмов и всего остального, что вы захотите разделить между несколькими компьютерами, и благодаря надежности старой доброй FreeBSD вам не придется беспокоиться о том, что он испустит дух.

FreeNAS невероятно гибок, и в этом кратком руководстве мы коснулись только основных его возможностей; если вы хотите узнать больше о правах доступа пользователей/групп, репликации тома, отчетах и других возможностях, загляните в руководство. Удачи! LXF



# Arduino: СВОЙ ЧАТ-

Кому нужны люди, когда можно пообщаться со своим преданным другом Arduino? **Ник Вейч** вступает в разговор.



## Наш эксперт

Когда **LXF** только появился, его держали на плаву исключительно скрипты *Bash* от **Ника Вейча**. Потом их заменили «людьми», и это, по мнению Ника, стало шагом назад...

**Е**сли что-то не подключено к Интернету, существует ли оно на самом деле? Проектов с Arduino, не чувствующих себя одиноко, в общем, хватает, но многие из них можно было бы улучшить, если бы у них было постоянное подключение к сети и вы могли бы взаимодействовать с ними отовсюду. На нашем уроке мы рассмотрим два способа подключения Arduino, чтобы с ним можно было общаться по телефону, с ноутбука или другого устройства, подключенного к Интернету.

## Первый вариант — имитация

Если Arduino нужно подключить к сети, самый простой вариант — симитировать ее. Часто плата Arduino подключена к USB-кабелю, другой конец которого подключен к компьютеру, а тот, в свою очередь, скорее всего подключен к сети. С помощью последовательного интерфейса очень просто взаимодействовать с Arduino из приложения на этом компьютере и воспользоваться тем же приложением для дальнейшей передачи полученной информации в любое место.

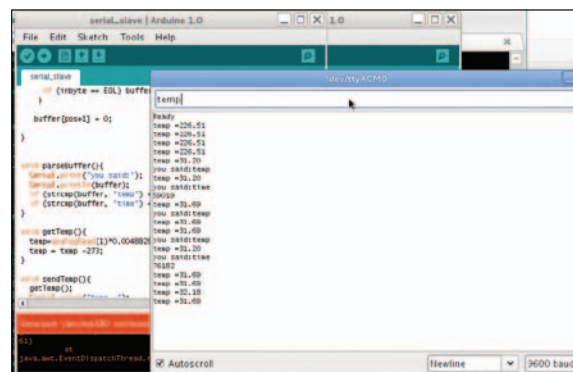
Существует огромное количество проектов, которые с помощью данной технологии заставляют Arduino обновлять web-страницы, и хотя это, безусловно, может быть полезно, но выглядит слегка банально. А не превратить ли Arduino в чат-бота, заставив его передавать информацию по протоколу мгновенного обмена сообщениями вроде *Jabber* (XMPP) — как в Google Chat?

Все, что нам нужно в этом случае — программа, которая будет работать на подключенном к Arduino компьютере и передавать сообщения. Есть несколько вариантов ее реализации, и, пожалуй, самый простой из них — воспользоваться Python (если

вы не знакомы с ним, не огорчайтесь — с минимальными изменениями в коде вы сможете запустить клиентскую программу с нашего DVD).

Понадобится указать данные своей учетной записи *Jabber* (на сайте [Jabber.org](http://jabber.org) за пару секунд можно завести новую — это проще, чем завести учетную запись в Gmail) и данные (т.е. адрес учетной записи XMPP, к которой нужно подключиться). Также потребуется установить начальное соединение между этими учетными записями — проще всего запустить какую-нибудь программу для обмена сообщениями, например, *Pidgin*, и инициировать соединение. Когда соединение будет принято обеими сторонами, вы сможете переписываться с помощью новой учетной записи *Jabber* из Google Chat или Facebook (там тоже используется XMPP).

```
import serial
import xmpp
# you will need to set these
ser = serial.Serial('/dev/ttyACM0', 9600)
jidname="evilbot1@jabber.org"
controller="someone@gmail.com"
pwd="xxxxxxxxxyxyxyxy"
def on_message(connection, message):
    global ser
    print "получено сообщение"
    txt=message.getBody()
    if (txt != None):
        ser.write(message.getBody())
    print "Соединение с jabber"
jid=xmpp.protocol.JID(jidname)
client=xmpp.Client(jid.getDomain(), debug=[])
client.connect()
client.auth(jid.getNode(), pwd)
client.sendInitPresence()
```



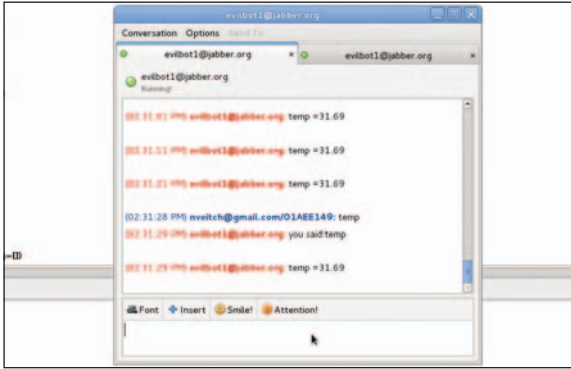
➤ Код Arduino использует последовательное соединение, и проверить его без подключения к Сети поможет утилита *Serial Monitor* в среде разработки Arduino.

## Что нам надо

- » ПО Arduino версии 1
- » Arduino Duemilanova или Uno
- » Для настоящего Ethernet-соединения — Wiznet 5100 или Ethernet-интерфейс к Arduino на базе ENC28J60

» **Месяц назад** Мы превратили свой Arduino в генератор волновых форм.

# БОТ ДЛЯ БЕСЕД



» С помощью программы на основе *Jabber* можно отправлять сообщения из любой программы для обмена сообщениями по XMPP, например, *Pidgin* или *Google Talk*.

```
presence = xmpp.Presence(status = 'Running!', show = 'chat',
priority = '1')
client.send(presence)
client.send(xmpp.protocol.Message(controller, "Жду ваших
команд"))
client.RegisterHandler("message", on_message)
while True:
    client.Process()
    if ser.inWaiting():
        print "получен последовательный сигнал"
        #serial input from device
        txt=ser.readline()
        #print txt
        client.send(xmpp.protocol.Message(controller, txt))
```

Программа на самом деле очень проста. Она устанавливает последовательное соединение с Arduino, затем подключается к серверу *Jabber*. После подключения учетная запись авторизуется путем указания пароля, и будет можно отправлять и принимать сообщения чата. Затем программа зацикливается, ожидая событий. При получении сообщения с последовательного порта Arduino она отправит его на сервер *Jabber*. Аналогично, при получении сообщения от подключенного пользователя (нет никаких причин ограничиваться одной учетной записью чата) оно будет перенаправлено Arduino по последовательной связи.

Однако методология немного отличается. Модуль XMPP для Python использует вызовы ответчика для обработки сообщений (ответчик – это функция, вызываемая автоматически при возникновении определенного события), что немного упрощает жизнь – но следует помнить, что на клиентской стороне стоит вызывать метод **Process()**, иначе сообщения будут просто накапливаться. Лучше всего делать это в цикле, в котором мы также будем проверять наличие последовательной связи с устройством. При получении сообщения *Jabber* при вызове **client.Process()** новое сообщение будет обнаружено и отправлено функции-ответчику, которую мы зарегистрировали для него (**on\_message**). Эта функция извлечет из пакета сообщения текст и отправит его через

## Основы работы сети

Сеть устроена сложно. Вся идея отправки пакетов данных по проводам (по крайней мере, так работают традиционные сети) реализуется с помощью нескольких аппаратных и программных слоев, выполняющих различные функции. Два самых нижних уровня – РЛУ (физический слой, т. е. проводники и компоненты для перевода сигнала в «единицы» и «нули») и MAC (Media Access Control [Контроль доступа к среде] – что-то вроде привратника физического слоя, который определяет, что поступает внутрь и уходит наружу) – обрабатываются на аппаратном уровне. Упоминаемые здесь микросхемы поставляются с библиотеками и прошивками, реализующими несколько следующих слоев (сетевой протокол и разрешенные типы пакетов данных), и чтобы программа заработала,

особенных действий не требуется. Но вот несколько терминов, которые нужно знать:
 

- » **MAC-адрес** Уникальный номер, идентифицирующий устройство и обычно «зашитый» в устройство; но для Arduino надо создать свой (и убедиться, что нет конфликта с другими MAC-адресами сети).

- » **IP-адрес** Десятичные числа, разделенные точками, идентифицирующие узел сети, например, 192.168.0.1.

- » **DNS** Сервер доменных имен напоминает большую адресную книгу – он связывает имена серверов (например, [linuxformat.com](http://linuxformat.com)) с адресами в Интернет.

Большая часть сетевого функционала поставляется в библиотеках, но помните, что Arduino – микроконтроллер с ограниченным объемом оперативной и постоянной памяти.

последовательный порт Arduino. Программу можно развить, добавив фильтры для приема только тех команд, которые понимает Arduino, или заменив команды байтовыми кодами, которые будет проще обрабатывать. Но все это остается на ваше усмотрение – ведь программа, кроме всего прочего, должна оставаться как можно более прозрачной.

На Arduino нам нужен код, который откроет последовательное соединение (и отправит сигнал сообщения коду на Python, что все запущено). Для этой проверки представьте себе, что на аналоговом порте 1 есть температурный датчик, который нам нужно проверять. Прежде всего объявим несколько переменных и инициализируем последовательный порт:

```
float temp;
char buffer[32];
uint32_t count;
void setup() {
    Serial.begin(9600);
    Serial.println("Готов");
}
```

Пока все очевидно. Для главного цикла мы настроим механизм вывода некоторых данных через равные интервалы времени. Но мы также будем слушать последовательный порт на наличие команд, подлежащих обработке.

```
void loop() {
    count = millis();
    while ((millis()-count)<10000){
        if (Serial.available()){
            readBuffer();
            parseBuffer();
        }
    }
}
```

### Скорая помощь

Jabber не является на 100% надежным протоколом – сообщения иногда теряются, поэтому убедитесь, что программа это переживет!

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

```

}
sendTemp();
delay(200);
}

```

Пока цикл будет продолжаться меньше заданного количества времени, он будет проверять функцию **Serial.available()**, которая вернет **true** при появлении данных во входном буфере (т.е. кто-то отправляет данные). В этом случае вызовутся таинственные функции для чтения этих данных и их преобразования в команды. Когда счетчик дойдет до конца, он вызовет еще одну таинственную функцию, затем **loop()** завершится и начнется снова. Итак, все что нам осталось – реализовать таинственные функции.

Самая сложная из них – чтение буфера, но даже она сложна не чрезмерно. Мы будем считывать в буфер по одной строке. Это значит, что нам нужно получать данные из последовательного устройства по одному байту, сохранять их в массиве **buffer**, для которого мы создали глобальную переменную в первой части кода (32 символа вам должно хватить!), пока не встретится символ конца строки и она не завершится.

```

void readBuffer(){
  int pos;
  int inbyte;
  const int EOL = 13; //Строки заканчиваются CR
  inbyte = Serial.read();
  delay(100);
  pos=0;
  while (Serial.available() > 0){
    buffer[pos] = inbyte;
    pos++;
    if (pos>(sizeof(buffer)/sizeof(char))) break;
    inbyte = Serial.read();
    if (inbyte == EOL) buffer[pos]=0;
  }
  buffer[pos+1] = 0;
}

```

Код может показаться слегка непростым, но это не так. Переменная **pos** содержит количество символов во входных данных. При получении каждого байта она увеличивается на единицу. Поэтому цикл **while** просто продолжит считывать байты, пока

не встретит символ конца строки (код ASCII – 13). Единственное, что мы добавили – завершение считывания, если количество символов во входных данных превышает размер буфера. Память в Arduino не защищена, и если вы начнете записывать в пространство за пределами того, для которого выделили место, то моментально сотрете что-нибудь важное.

Считав команду, ее нужно обработать. Так как мы закончили нулем строку в переменной **buffer**, то она представляет собой обычную строку на C. Это удобно, потому что мы сможем воспользоваться стандартной функцией сравнения строк, чтобы понять, является ли эта строка командой:

```

void parseBuffer(){
  Serial.print("you said:");
  Serial.println(buffer);
  if (strcmp(buffer, "temp") == 0) sendTemp();
  if (strcmp(buffer, "time") == 0) Serial.println(millis());
}

```

В данном случае мы реализовали две команды: одну, которая будет считывать температуру, и другую, которая вернет время работы Arduino. Конечно, при желании можно реализовать и более сложные вещи – разбирать параметры или использовать команды для управления выводами вместо чтения значений датчика. Оставляю вашему воображению представить то, что могут сотворить армии сетевых роботов-разрушителей. Последний кусочек мозаики – функции, которые считывают данные аналогового датчика и отправляют их на последовательное устройство:

```

void getTemp(){
  temp=analogRead(1)*0.004882812*100;
  temp = temp -273;
}
void sendTemp(){
  getTemp();
  Serial.print("temp =");
  Serial.println(temp);
}

```

Здесь нет никаких хитростей. Помните, что для последовательного вывода нужно использовать функции **print()** и **println()**, так как для обозначения конца строки используется символ конца строки. Загрузив этот код в Arduino, вы сможете последовательно проверить его утилитой Serial Monitor, прежде чем отправлять информацию по сети.

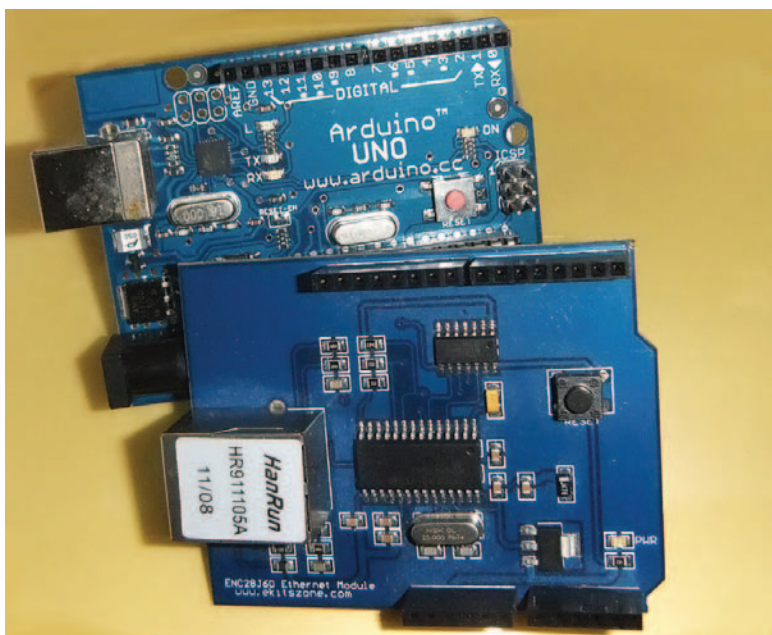
## Второй вариант — реальность

XMPP – прекрасный протокол обмена сообщениями. Он используется не только в клиент-серверных программах мгновенного обмена сообщениями вроде Google Chat, но и для передачи пакетов данных между серверами в Интернете.

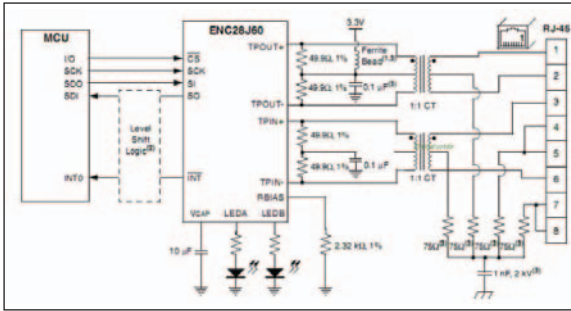
К сожалению, XMPP сложноват для создания подключенного к сети Ethernet коммуникатора «все в одном» на базе Arduino. В ограниченной по объему памяти Arduino можно развернуть клиент XMPP, но особенности протокола (такие как XML для всего) означают, что большую часть времени Arduino будет тратить на работу системы обмена сообщениями, не выполняя никакой полезной работы. Накладные расходы на XMPP слишком велики. Но не душите мечту – есть другие варианты!

Подключить Arduino к сети не так сложно, как вы могли бы подумать: существуют микросхемы, предназначенные для подключения микроконтроллера к сети Ethernet и реализующие аппаратную часть протокола. Сообщество Arduino использует для этих целей микросхему Wiznet 5100, в основном потому, что она использовалась в версиях Arduino с поддержкой Ethernet и в некоторых официальных версиях Ethernet-интерфейсах Arduino. Она

➤ Существует масса Ethernet-интерфейсов для Arduino. При покупке убедитесь, что это схема Wiznet/Arduino или что она поставляется с актуальной версией библиотеки для Arduino.



➤ Пропустили номер? Узнайте на с. 104, как получить его прямо сейчас.



» Собственную схему создать *можно*, но из-за стоимости и природы компонентов (пять элементы в корпусе ИС рискованно) проще сэкономить время и усилия и купить готовую.

также хорошо подкреплена прошивками, поскольку библиотека драйверов для нее включена в основную поставку Arduino. Альтернативный вариант – Microchip ENC28J60. Существует много сборочных плат от сторонних производителей, и они могут быть гораздо дешевле (что-то вроде £5 против 20) по сравнению с платами от Arduino/Wiznet. Купив такую плату, можно собрать схему самому. Ее схема показана ниже, но гораздо проще купить сборочную плату.

## IRC: чат для настоящих роботов

IRC существует почти столько же, сколько Интернет. Это простой и эффективный способ оскорблять людей онлайн или заигрывать с ними – в зависимости от ваших предпочтений. Он также очень прост в реализации, потому что появился во времена, когда последовательное соединение и мигающий зеленый курсор были нормой.

Об этом даже не придется много думать – все очень просто и делалось уже не один раз. Одна из самых лучших и самых простых реализаций принадлежит Кейрану «Аффиксу» Смиту [Keiran 'Affix' Smith], ею мы и воспользуемся.

Первым делом подключим необходимые библиотеки. Если вы пользуетесь enc28j60, замените имя соответствующей библиотеки:

```
#include <SPI.h>
#include <Ethernet.h>
```

Несколько глобальных переменных позаботятся о настройках. MAC-адрес, как вы, наверное, знаете, должен просто представлять собой уникальное число. IP-адрес нужно задавать по отношению к своей локальной сети. Адрес сервера представляет собой адрес IRC-сервера, которым мы хотим воспользоваться. Все эти адреса хранятся в массивах – так к ним проще обращаться. Также нам нужны несколько строк для хранения команд и текста, которым мы будем обмениваться с сервером:

```
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
byte ip[] = { 192,168,0,28 };
byte server[] = { 38,229,70,20 }; // Freenode
int port = 6667;
String chan = "#arduino";
String nick = "evilbot";
String join = "JOIN ";
String nickcmd = "NICK ";
String user = "USER ";
String out = "TEMP=";
EthernetClient client;
```

Клиент представляет собой класс из библиотеки Ethernet, и с ним проще подключиться к определенному серверу/порту и обмениваться данными. У него есть методы, которые имитируют

последовательное подключение, так что для отправки информации будет можно использовать методы `print()` и `println()`. Установить соединение тоже просто:

```
void setup() {
  Ethernet.begin(mac, ip);
  Serial.begin(9600);
  delay(200);
  Serial.print("connecting to IRC... ");
  if (client.connect(server,port) ) {
    Serial.println("connected!");
    client.println(nickcmd.concat(nick));
    client.println(user.concat(nick));
    client.println();
  }
  else {
    Serial.println("connection failed");
  }
}
```

Основное здесь – инициализация устройства Ethernet (номера контактов указывать не нужно, так как оно всегда остается в одном и том же месте – на контактах 10,11,12,13). Затем клиент подключается к указанному порту заданного сервера. Этот код использует класс EthernetClient, который в новых библиотеках для релиза 1.0 называется просто 'client', поэтому следите за этим, если попытаетесь адаптировать старый код.

Затем основной цикл просто считывает все входящие данные в ожидании ключевого слова, а когда находит его, отвечает запрашиваемыми данными.

```
void loop()
{
  if (client.available()) {
    client.println(join.concat(chan));
    while(true)
    {
      String data;
      data = client.read();
      if(data.startsWith("TEMP"))
      {
        temp=analogRead(1)*0.004882812*100 -273;
        client.println(out.concat(String(temp)));
      }
    }
  }
}
```

Скорее всего, вам захочется сделать что-то еще – как видите, подключение реализуется очень просто, а с помощью строк относительно легко реализовать систему команд и ответов.

## Идем дальше

На следующем уроке мы посмотрим, как реализовать другие удобные сетевые возможности. Мы вернемся ко входам и исследуем различные способы реализации клавиатуры! **LXF**

## Arduino 1.0

Если вы пропустили предыдущий урок, из которых – отличие в расширении файлов-напомню, что теперь в этих руководствах используется версия 1.0 кода Arduino. Между версиями есть несколько мелких различий, не самое незначительное – специально для вас мы записали новую версию на DVD.

» **Через месяц** Пора взяться за дело и исследовать клавиатуры.



# Erlang: О пользе

**Андрей Ушаков** проходимся по функциям, сильно облегчающим жизнь многозадачным и распределенным системам.



**Наш эксперт**

**Андрей Ушаков** активно приближает тот день, когда функциональные языки станут мейнстримом.

**В** нескольких последних номерах мы говорили о принципах построения многозадачных, отказоустойчивых и распределенных систем. И мы, естественно, затрагивали тему о библиотечных функциях, упрощающих нам жизнь. Данная статья посвящена как раз этой теме: мы рассмотрим большую часть библиотечных функций, о которых пока не говорилось.

Вернемся немного назад и обсудим хранение состояния между вызовами функций. Одно из основных положений функционального программирования – то, что результат выполнения любой функции зависит только от ее аргументов. Передавая одни и те же аргументы несколько раз, мы должны получить один и тот же результат. А значит, функция не должна помнить какое-либо состояние между ее вызовами. Что делать, если некоторое состояние между вызовами функции все же необходимо хранить? Ответ очевиден: переложить ответственность по хранению состояния на вызывающую сторону и передавать состояние в функцию как один из аргументов. Основной строительной единицей многозадачных приложений в языке Erlang являются процессы. «Сердце» процесса Erlang – обычно цикл обработки сообщений, представляющий собой функцию – обработчик сообщений. Эта функция вызывает сама себя (это единственный способ в языке Erlang создать конечный или бесконечный цикл) при помощи хвостовой рекурсии. Соответственно, в таком случае цикл – обработчик сообщений будет и вызывающей, и вызываемой сторонами, и за передачу состояния будет ответственна функция-обработчик. Естественно, что при создании процесса и первом вызове функции-обработчика сообщений (произведенного извне), ответственность за передачу начального состояния лежит на стороне, создающей процесс.

Описанный подход к хранению состояния является строго функциональным. Среда выполнения Erlang содержит и другой подход к хранению состояния: это использование памяти, приданной процессу. С каждым процессом связан некий объем памяти в виде изменяемого словаря; доступ к этой памяти имеет только код, выполняемый процессом. Если два разных процесса выполняют один и тот же код, который обращается к памяти, связанной с процессом, то этот код будет обращаться к разным областям памяти, в зависимости от того, какой из процессов его выполняет. Для работы с такой памятью в языке Erlang определен целый ряд функций (а точнее, BIF). Сохранить некоторое значение **Value** и ассоциированный с ним ключ **Key** (при помощи которо-

го будет осуществляться доступ к этому значению в дальнейшем) можно при помощи функции **put(Key, Value)**. Если с данным ключом уже было ассоциировано какое-либо значение, оно будет возвращаемым значением функции **put/2** (иначе возвращаемым значением будет атом **undefined**). Для получения значения **Value** по ключу **Key** служит функция **get(Key)**, которая возвращает либо значение **Value**, ассоциированное с ключом **Key**, либо атом **undefined**. Мы можем получить все данные сразу, в виде пар ключ–значение: для этого служит функция **get/0**. Помимо этого, можно получить список всех ключей, ассоциированных со значением **Value**, при помощи функции **get\_keys(Value)**. И, наконец, чтобы удалить некоторое значение по ключу (и ассоциированный с ним ключ), используется функция **erase(Key)**; чтобы полностью очистить память, связанную с процессом – функция **erase/0**.

Следует сразу сказать, что использование памяти, связанной с процессом и этим набором функций, имеет два больших минуса по сравнению с «классическим» функциональным подходом. Во-первых, можно легко написать функцию, возвращаемое значение которой при одних и тех же аргументах будет различаться, что нарушает положение функционального программирования о том, что результат выполнения функции должен зависеть только от ее аргументов. Конечно, есть ряд функций, которые это положение нарушают, но все эти функции являются интерфейсом к внешнему миру – например, функция **now/0**, возвращающая текущее время. Во-вторых, использование памяти, связанной с процессом, создает невяную связь как между функцией и некоторым состоянием, так и между несколькими функциями. Это увеличивает связность и сложность приложения. В общем, использовать память, связанную с процессом, или нет – решать, конечно, вам.

Раз уж мы затронули функции, определенные в модуле **erlang** (а как мы помним, BIF также определены в этом модуле), давайте коротко рассмотрим те функции, о которых мы еще не говорили (естественно, это касается только функций, интересных с точки зрения создания многозадачных и распределенных приложений).

Начнем с разговора о лидере группы процессов. Все процессы в языке Erlang принадлежат той или иной группе, и в каждой группе есть процесс, называемый лидером этой группы. Весь ввод/вывод внутри группы направляется из/в лидера группы. Когда один процесс создает другой процесс, то у нового процесса будет тот же самый лидер группы (и та же самая группа), что и у родительского процесса. Для получения идентификатора лидера группы, которой принадлежит текущий процесс, следует использовать функцию **group\_leader/0**. Чтобы для группы, которой принадлежит процесс с идентификатором **Pid**, задать нового лидера, следует использовать функцию **group\_leader(GroupLeaderPid, Pid)**, где **GroupLeaderPid** – идентификатор процесса, который станет новым лидером группы.

Отметим также, что для управления группами процессов следует использовать функции из модулей **pg** или **pg2**. Модуль **erlang**, помимо вышеупомянутых функций, содержит также функции, реализующие функциональность таймера: **erlang:start\_timer/3**, **erlang:read\_timer/1**, **erlang:cancel\_timer/1** и **erlang:send\_after/3** (о них мы поговорим подробнее в одном из практикумов).

## Функции, использующие состояние

Примером функций, использующих состояние, сохраненное между их вызовами вызывающей стороной, являются функции для расчета значения хэша MD5 (определенные в модуле **erlang**). Функция **erlang:md5\_init()** инициализирует состояние, называемое контекстом **Context**, и возвращает его.

Функция **erlang:md5\_update(Context, Data)** принимает в качестве аргументов старое состояние **Context** и очередную порцию данных **Data**, и возвращает обновленное состояние. Функция **erlang:md5\_final(Context)** принимает состояние **Context** и возвращает значение хэша MD5.

# библиотек

Следующий модуль, которым мы займемся – небольшой модуль **lib**, содержащий разнообразные полезные функции. Из этих функций для нас интересны (в ключе нашей статьи) в первую очередь две: **lib:flush\_receive/0** и **lib:sendw/2** (также этот модуль содержит функцию **lib:send/2**, полностью идентичную функции **erlang:send/2**). Функция **lib:flush\_receive/0** служит для очистки очереди сообщений процесса от необработанных сообщений. Мы помним, что каждый процесс содержит собственную очередь сообщений. Когда мы посылаем сообщение некоторому процессу, оно помещается в очередь сообщений этого процесса.

Выражение **receive** извлекает первое подходящее сообщение из очереди сообщений, оставляя остальные сообщения в очереди. Поэтому вполне возможно возникновение ситуации, когда количество сообщений в очереди какого-либо процесса постоянно увеличивается, что, в конечном итоге, приведет к замедлению вычисления (выполнения) выражения **receive** и переполнению очереди сообщений. Такая ситуация возможна, если процесс получает сообщения, не подходящие ни для одного выражения **receive**, либо если сообщения приходят быстрее, чем процесс их обрабатывает. Первого случая можно избежать, если в выражении **receive** всегда добавлять в конец вариант,

обрабатывающий все сообщения. Однако этот подход не будет работать, если вид выражения **receive** (варианты обрабатываемых сообщений) может изменяться (например,

в зависимости от некоторого состояния). В этом случае (а также в случае переполнения очереди сообщений) можно воспользоваться функцией **lib:flush\_receive/0**. Чтобы узнать, сколько же сообщений содержится в очереди сообщений процесса, необходимо использовать **BIF process\_info(Pid, [message\_queue\_len])**, где **Pid** – идентификатор процесса, размер очереди сообщений которого мы хотим узнать.

Мы также можем получить список всех необработанных сообщений в очереди сообщений процесса, вызвав **process\_info(Pid, [messages])**. Функция **lib:sendw(To, Msg)** отсылает сообщение процессу **To** и ожидает ответного сообщения, которое она и возвращает. Эта функция позволяет упростить написание кода для синхронизации взаимодействия между процессами, когда требуется послать сообщение процессу, дождаться от него ответа и продолжить выполнение задачи.

У этой функции есть два недостатка, о которых необходимо помнить при ее использовании. Во-первых, она ждет ответа от другого процесса бесконечно долго; если ответ от другого процесса по каким-либо причинам не придет ожидающему процессу, то ожидающий процесс будет заблокирован навсегда. Во-вторых, эта функция обрабатывает первое сообщение от любого процесса, которое придет после отправки сообщения какому-то определенному процессу. Вполне возможна ситуация, что мы посылаем сообщение одному процессу, первым приходит сообщение от другого процесса, и именно его нам функция **lib:sendw/2** и возвратит.

Теперь поговорим про один из самых, пожалуй, полезных модулей для создания многозадачных приложений: это модуль **rpc**.

Функции из этого модуля упрощают выполнение таких операций, как вызов какой-либо функции (экспортируемой из некоторого модуля) на удаленной стороне, в отдельном процессе на некотором узле (или группе узлов). При создании любого узла на нем автоматически запускается **grpc-сервер**, с которым и взаимодействуют функции из этого модуля.

Начнем с функций, позволяющих выполнить некоторую функцию на удаленной стороне и вернуть результат выполнения этой функции: **rpc:call(Node, Module, Func, Args)** и **rpc:call(Node, Module, Func, Args, Timeout)**. Обе эти функции на узле **Node** в отдельном процессе выполняют функцию **apply(Module, Func, Args)**, т.е. вызывают функцию **Module:Func** с аргументами **Args** и возвращают результат этого вызова. Если вызов функции был успешен, то результат вызова возвращается напрямую; если же во время вызова возникла ошибка (исключение времени выполнения), то эти функции в качестве результата вернут кортеж (**badrpc, Reason**), где **Reason** – причина возникновения ошибки. Обе эти функции синхронные – то есть вызывающий процесс блокируется до получения результата. Единственное отличие между этими двумя функциями – первая функция ожидает результата бесконечно

долго, тогда как вторая – в течение отрезка времени, заданного параметром **Timeout**.

Функции **rpc:block\_call/4** и **rpc:block\_call/5** аналогичны двум предыдущим функциям, за одним исключением:

в отличие от предыдущих, они не создают новый процесс на узле **Node**, а выполняют вызов в процессе **grpc-сервера**. Это приводит к тому, что **grpc-сервер** на узле **Node** блокируется до окончания выполнения исходного вызова. Получается, что пока на узле **Node** вызов функции **rpc:block\_call/4** или **rpc:block\_call/5** не завершится, никакой другой вызов какой-либо функции из модуля **rpc** на узле **Node** выполняться не начнет. Однако есть некоторые тонкости, связанные с этими функциями. Предположим, что на узле **Node** мы инициировали выполнение функции **rpc:call/4** (или любой другой функции, отличной от функций **rpc:block\_call/4** и **rpc:block\_call/5**). После этого из другого процесса мы инициировали выполнение функции **rpc:block\_call/4** (или **rpc:block\_call/5**) опять же на узле **Node**, что приведет к блокированию **grpc-сервера** на узле **Node**. Но при этом вызов функции **rpc:call/4** продолжит выполняться на узле **Node**, т.к. он выполняется в своем процессе (а не в процессе **grpc-сервера**). Далее предположим, что вызов функции **rpc:call/4** завершается раньше вызова функции **rpc:block\_call/4**.

По завершении вызова функции **rpc:call/4** вызывающая сторона должна получить результат работы этой функции; однако результат передается не напрямую, а через **grpc-сервер**, что приведет к тому, что результат вызова функции **rpc:call/4** будет передан вызывающей стороне только после завершения вызова функции **rpc:block\_call/4**. Ну и никто не запрещает нам создавать процессы на узле **Node** при помощи одной из функций семейств **spawn/2,4** и **spawn\_link/2,4**, даже если в этот момент на узле **Node** выполняется вызов одной из функций **rpc:block\_call/4** и **rpc:block\_call/5**. »

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

## «Неподходящие» объекты языка

При использовании функций семейства `global:set_lock/1,2,3` для установления блокировки применяется идентификатор блокировки, имеющий вид `{ResourceId, LockRequesterId}`, где `ResourceId` – идентификатор запрашиваемого ресурса, `LockRequesterId` – идентификатор стороны, запрашивающей блокировку

на ресурс. В качестве `ResourceId`, так и `LockRequesterId` может выступать любой объект языка Erlang. Но на величину `ResourceId` есть ограничения: атомы `global`, `dist_ac`, `mnesia_adjust_log_writes`, `pg2`, `mnesia_table_lock` не годятся как идентификаторы ресурсов при установлении блокировки, и их применять не рекомендуется.

Предположим, что перед нами стоит задача выполнить удаленный вызов функции и получить результат этого вызова сразу на нескольких узлах. Конечно, мы можем ее решить, последовательно вызывая функцию `rpc:call/4` для каждого узла, где необходимо удаленно выполнить функцию и получить результат. Вполне очевидно, что такое решение неэффективно: каждый следующий вызов функции `rpc:call/4` мы сможем сделать только когда предыдущий вызов вернет значение, а задача, что очевидно, легко распараллеливается. Но нам нет нужды реализовывать такое параллельное решение; для этого модуль `rpc` содержит функции `rpc:multicall/4` и `rpc:multicall/5`. Функция `rpc:multicall(Nodes, Module, Func, Args, Timeout)` вычисляет значение функции `Module:Func` с аргументами `Args` на узлах из списка `Nodes` и возвращает результат в виде `{ResL, BadNodes}`, где `ResL` – список полученных результатов, `BadNodes` – список узлов, на которых вычисление функции окончилось неудачей. Параметр `Timeout` в функции `rpc:multicall/5` устанавливает максимальное время ожидания ответа от каждого из узлов. Если за время `Timeout` от какого-либо узла ответа не получено, узел попадает в список `BadNodes`, т. е. мы считаем, что вычисление функции на данном узле окончилось неудачей. Список полученных результатов `ResL` содержит результаты вычисления функции `Module:Func` с аргументами `Args` в произвольном порядке (относительно порядка узлов в списке `Nodes`) и без привязки к узлу, на котором этот результат был получен.

Если такая привязка нужна, то сама функция должна ее возвращать – например, в виде кортежа, состоящего из имени узла и вычисленного результата. Понятно, что функция `rpc:multicall/5` (и `rpc:multicall/4`) возвращает управление только тогда, когда будет сформирован результат, т. е. является синхронной функцией.

Все вышеописанные функции из модуля `rpc` были синхронными функциями, но вызов синхронной функции приводит к блокированию вызывающего процесса, что не всегда желательно. А значит, необходим асинхронный вызов функции на удаленном узле: для этого мы сначала вызываем функцию `rpc:async_call/4`, которая возвращает ключ, используемый потом для получения результата вызова при помощи одной из функций `rpc:yield/1`, `rpc:nb_yield/1` или `rpc:nb_yield/2`. Функция `rpc:yield/1` блокирует вызывающий процесс до получения результата (если результат вызова уже готов, то никакой блокировки не будет). Функция `rpc:nb_yield(Key, Timeout)` ждет готовности результата вызова согласно параметру `Timeout` и возвращает либо результат (точнее, кортеж `{value, Value}`, где `Value` – результат вызова), либо атом `timeout`; функция `rpc:nb_yield/1` эквивалентна вызову `rpc:nb_yield(Key, 0)`.

Давайте коротко поговорим об оставшихся функциях из модуля `rpc`. Если нам возвращаемое значение не нужно (обычно это

функции с некоторым побочным эффектом), то для удаленного вызова можно воспользоваться функцией `rpc:cast/4`, которая не блокирует выполнение вызывающего потока. Если необходимо вычислить значение функции, возвращаемое значение которой нас не интересует, на нескольких узлах, для этого можно воспользоваться одной из функций `rpc:eval_everywhere/3,4`.

Предположим, что на некотором наборе узлов у нас есть процессы (на каждом узле) с одинаковым локальным зарегистрированным именем `Name`. Если мы хотим послать всем этим процессам сообщение синхронно, то для этого модуль `rpc` содержит функцию `rpc:sbcast(Nodes, Name, Msg)` (а также функцию `rpc:sbcast/2`). Эта функция шлет сообщение `Msg` процессам с локальным именем `Name` на узлах из списка `Nodes` и возвращает кортеж `{GoodNodes, BadNodes}`, где `GoodNodes` – список узлов, на которых существует процесс с именем `Name`, `BadNodes` – список узлов, на которых процесса с именем `Name` нет. Эта функция гарантирует только, что сообщение `Msg` будет послано всем существующим процессам с локальным именем `Name` на узлах из списка `Nodes`, но не обработано этими процессами. Если мы хотим выполнить ту же самую задачу, но асинхронно, то к нашим услугам в модуле `rpc` есть функции `rpc:abcast/2` и `rpc:abcast/3`.

Предыдущие функции позволяли просто послать некоторое сообщение удаленному процессу и не ожидали никакого ответа от него (хотя, включив в сообщение идентификатор процесса отправителя, вполне можно реализовать и получение ответного сообщения от удаленного процесса). Если же наша задача состоит в том, чтобы послать сообщение и получить ответ, то для этой цели модуль `rpc` содержит еще ряд функций. Функция `rpc:server_call(Node, Name, ReplyWrapper, Msg)` предназначена для синхронной отправки сообщения `Msg` процессу с локальным именем `Name`, зарегистрированным на узле `Node`. При этом удаленный процесс должен ожидать сообщение в виде `{From, Msg}`, где `From` – идентификатор процесса инициатора, и отсылать ответное сообщение обратно в виде `{ReplyWrapper, Node, Reply}`. Здесь `Reply` – это результат выполнения запроса; это либо просто некоторый

## «Необходим асинхронный вызов функции на удаленном узле.»

объект языка Erlang, либо кортеж `{error, Reason}` в случае возникновения ошибки. Если мы хотим отослать сообщения нескольким процессам с локальным именем `Name`, зарегистрированным на разных узлах, то для этого в модуле `rpc` определены функции `rpc:multi_server_call/2` и `rpc:multi_server_call/3`.

И, наконец, для параллельного решения типичных задач в модуле `rpc` определены еще две функции: `rpc:parallel_eval/1` и `rpc:pmap/3`. Функция `rpc:parallel_eval/1` позволяет параллельно вычислить значения функций на узлах, соединенных с данным узлом, и возвращает результат этого вычисления в том же порядке, в каком шли исходные функции. Функция `rpc:pmap/3` является версией функции `lists:map/2`, работающей параллельно.

Займемся работой с узлами. В прошлом номере мы говорили о модуле `global`, который содержит функции для работы с глобальными зарегистрированными именами процессов (это функции `global:register_name/2,3`, `global:re_register_name/2,3`, `global:registered_names/0`, `global:unregister_name/1`, `global:whereis_name/1` и `global:send/2`). Помимо этого, данный модуль содержит функциональность для работы с глобальными блокировками. Глобальная блокировка – это блокировка на какой-либо ресурс (заданный идентификатором) либо на уровне узла, либо на уровне всех узлов, соединенных друг с другом. Для управления блокировками используется специальный идентификатор, который имеет

» Пропустили номер? Узнайте на с. 104, как получить его прямо сейчас.



вид `{ResourceId, LockRequesterId}`, где `ResourceId` – некоторый идентификатор ресурса, `LockRequesterId` – идентификатор того, кто запрашивает. И `ResourceId`, и `LockRequesterId` могут быть любыми объектами языка Erlang; идентификатор запрашиваемой стороны не обязательно должен быть идентификатором или зарегистрированным именем процесса. Мы поговорим более подробно о глобальных блокировках на следующем уроке, а сейчас просто рассмотрим предназначенные для этого функции. Чтобы установить блокировку, используется семейство функций `global:set_lock/1,2,3`. Чтобы снять блокировку, используется семейство функций `global:del_lock/1,2,3`. Наконец, у нас есть возможность установить блокировку и после успешного установления вычислить значение некоторой заданной функции: для этого определено семейство функций `global:trans/2,3,4`. Если при установлении блокировки окажется, что блокировка на запрашиваемый ресурс уже установлена кем-то еще, то процесс, во время выполнения которого мы устанавливаем блокировку, перейдет в состояние ожидания. Это состояние закончится, как только блокировка на запрашиваемый ресурс будет снята другим процессом и установлена нашим процессом. А если установление блокировки на ресурс ожидают несколько процессов, то блокировку установит любой из них (мы не можем полагаться на какой-либо порядок установления блокировки ожидающими процессами). Состояние ожидания может также закончиться, если будет превышено предельное время ожидания, передаваемое в качестве параметра (на самом деле передается не предельное время ожидания, а количество попыток установления блокировки), если мы не указали, что установления блокировки процесс должен ожидать бесконечно долго.

Последняя наша сегодняшняя тема – это создание и работа с вспомогательными дочерними узлами [slave nodes]. Пусть для создания распределенной системы нам необходимы несколько узлов, расположенных на разных компьютерах (хостах). Понятно, что развертывание такой инфраструктуры потребует некоторых добавочных усилий по ее администрированию и поддержке. Но если нас устраивает ситуация (а в большинстве случаев это так), что время жизни всех разворачиваемых узлов будет зависеть от времени жизни некоторого главного узла, и весь ввод/вывод будет идти через этот главный узел (консольный и файловый ввод/вывод), то мы можем сильно упростить себе жизнь.

Для этого нам достаточно создать только один главный узел руками, после чего в коде инициализации (или где-то еще) мы можем создавать вспомогательные дочерние узлы на определенных компьютерах (хостах). Это означает, что множество компьютеров и узлов на них мы зададим при помощи некоторого файла конфигурации. Вся эта функциональность реализована в модуле `slave`. Мы можем создать новый дочерний вспомогательный узел на компьютере (хосте) `Host`, с именем `Name` (имя создаваемого

узла будет `Name@Host`) и параметрами командной строки для запуска среды времени выполнения Erlang (процесса `erl`) при помощи семейства функций `slave:start/1,2,3`. Мы можем сделать то же самое при помощи семейства функций `slave:start_link/1,2,3`, но тогда время жизни созданного вспомогательного узла будет зависеть от вызывающего процесса. И, наконец, можно завершить работу узла при помощи функции `slave:stop/1`. Следует сказать, что для создания узлов на компьютерах (хостах), отличных от того, на котором запущен главный узел, используется утилита `rsh` (или при запуске главного узла при помощи ключа `-rsh Program` задается альтернатива утилите `rsh`).

С дочерними вспомогательными узлами (и модулем `slave`) связана еще одна близкая тема: пулы узлов. Это предопределенные наборы уже созданных узлов, предназначенные для обслуживания запросов клиентов на выполнение той или иной функции; при этом выбором узла и созданием процесса для выполнения занимается сам пул узлов. В качестве такого узла выбирается узел с наименьшей загрузкой. Создать пул узлов с именем `Name` можно при помощи семейства функций `pool:start/1,2`. Узел, на котором происходит вызов одной из функций `pool:start/1,2`, становится главным узлом пула. Вспомогательные узлы пула создаются при помощи одной из функций `slave:start/2,3`; файл `.hosts.erlang` определяет хосты, где будут созданы вспомогательные узлы (на каждом хосте из этого файла по одному узлу). Если файл `.hosts.erlang` в системе отсутствует, создание пула узлов завершится с ошибкой. Мы можем присоединить уже созданный узел к пулу при помощи функции `pool:attach/1`. При помощи функции `pool:stop/0` мы можем завершить работу пула (и «убить» все дочерние узлы). При помощи функции `pool:get_nodes/0` мы получим список всех узлов, составляющих пул, а при помощи функции `pool:get_node/0` – наименее загруженный узел из пула.

И, наконец, при помощи функций `pool:pspawn/3` и `pool:pspawn_link/3` создается процесс для выполнения функции `Module:Func` с аргументами `Args`; при этом выбором узла и созданием процесса занимается пул потоков. Функции `pool:pspawn/3` и `pool:pspawn_link/3` возвращают управление сразу после создания процесса; при этом они возвращают идентификатор созданного процесса. Это означает, что если надо получить значение функции, об этом следует позаботиться самому; например, в качестве `Module:Func` передавать некоторую функцию-обертку, которая внутри выполняет интересующую нас функцию и посылает нам ее значение через механизм сообщений.

В этой статье мы поговорили о библиотеках и функциях, которые предназначены для упрощения жизни нам, разработчикам многозадачных и распределенных приложений. По-моему, создателям языка Erlang это удалось. А в следующий раз мы рассмотрим интересную и важную тему синхронизации задач. **LXF**

## Хранение состояния в цикле обработки сообщений.

Рассмотрим пример некоторой гипотетической функции сообщений, демонстрирующей, как хранение состояния в цикле обработки сообщений:

```
message_handler(State) ->
  receive
    {m1, From} ->
      ReturnValue = some_func1(State),
      From ! {r1, ReturnValue},
      message_handler(State);
    {m2, Data} ->
      NewState = some_func2(State, Data),
      message_handler(NewState);
    _Other -> message_handler(State)
  end.
```

Приведенный пример содержит три обработчика сообщений: для сообщений `{m1, From}`, для сообщений `{m2, Data}` и для всех остальных сообщений.

При обработке сообщения `{m1, From}` мы вычисляем некоторое значение с использованием текущего состояния, возвращаем его инициатору запроса `From`, и вызываем рекурсивно сами себя, передавая текущее состояние `State` в качестве параметра. Видно, что в данном обработчике состояние не меняется.

При обработке сообщения `{m2, Data}` мы вычисляем новое состояние `NewState` с использованием старого состояния `State` и некоторых данных `Data`, после чего рекурсивно вызываем сами себя,

передавая новое состояние `NewState` в качестве параметра.

Последний обработчик демонстрирует нам, как при помощи механизма соответствия шаблону (pattern matching) мы можем обрабатывать (в нашем случае просто извлекать из очереди сообщений) все остальные сообщения. В этом обработчике мы с сообщением ничего не делаем, а просто вызываем рекурсивно сами себя, передавая текущее состояние `State` в качестве параметра.

И, наконец, при создании процесса мы задаем начальное состояние для обработчика сообщений следующим образом: `spawn(fun() -> message_handler(InitState) end)`.

# Nmap: Сканируем

Внимание Игоря Штомпеля привлекло кроссплатформенное приложение, с помощью которого вы узнаете многое...



Наш  
эксперт

Игорь Штомпель каждый день открывает в Linux что-то новое и использует его где только возможно. А Unity ему в этом ничуть не мешает.

Многих пользователей волнуют вопросы сетевой безопасности. Одним из инструментов ее достижения является программа *Nmap*, новая версия которой вышла недавно. Мы расскажем, как начать с ней работать.

## Знакомство с Nmap

*Nmap* (<http://nmap.org/>) – это свободное (доступно на условиях лицензии GNU GPLv2) кроссплатформенное программное обеспечение, которое позволяет осуществлять исследование и аудит сетевой безопасности. Платформы, на которых может работать *Nmap* – Linux, семейство BSD (Free-, Net-, Open-), Mac OS X, Solaris, Windows и ряд других. Хотя сегодня программа доступна для различных операционных систем, первоначально она создавалась для систем семейства \*NIX.

История создания *Nmap* такова. Ее автор Гордон Лайон [Gordon Lyon] был недоволен возможностями доступных на тот момент (вторая половина 90-х гг.) сетевых сканеров безопасности (*SATAN*, *Strobe* от небезызвестного Джулиана Ассанжа [Julian Assange] и других). Он пытался модифицировать существующие программы, но результат его не устраивал. После этого Лайон решил сам написать программное обеспечение, которое бы обладало всеми необходимыми ему возможностями.

Что же это за возможности, которые были необходимы будущему разработчику *Nmap*? Все довольно просто: ему не требовалось ничего необычного, хотя по тем временам, наверно, это и было необычно: поддержка всех видов сканирования, быстрая работа. Именно этого он не мог получить от существующих к моменту начала разработки *Nmap* сетевых сканеров безопасности.

Итак, поставив указанные требования во главу угла, Лайон разрабатывает свою программу, пользуется ею и сперва тестирует ее самостоятельно, а затем передает *Nmap* в электронный журнал *Phrack* (где 1 сентября 1997 года был опубликован исходный код: <http://www.phrack.org/issues.html?issue=51&id=11#article>). С тех пор *Nmap* продолжает свое развитие, успел получить широкое распространение в соответствующей сфере безопасности

информационных технологий, а для многих стал незаменимым инструментом аудита и сканирования Всемирной паутины.

## Установка Nmap

21 мая 2012 года был представлен новый релиз *Nmap* – шестая версия. В репозиториях Ubuntu 12.04 (именно в ней мы будем осуществлять всю работу со сканером) доступна только старая версия, 5.21, что и понятно: ведь операционная система была выпущена в апреле. Поэтому установим *Nmap 6* в Ubuntu 12.04 из архива с исходным кодом.

Загрузить данный архив можно со специальной страницы официального сайта *Nmap* (<http://nmap.org/download.html>). Версия *Nmap*, доступная на момент написания статьи – 6.01. Мы загрузили соответствующий архив: **nmap-6.01.tar.bz2**. Теперь произведем установку.

Скопируйте архив *Nmap 6.01* в каталог назначения, где вы будете осуществлять установку. Далее извлеките файлы из архива:

```
$ bzip2 -cd nmap-6.01.tar.bz2 | tar xvf -
```

После удачного завершения выполнения команды вы увидите новый каталог – **nmap-6.01**. Перейдем в него:

```
$ cd nmap-6.01
```

Далее произведем конфигурирование стандартным **/configure**. В случае успешного завершения конфигурирования вы увидите примерно то, что показано на рис. 1.

Осталось осуществить сборку программы –

```
$ make
```

и, конечно, ее установить (не забудьте получить права суперпользователя):

```
$ sudo make install
```

В итоге в выводе этой команды вы должны увидеть следующее:

```
NMAP SUCCESSFULLY INSTALLED
```

Вот и все – установка завершилась успешно, и *Nmap* готов к использованию.

## Новшества Nmap 6

Скажем несколько слов о новшествах, которые включила шестая версия *Nmap*. Основные большие изменения затронули шесть областей (<http://nmap.org/6/#6changes>): поддержку NSE-скриптов (специальные скрипты на базе *Nmap Scripting Engine*, которые позволяют автоматизировать стандартные действия при сканировании); улучшение сканирования web-сервисов; полная поддержка IPv6; новая утилита *pring*; усовершенствование *Zenmap*; увеличение скорости сканирования.

Дадим подробности. Так, число NSE-скриптов, по сравнению с *Nmap 5*, увеличено с 59 до 348. Поддержка сканирования web-сервисов позволяет получить информацию о параметрах web-сервера (техники кэширования, поддержка HTTP и т.д.). Кроме того, количество NSE-скриптов для сканирования web-сервисов возросло с 6 до 54. Реализация полной поддержки IPv6 позволяет производить сканирование с помощью raw-пакетов портов систем, использующих IPv6. Новая утилита *nping* ориентирована на анализ пришедших пакетов и времени их прихода, а также поддерживает различные сетевые протоколы, для которых позволяет

» Рис. 1. Конфигурирование *Nmap* завершено.



# Паутину

генерировать соответствующие произвольные пакеты. Графический интерфейс *Zenmap* получил новую функциональность – например, появились локализации интерфейса (в том числе поддержка русского языка) и возможности по фильтрации хостов. В новом релизе *Nmap* также повысилась скорость сканирования. В частности, был переписан код системы трассировки, с целью достижения высокой производительности работы и реализации новых возможностей распараллеливания операций для движка NSE и системы определения версий.

## Сканирование с Nmap

*Nmap* обладает богатым арсеналом возможностей по получению информации об удаленном компьютере, локальной сети и т.д. Целой книги не хватит, чтобы описать все методы работы с программой. Но начинать с чего-то надо, и мы начнем с простого – разберем формат команды **nmap**, который используется для сканирования:

**nmap** [Тип(ы) сканирования] [Опции] [Цель]

Здесь название [Тип(ы) сканирования] говорит само за себя, о типах подробнее далее; [Опции] – специальные опции, которые специфицируют сканирование (не являются обязательными); [Цель] – цель сканирования (например, локальная петля 127.0.0.1).

Прежде чем перейти непосредственно к изучению сканирования с помощью *Nmap*, обратим ваше внимание на важный момент. Все опции, начинающиеся с **-s**, относятся к осуществлению сканирования:

- » **-sP** – пинг-сканирование;
- » **-sV** – получить информацию о версии сервисов;
- » **-sU** – UDP-сканирование;
- » **-sO** – сканирование протоколов;
- » **-sS/-sT/-sA/-sW/-sM** – различные виды TCP SYN-сканирования;
- » **-sN** – TCP Null-сканирование;
- » **-sF** – FIN-сканирование;
- » **-sX** – Xmas-сканирование.

Некоторых из приведенных методов мы опишем поподробнее.

Для начала осуществим пинг-сканирование, применяемое для определения доступности цели сканирования. В отличие от утилиты *ping*, отправляющей так называемые «эхо-запросы»

на основе протокола ICMP, которые нередко блокируются на исследуемых хостах, *Nmap* отправляет по умолчанию ACK-пакеты на порт 80. ACK – это флаг пакета протокола TCP, который призван сообщать о том, что предыдущий пакет был получен. Итак, произведем сканирование:

```
$ sudo nmap -sP scanme.nmap.org
```

Сканируемый узел оказался доступен.

Кстати, просканировать несколько машин можно, например, таким образом:

```
$ sudo nmap -sP 192.168.0.1, 20-25
```

В данном случае будут просканированы 7 IP-адресов (192.168.0.1 и, соответственно, в этом же адресном диапазоне с 20 по 25).

Есть и другие способы:

```
$ sudo nmap -sP 192.168.0.*
```

```
$ sudo nmap -sP 192.168.0.1/24
```

Обратите внимание, что для пинг-тестирования можно воспользоваться дополнительными опциями (более полный перечень доступен в страницах руководств):

- » **-PA** [список портов] – отправка ACK-пакетов;
- » **-PE** – пингование с использованием эхо-запросов на базе протокола ICMP;
- » **-PN** – пропустить обнаружение хостов, считать все хосты работающими;
- » **-PO** [список протоколов] – пингование с использованием протокола IP;
- » **-PR** – ARP-пингование;
- » **-PS** [список портов] – отправка SYN-пакетов (SYN – это флаг TCP-пакета, который информирует о начале TCP-соединения);
- » **-PU** [список портов] – UDP-пингование.

Проведем сканирование без использования опций, что позволит нам получить информацию о запущенных сервисах и открытых портах на цели сканирования. В качестве цели сканирования мы воспользуемся сайтом, который разработчики *Nmap* реализовали и предоставили специально для тренировки в использовании программы – [scanme.nmap.org](http://scanme.nmap.org):

```
$ nmap scanme.nmap.org
```

Кстати, если во время сканирования нажимать какую-нибудь клавишу (за исключением функциональных – F1 и т.д.), то можно

» Рис. 2. Просмотр состояния процесса сканирования.

```
igor@calc:~$ nmap scanme.nmap.org
```

```
Starting Nmap 6.01 ( http://nmap.org ) at 2012-07-06 12:16 MSK
Stats: 0:00:01 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 19.35% done; ETC: 12:16 (0:00:04 remaining)
Stats: 0:00:02 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 28.50% done; ETC: 12:16 (0:00:05 remaining)
Stats: 0:00:02 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 34.15% done; ETC: 12:16 (0:00:04 remaining)
Stats: 0:00:04 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 78.47% done; ETC: 12:16 (0:00:01 remaining)
```

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

увидеть информацию о текущем состоянии процесса сканирования (см. рис. 2).

В выводе мы видим, что на цели сканирования открыты порты 22, 25, 80 и 9929, на которых запущены соответствующие сервисы (*ssh*, *smtp*, *http*, *nping-echo*). Кстати о сервисе, работающем на порту 9929: он предназначен для исследования работы новой утилиты *nping*, о которой мы упоминали выше.

Еще несколько слов о предыдущем выводе *Nmap*. Условно говоря, в нем мы увидели три столбца:

» **PORT** – информация о номере порта и протоколе;

» **STATE** – состояние портов (*open* – открыт, т.е. пакеты на этом порту будут приниматься и обрабатываться соответствующим сервисом;

*closed* – никакой сервис на порту не работает, но порт может быть открыт; *filtered* – порт закрыт фильтром, например, брандмауэром, который, возможно, его блокирует, и поэтому нельзя точно определить, закрыт порт или открыт; *unfiltered* означает, что порты обслуживают пакеты, которые отправляет *Nmap*, но последний не определил, открыты они или закрыты.);

» **SERVICE** – название сервиса.

Давайте попытаемся определить операционную систему. Для этого в *Nmap* доступна опция **-O**:

```
$ nmap -O scanme.nmap.org
```

Но, выполнив эту команду, мы не получили желанного результата – запуск *Nmap* выполнялся без прав суперпользователя, необходимых при использовании методов идентификации операционной системы на основе результатов сканирования с использованием протоколов TCP/IP (“TCP/IP fingerprinting (for OS scan) requires root privileges”).

Повторим попытку:

```
$ sudo nmap -O scanme.nmap.org
```

Теперь сканирование завершилось удачно.

*Nmap* позволяет осуществить сканирование сразу нескольких целей. Для примера, модифицируем команду, которую мы отдавали выше, добавив в качестве дополнительной цели локальную:

```
$ sudo nmap -O scanme.nmap.org 127.0.0.1
```

В выводе этой команды мы увидим, как показано на рис. 3, условно, два отчета, по каждой цели сканирования.

» **Рис. 3.** Сканирование двух целей.

```
Starting Nmap 6.01 ( http://nmap.org ) at 2012-07-06 12:20 MSK
Nmap scan report for scanme.nmap.org (74.207.244.221)
Host is up (0.21s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    filtered smtp
80/tcp    open  http
9929/tcp  open  nping-echo
Device type: general purpose|storage-misc|WAP|media device|webcam
Running (JUST GUESSING): Linux 2.6.X|3.X (94%), HP embedded (93%), Netgear embedded (92%), western
Digital embedded (92%), Linksys Linux 2.4.X|2.6.X (91%), AXIS Linux 2.6.X (91%), Asus Linux 2.6.X
(90%)
OS CPE: cpe:/o:linux:kernel:2.6 cpe:/o:linux:kernel:3 cpe:/o:linksys:linux:2.4 cpe:/o:linux:kernel
:2.6.22 cpe:/o:axis:linux:2.6 cpe:/h:asus:rt-n16 cpe:/o:asus:linux:2.6 cpe:/o:linux:kernel:2.6.18
Aggressive OS guesses: Linux 2.6.39 (94%), HP P2000 G3 NAS device (93%), Linux 2.6.32 - 2.6.39 (92
%), Netgear DG834G WAP or Western Digital WD TV media player (92%), Linux 2.6.22 - 2.6.36 (92%), L
inux 2.6.32 (92%), Linux 2.6.34 (92%), Linux 2.6.37 (92%), Linux 2.6.36 (91%), Linux 3.0 - 3.1 (91
%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 12 hops

Nmap scan report for localhost (127.0.0.1)
Host is up (0.000059s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
631/tcp   open  lpp
Device type: general purpose
Running: Linux 2.6.X|3.X
OS CPE: cpe:/o:linux:kernel:2.6 cpe:/o:linux:kernel:3
OS details: Linux 2.6.32 - 3.2
Network Distance: 0 hops

OS detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 2 IP addresses (2 hosts up) scanned in 12.28 seconds
```

На удаленных хостах имеется ряд нюансов определения операционных систем, и для этого предусмотрены дополнительные опции. Скажем, с опцией **--osscan-limit** –

```
$ sudo nmap -O --osscan-limit scanme.nmap.org
```

– *Nmap* не будет определять операционные системы тех целей сканирования, которые не имеют как минимум одного открытого и одного закрытого порта. По утверждению разработчика программы, определение операционной системы будет более эффективно именно в таком случае. Пропуск при сканировании не удовлетворяющих этому условию целей позволяет оптимизировать процесс сканирования. Например, при сканировании большого количества целей таким образом можно сократить время сканирования без ущерба для его качества.

*Nmap* может «угадывать» результаты сканирования, задействовав опции **--osscan-guess** или **--fuzzy**:

```
$ sudo nmap -O --osscan-guess scanme.nmap.org
```

Такое сканирование эффективно в ситуациях, когда *Nmap* не может точно определить операционную систему на цели сканирования. В результате программа выводит процент соответствия «угаданной» операционной системы реальной.

В ситуации, когда необходимо точное соответствие, можно задействовать опцию **--max-os-tries**, которая позволяет задать максимальное количество попыток определения операционной системы на цели сканирования:

```
$ sudo nmap -O --max-os-tries 10 scanme.nmap.org
```

Дело в том, что по умолчанию *Nmap* в «благоприятных» условиях для сканирования совершает две попытки, а в «неблагоприятных» – пять. При малом количестве попыток *Nmap* может пропускать некоторые результаты сканирования, которые при последующих попытках помогли бы в определении операционной системы.

Общую информацию о цели сканирования мы научились получать в виде запущенных сервисов и открытых портов. Теперь исследуем, какие версии сервисов запущенных на интересующей на цели. Для этого воспользуемся опциями **-sV**.

Итак, получим информацию о версиях сервисов на цели сканирования:

```
$ sudo nmap -sV scanme.nmap.org
```

Как видно из рис. 4, на цели сканирования запущены сервисы *OpenSSH 5.3p1*, *Apache 2.2.14* и, по всей вероятности, стоит операционная система *Ubuntu*. По крайней мере, точно известно, что на цели сканирования работает операционная система *Linux*: см. в выводе **Service Info: OS**.

Перейдем к следующему виду сканирования. Обратим внимание читателей на то, что ряд важных сервисов может работать по протоколу UDP (например, *DNS*, *RPC* и другие). Поскольку UDP-сканирование *scanme.nmap.org* у нас прошло неудачно (программа продолжала работать, а вывода, соответственно, не было, причем продолжительное время), мы решили просканировать локальный адрес:

```
$ sudo nmap -sU 127.0.0.1
```

Выполнение команды завершилось успешно, и мы увидели вывод с информацией о портах и соответствующих сервисах.

Еще один метод получил название сканирования протоколов. Дело в том, что протоколы транспортного уровня стека TCP/IP имеют номера. Так, в поле с названием *protocol* пакета IP содержится информация о заголовке пакета и номере протокола, соответственно, транспортного уровня. *Nmap* отправляет

» **Пропустили номер?** Узнайте на с. 104, как получить его прямо сейчас.

```
igor@calc:~$ sudo nmap -sV scanme.nmap.org
Starting Nmap 6.01 ( http://nmap.org ) at 2012-07-06 13:24 MSK
Nmap scan report for scanme.nmap.org (74.207.244.221)
Host is up (0.24s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 5.3p1 Debian 3ubuntu7 (protocol 2.0)
25/tcp    filtered smtp
80/tcp    open  http         Apache httpd 2.2.14 ((Ubuntu))
9929/tcp  open  nping-echo   Nping echo
Service Info: OS: Linux; CPE: cpe:/o:linux:kernel

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.55 seconds
igor@calc:~$
```

Рис. 4. Nmap определил версии работающих сервисов на цели сканирования.

на цель сканирования пакеты с номерами протоколов, и если соответствующие ответы получены, то протокол на цели сканирования доступен. Воспользуемся опцией **-sO** для осуществления данного типа сканирования:

```
$ sudo nmap -sO 127.0.0.1
```

В выводе мы увидим, что ряд протоколов доступен: это ICMP (номер 1), TCP (6), UDP (17).

Рассмотрим некоторые возможности, которые могут расширить и детализировать процесс сканирования тех или иных целей. Nmap позволяет специфицировать порты, которые необходимо просканировать. Для этого существует опция **-p**:

```
$ sudo nmap -p 1-100 scanme.nmap.org
```

Здесь мы сканируем только те порты на удаленном хосте, которые нас интересуют. Обратите внимание, что начальное или конечное значение диапазона портов может быть опущено (весь диапазон возможных значений – от 1 до 65535). Приведем пример без начального значения:

```
$ sudo nmap -p -100 scanme.nmap.org
```

В данном случае будут сканироваться порты с 1 по 100. А вот пример с пропущенным конечным значением диапазона портов:

```
$ sudo nmap -p 1- scanme.nmap.org
```

Просканированы будут порты с 1 по 65535, т.е. полный диапазон портов. Полный диапазон портов можно также просканировать, указав следующее:

```
$ sudo nmap -p- scanme.nmap.org
```

Диапазон портов можно указать также с помощью квадратных скобок:

```
$ sudo nmap -p [-23] scanme.nmap.org
```

Будут просканированы порты с номерами от 23 и менее.

А как быть в ситуации, когда необходимо просканировать отдельный порт? Позволяет ли это делать Nmap? Конечно! Помимо указания диапазона портов, в Nmap можно специфицировать отдельные порты. Например, порт 80:

```
$ sudo nmap -p 80 scanme.nmap.org
```

При сканировании TCP/UDP-портов Nmap позволяет специфицировать протокол. Для TCP необходимо указывать перед номером порта **T**, а для UDP – **U**. Приведем пример:

```
$ sudo nmap -p T:21,80,U:137 scanme.nmap.org
```

Мы специфицировали как TCP порты 21 и 80, а 137 – как UDP. Допустимо указывать и диапазон портов:

```
$ sudo nmap -p T:21-23,U:137
```

В этом случае порты с 21 по 23 будут сканироваться в качестве TCP-портов.

И еще, обратите внимание на то, что если не указан определитель протокола (**T**: или **U**:), то порт будет сканироваться для всех протоколов.

Вместо числовых значений порты могут быть специфицированы и по именам, которыми они обозначены в файле **nmap-**

**services**. Это файл находится в каталоге **/usr/share/nmap**. Просмотреть его содержимое вы можете, например, так:

```
$ less /usr/share/nmap/nmap-services
```

Рассмотрим примеры сканирования с указанием имен портов вместо указания их номеров. Воспользуемся двумя последними приведенными выше примерами. Для первого у нас получится следующее:

```
$ sudo nmap -p T:ftp,http,U:netbios-ns
```

а для второго –

```
$ sudo nmap -p T:ftp,ssh,telnet,U:netbios-ns
```

Как видим, к сожалению, во втором случае, как и вообще при использовании имен вместо номеров портов, указывать диапазон портов не получается.

Кроме того, вы можете использовать символы **?** и **\*** при спецификации имен протоколов. Традиционно, символ **?** означает один пропущенный символ, а **\*** – множество. Приведем пример:

```
$ sudo nmap -p T:htt? scanme.nmap.org
```

В выводе мы увидим порты, полное имя которых – http. Для нашего случая со [scanme.nmap.org](http://scanme.nmap.org) это порты 80 и 8080.

Рассмотрим другой пример:

```
$ sudo nmap -p T:htt* scanme.nmap.org
```

В этом случае вывод будет гораздо более обширным. В нем мы увидим следующие порты: 80 – http, 280 – http-mgmt, 443 – https, 591 – http-alt, 593 – http-rpc-epmap, 8000 – http-alt, 8008 – http, 8080 – http-proxu, 8443 – https-alt. Итак, как видим, Nmap вывел порты, имена которых начинаются с **htt**.

На этом мы завершим рассмотрение работы с Nmap. Во второй части урока мы поговорим о работе с графическими оболочками для Nmap (например, Zenmap, Umit), утилите hping3.

## Заключение

Nmap – мощное средство исследования Сети, доступное для различных платформ, а что касается Linux, то во многих дистрибутивах из репозитория. Правда, с новой версией Nmap репозитории иногда отстают. Однако установка программы не составляет проблем и осуществляется достаточно традиционно.

В целом, Nmap предоставляет богатые возможности по сканированию Сети, а само сканирование осуществляется быстро. Применяя те или иные опции, можно оптимизировать процесс сканирования – в частности, сократить его время и получить только интересующую информацию.

Настоятельно рекомендуем использовать Nmap исключительно в исследовательских или рабочих целях, согласовав использование программы со всеми заинтересованными лицами, а также исключительно в рамках действующего законодательства. Следует учитывать и то, что, например, многие провайдеры в договорах со своими абонентами запрещают производить сканирование Сети. **LXF**

# ОТВЕТЫ

Есть вопрос по открытому ПО? Пишите нам по адресу [answers@linuxformat.ru](mailto:answers@linuxformat.ru), и мы найдем ответ.

В этом месяце мы ответим на вопросы про...

- 1 Автоматизацию серфинга
- 2 Утилиту для проверки орфографии Enchant
- 3 Резервное копирование с rsync
- 4 Загрузчик Grub
- 5 Групповое переименование файлов JPEG
- 6 Операционные системы

## 1 Автосерфинг

В На работе мне, как и многим другим, приходится классифицировать, сравнивать, извлекать и проверять длинные файлы с артикулами, количеством, ценами и другими атрибутами. Мне уже удалось упростить и ускорить некоторые из этих действий путем установки *Cygnin* и запуском всяких скриптов на *Python* и *Bash*. Но я по-прежнему трачу массу времени на получение данных с обычно медленных серверов баз данных через web-интерфейс. Пока я не придумал, как автоматически подключиться к HTTP-странице, аутентифицироваться, щелкнуть по нужным ссылкам, заполнить текстовые поля и, наконец, вернуть данные, полученные в ответ от сервера.

Большинство web-интерфейсов основного сервера работают на *Java 1.6*, и я не могу напрямую изменить код страницы. Нет ли открытых программ, способных как-то записать мои переходы по ссылкам и преобразовать их в редактируемый макрос, просто вызываемый из скрипта *Bash* или *Python*, чтобы мне не приходилось вручную открывать браузер и выполнять тысячи повторных копирований и вставок?

Лоран [Laurent]

Вы не сказали, пытались ли подключиться к базе данных напрямую, хотя многие базы данных, доступные через web-интерфейс, блокируют любые попытки доступа, исходящие не от web-сервера. Если API для доступа к базе данных из скрипта нет, можно применить программу для записи макроса, хотя веб-дизайнеры имеют склонность перемещать все с места на место во имя прогресса (или для имитации кипучей деятельности?).

Вам должен подойти модуль расширения *iMacros* для *Chrome* и *Firefox* (и для *IE*, если другого выхода нет). Версия для *Chrome* пока на стадии бета, поэтому лучше воспользоваться этим модулем в *Firefox*. После установки дополнения зайдите на web-страницу, нажмите на кнопку *iMacros* на панели инструментов, чтобы открыть его панель, а затем на кнопку *Rec* [Запись].



➤ Автоматизируйте рутинные задачи web-браузера с помощью расширений *iMacros* для *Firefox* и *Chrome*.

Выполните необходимые действия и нажмите *Stop* [Остановка]. Теперь очистите вкладку или перейдите на другой сайт и нажмите *Play* [Воспроизвести] в панели *iMacros* – и Вы должны увидеть повтор своих действий.

Модуль не только записывает макросы – после записи их можно редактировать и встраивать в скрипты, чтобы обработать полученную с их помощью информацию далее в скрипте. Детали зависят от того, что именно нужно сделать, но на сайте *iMacros* [www.iopus.com/imacros/firefox](http://www.iopus.com/imacros/firefox) на вики и форумах можно накопить массу примеров и справку.

Запуск из скрипта означает, что Вам даже не придется запускать браузер, поэтому можно поместить все задачи в скрипт, который запускает макрос с помощью *cron* и отправляет результаты Вам по почте или передает их другой программе. Ваши возможности, насколько можно судить, ограничены только воображением и навыками в написании скриптов.

## 2 Лух-чаровник

В У меня проблема с проверкой орфографии в текстовом процессоре *Лух*. Я пользуюсь им в *Ubuntu 11.10*, но проверка орфографии

в редакторе недоступна. Я попытался активировать ее через меню *Tools > Preferences* [Инструменты > Настройки]. Появляется только слово *ENCHANT* [Очарованный], а проверка не работает, и результат не сохраняется. Я поискал решение на форумах, но только и узнал, что у еще нескольких человек та же проблема, а решения никто так и не нашел.

Все пакеты я загружал из репозитория *Ubuntu* через центр управления ПО. В ранних релизах проверка орфографии в *Лух* работала. Я решил установить *Лух* на ноутбук с *Ubuntu 11.04*, на котором я копался в форумах. Все заработало! Единственная разница в том, что в *11.04* загружается версия *Лух 1.6.7*, а в *11.10* – *2.0.0*.

Брайан Митчелл [Bryan Mitchell]

Различие между версиями важно, так как в *2.0.0* для проверки орфографии используется *Enchant*. Это не утилита проверки орфографии, а клиент к другим утилитам, таким как *Hunspell*, *Aspell* и *Ispell*. Идея состоит в том, что вместо выбора одной утилиты проверки орфографии – а в разных программах они разные, что заставляет Вас поддерживать несколько словарей или добавлять в код возможность выбора

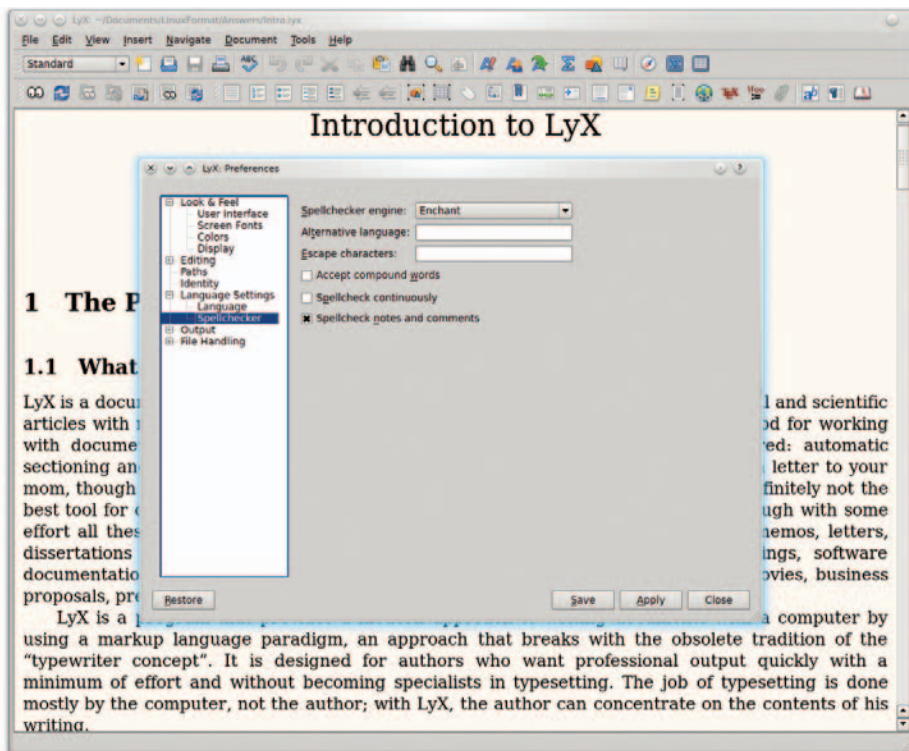
утилиты – все программы поддерживали бы только *Enchant*, который сам вызовет нужную утилиту проверки орфографии. Это позволяет разработчику сэкономить массу времени, которое можно потратить на реализацию новых чудесных функций, а Вам – обойтись одной утилитой проверки орфографии и одним словарем. Проблема, с которой Вы столкнулись, говорит о том, что *Enchant* либо не установлен, и это легко исправить, либо установлен, но не настроен. Последнее вряд ли имеет место, так как по умолчанию у *Enchant* вполне нормальная конфигурация; но настроить его легко. Создайте файл `.enchant/enchant.ordering` в своем домашнем каталоге и добавьте туда всего одну строку:

```
*:aspell,hunspell
```

Фрагмент перед двоеточием определяет языки, к которым применяется правило (в данном случае – ко всем); остальное – список утилит для проверки орфографии, которые будут запущены. Так, в моей системе для проверки орфографии по умолчанию используется *Aspell*, но если его не удастся найти, будет вызван *Hunspell*. Кстати, чтобы это заработало в Ubuntu 11.10, мне пришлось дважды нажать кнопку Save [Сохранить].

### 3 Чувство синхронизма

В Я пытался сделать резервную копию данных с *rsync* – безуспешно. В моей системе используется двойная загрузка – Windows XP и Linux Mint 9 Isadora. Я пользуюсь XP для двух вещей: для моего сканера (Visioteer 9020), который не работает в Linux, и для еженедельного инкрементального копирования с помощью *syncback* с *sdb1* (диск объемом 500 Гб) на *sda5* каталогов *data\_in* и *data-bu* соответственно. В течение недели



Теперь *Lyx* использует интерфейс *Enchant*, а тот, в свою очередь, вызывает нужную утилиту проверки орфографии.

каждый раз по окончании работы я делаю резервную копию домашнего каталога в *data\_in* с помощью *Back In Time* и удаляю старую резервную копию, чтобы не засорять каталог *data\_in* прежними ненужными копиями.

Я хотел бы делать это из Linux, но получаю сообщение о том, что *sdb1* и *sda5* не являются

каталогами. Так как оба раздела имеют файловую систему NTFS, я решил проверить их идентификаторы (UUID), и они оказались одинаковы! А я-то думал, что они должны быть уникальны!

Джон Йейтс [John Yates]

Кажется, что идентификаторы файловых систем в Linux должны быть одинаковы, но на самом деле это не обязательно, потому что вы можете задать UUID файловой системы чему угодно. Однако у файловых систем NTFS и FAT нет UUID, а есть номер тома, присвоенный при создании файловой системы и не обладающий такой степенью «уникальности», как UUID. Более важно, что при клонировании файловой системы номер тома тоже копируется. Изменить номер тома NTFS непросто (в отличие от UUID ext2/3/4). Варианты – отредактировать двоичные данные в заголовке файловой системы напрямую (довольно рискованно) или переформатировать файловую систему. Последний вариант – самый простой, так как номер тома среди прочего формируется на основании даты создания.

Однако причина ошибок с *rsync* не в этом, а в Вашем неправильном понимании работы *rsync*. *rsync* работает на уровне файлов и каталогов, а Вы пытаетесь применять его к блочным устройствам. Вы пытаетесь воспользоваться командой:

```
rsync -av /dev/sdb1/ /dev/sda5/
```

тогда как верный синтаксис (с корректными путями к каталогам) таков:

```
rsync -av /path/to/source/ /path/to/backup/
```

Параметр `-v` выводит подробную информацию, но иногда ее слишком много, и это замедляет процесс. С *rsync* Вам не нужно пользоваться другой



## Коротко про...

# MD5

Если вы когда-нибудь загружали ISO-образы дистрибутивов, вместе с файлами ISO вы видели файлы с именами вроде `cooldistro1.0.iso.md5`. Сумма MD5 – 128-битная криптографическая свертка содержимого файла. Свертка у каждого файла всегда получается одинаковая, но стоит изменить в файле хотя бы один бит – и она изменится радикально. MD5 была предназначена для целей безопасности; проверив сумму MD5, можно убедиться, что файл не был изменен. Теперь она также используется для проверки случайных повреждений файлов при передаче. При загрузке ISO-образа вы получаете и файл MD5. Удостовериться, что файл не поврежден, можно посредством следующей команды:

```
md5sum cooldistro-1.0.iso
```

В результате должна получиться та же строка шестнадцатеричных цифр, что указана в файле `cooldistro1.0.iso.md5`. Сравнить 128-битные

числа друг с другом – не слишком веселое занятие; пусть за нас это сделает программа:

```
md5sum --check cooldistro-1.0.iso.md5
```

Эта команда даст вам знать, соответствует ли контрольная сумма содержимому файла. В файле MD5 можно хранить суммы для нескольких файлов. Например, исходники Gnome поставляются во множестве файлов, но их суммы MD5 находятся в одном файле, часто называемом `MD5SUMS`. Можно загрузить все файлы и проверить их одной командой:

```
md5sum -c MD5SUMS
```

Сгенерировать файл `MD5SUMS` можно следующей командой:

```
md5sum *.iso >MD5SUMS
```

Свертки SHA1 и SHA256 работают по похожей схеме, но обеспечивают немного более высокий уровень защиты. Принцип их работы, как и синтаксис, такие же, лишь используемые программы другие: `sha1sum` и `sha256sum`.

программой для резервного копирования или удалять старые резервные копии. Добавив параметр `--delete`, можно удалить файлы, которых уже нет в исходном каталоге, и мы получим резервную копию, являющуюся точной копией источника. Команда

```
rsync --archive --delete ~/ /mnt/backup/myhome/
```

создаст точную копию домашнего каталога в `/mnt/backup/myhome`. Ее можно запускать как задание `cron`, и тогда о ней даже не придется помнить. Простейший способ запускать ее каждый день – поместить команду в короткий скрипт:

```
#!/bin/sh
rsync --archive --delete /home/myuser/ /mnt/backup/myhome/
```

Поместите скрипт в `/etc/cron.daily`, сделайте файл исполняемым, и резервное копирование будет выполняться каждый день в одно и то же время. Путь до домашнего каталога нужно указывать полностью, так как скрипты из `cron.daily` запускаются от имени пользователя `root`.

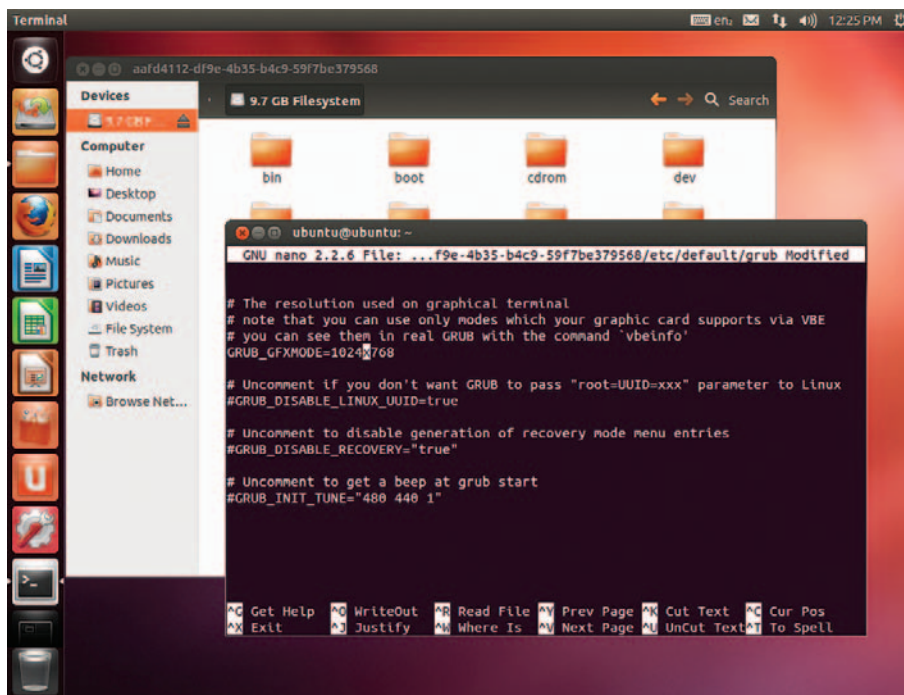
Недостаток этого подхода в том, что теряются все предыдущие копии. Если вы что-то испортите в файле, то испорченный вариант файла попадет в резервную копию. Одно из решений проблемы – воспользоваться `rdiff-backup` ([www.nongnu.org/rdiff-backup](http://www.nongnu.org/rdiff-backup)), который использует `rsync`, но сохраняет определенное (настраиваемое) количество прежних копий каждого файла, чтобы его можно было восстановить. `Rdiff-backup` также работает с командной строки – и его очень удобно запускать с `cron`.

## 4 Один диск — один дистрибутив?

В Я перешел с Windows на Mint, и впечатления самые приятные. Раньше мне попадали старые компьютеры, прошедшие через вторые или третьи руки, и вот в возрасте 73 лет я решил купить новый. На компьютере, который я хочу купить, два диска по 500 Гб. На один я поставлю Mint, а что посоветуете для второго?

Терри Бэйл (Terry Bale)

О На одном диске не обязательно размещать только одну операционную систему. На самом деле, так бывает редко. Привязывать одну операционную систему к одному диску – трата места, так как большинству операционных систем нужно для своих файлов меньше 15 Гб. Остальное обычно отдается под файлы пользователей. Так как Вы скорее всего захотите слушать одну и ту же музыку в обеих системах, большую часть места стоит выделить под домашние каталоги, а операционным системам оставить всего несколько десятков Гб. Сначала установите Mint на первый диск, создав во время установки отдельный раздел для домашнего каталога. Настройте все необходимое, пока не приступая ко второму диску. Если Вам понадобится больше места, чем 500 Гб, можно добавить разделы на второй диск и смонтировать их в своей файловой системе позже. Например, если у Вас большая коллекция видео, можно создать для нее отдельный раздел на втором диске и смонтировать



► Если вам или вашему монитору не нравится разрешение *Grub* по умолчанию, его легко изменить.

в каталог `videos` в своем домашнем каталоге. Для этого создайте раздел и файловую систему стандартной утилитой для работы с разделами, например, `Gparted`, и добавьте следующую строку в файл `/etc/fstab`.

```
/dev/sdb5 /home/terry/videos ext4 defaults 0 0
```

После перезагрузки новый раздел будет смонтирован в каталог `videos` в домашнем каталоге. Чтобы записывать в него данные под своим пользователем, нужно выполнить следующую команду после первого монтирования (или первой перезагрузки):

```
sudo chown -R terry: ~terry/videos
```

Обязательно сохраните свободное место, которое Вам пока не нужно, для экспериментов с другими дистрибутивами, но сначала настройте главную систему. Когда захотите поэкспериментировать, начните с дистрибутивов, которые не основаны на Ubuntu, так как у них слишком велико сходство с Mint. Такие дистрибутивы, как Fedora, OpenSUSE, PCLinuxOS и Mageia, могут оказаться хорошей отправной точкой.

При установке другого дистрибутива откажитесь от опции по умолчанию перезаписать загрузчик главной загрузочной записи первого диска. Пусть новый дистрибутив разместит загрузчик на собственном разделе. При этом загрузчик Mint останется нетронутым, так что новый дистрибутив сначала будет недоступен. Вернитесь в Mint и выполните команду:

```
sudo update-grub
```

в терминале. Она проверит жесткие диски на наличие операционных систем и добавит их в загрузчик Mint. После этого в одном меню будут все операционные системы. Каждый раз при добавлении, удалении или обновлении другого дистрибутива, эту команду нужно будет запустить в Mint один раз.

## 5 Проблема с монитором

В У меня возникла проблема при установке Ubuntu в систему с двойной загрузкой — XP и Ubuntu. Та же проблема иногда возникала в прошлом — разрешение загрузчика было маленьким, появлялось сообщение о том, что частота и разрешение выходят за диапазон работы монитора, и изображения не было. Раньше я решал эту проблему так: загружался в Ubuntu и запускал `startup-manager`, в котором увеличивал разрешение, и после этого система загружалась с нормальным разрешением. После установки 12.04 `startup-manager` уже не установить, и, следовательно, нельзя изменить настройки менеджера загрузки. Есть ли простой способ решить эту проблему во время установки системы?

Рэй Хьюз (Ray Hughes)

О В *Grub* по умолчанию используется разрешение 640×480, так как его поддерживают почти все мониторы, а для нескольких строк текста большего и не надо. В *Grub 2*, который сейчас используется в большинстве дистрибутивов, разрешение легко изменить. От имени суперпользователя-`root` откройте файл `/etc/default/grub`, и Вы увидите строку:

```
#GRUB_GFXMODE=640x480
```

Символ `#` в начале строки – комментарий, с ним строка становится недействующей. Удалите символ `#` и измените разрешение на более подходящее для своего монитора, затем сохраните файл. Если Вы не уверены, что редактируете файл от имени суперпользователя, откройте терминал и выполните команду:

```
sudo nano /etc/default/grub
```

Она откроет файл в простом редакторе в терминале от имени `root`. После внесения изменений нажмите `Ctrl+X` для сохранения файла и выхода.



Помогите нам помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, так как 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-файл, который вы сможете приложить к своему письму. Альтернативный и не менее удобный вариант – *lshw* (<http://ezix.org/project/wiki/HardwareLiSter>). Одна из этих программ должна быть включена в ваш дистрибутив (а то 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
```

ливать и изучать отдельные программы для решения однократной задачи. **LXF**

Теперь обновите файл меню *Grub* командой `sudo update-grub`

и разрешение при следующей загрузке системы должно измениться на заданное вами. При установке системы со стандартных CD Ubuntu изменять его нельзя – Вам понадобятся альтернативные CD.

Однако можно загрузиться со стандартного CD после установки и открыть файловый менеджер для монтирования корневой файловой системы. Файл нельзя изменить из нее, но после монтирования диска файл можно будет открыть с терминала в *Nano* как указано выше.

Диск будет смонтирован в каталог `/media/xxxxxxx`, где `xxxxxx` – UUID корневой файловой системы, который будет показан в заголовке окна файлового менеджера.

### 6 Групповуха с именами

**В**Моя жена привыкла к дурацкому способу опускать расширения в Windows, и переименовала все свои фотографии без расширения JPG. Их сотни. Пока ее ноутбук был в ремонте, она пользовалась ноутбуком дочери с Mint 11 и думает, что это Linux во всем виноват. Когда я говорю, что это она переименовала файлы, она обижается. Нет ли простого способа добавить к каждому расширению .jpg, не переименовывая каждый файл вручную?

Heiowge, с форумов

Как водится, в Linux есть несколько способов это сделать, в зависимости от имеющихся у Вас программ. В KDE можно воспользоваться простой графической программой *KRename*; пользователи Gnome мо-

гут установить *gprename*; а у пользователей *Xfce* данная возможность есть в файловом менеджере *Thunar*. Но это легко сделать и с командной строки. Какой бы способ Вы ни выбрали, сделайте резервную копию каталога – ошибка при переименовании может привести к перезаписи всех файлов.

Команда **find** входит в базовый набор программ всех дистрибутивов. Откройте в терминал, зайдите в каталог с фотографиями командой **cd** и выполните команду:

```
find -type f ! -iname \*.jpg -exec mv "{}" "{}.jpg" \;
```

Она ищет в текущем каталоге (чтобы искать в другом каталоге, укажите его сразу после **find**) все объекты типа `f` (обыкновенные файлы), имена которых не заканчиваются на `.jpg` или `.JPG`. Директива `-iname` проверяет соответствие имени с учетом регистра, а `!` в начале инвертирует условие, и мы получаем только файлы, которые не соответствуют условию. Часть `-exec` велит команде **find** выполнить указанную команду, заменив `{}` на имя файла, и эта команда переименует все файлы, у которых нет расширения `.jpg`. Кавычки нужны для обработки имен файлов с пробелами. Символы `*` и `;` экранируются обратным слэшем `\`, чтобы оболочка не пыталась интерпретировать их перед передачей команде **find**.

Существуют и другие команды для переименования файлов. У Вас есть команда **rename**, являющаяся частью пакета **coreutils**, который есть в каждом дистрибутиве, но в данном случае пользоваться ею неудобно.

Можно также воспользоваться альтернативой команде **rename** с <http://rename.berlios.de>, или **mmv** (<http://packages.debian.org/unstable/utils/mmv>), но команда **find** у Вас уже есть, и она делает именно то, что необходимо – Вам нет нужды установ-



## Часто задаваемые вопросы

# Wine

### » Что такое Wine?

Это аббревиатура от «Wine Is Not an Emulator [Wine – это не эмулятор]».

### » Если это не эмулятор, тогда что?

Ну, в какой-то степени все же эмулятор. Wine – открытая реализация API Windows поверх X и Unix.

### » С Wine можно запустить Windows в Linux?

Не совсем. Wine – слой совместимости между программами Windows и Linux. Когда вы запускаете программу Windows в Wine, программа думает, что работает в Windows, но вызовы функций Windows, которые она выполняет, преобразуются в вызовы функций Linux.

### » Нужна ли установленная Windows, чтобы Wine заработал?

Нет, но если Windows установлена, Wine воспользуется ее DLL-библиотеками. Если нет, у нее есть альтернативные варианты.

### » Если у меня есть Windows, зачем мне Wine?

Прежде всего это означает, что можно запустить программу Windows, не перезагружаясь в Windows и обратно в Linux. Можно запускать программы Windows вместе с программами в Linux.

С Wine также можно пользоваться модулями расширения Windows для программ Linux, такими как web-браузеры и медиапроигрыватели. Это дает программам Linux доступ к форматам, поддерживаемым только проприетарными кодеками Windows.

### » Бывает, что эмуляторы тормозят – а насколько быстр Wine?

Wine эмулирует только API Windows, а не устройства. Это означает, что программы в Wine будут работать с той же скоростью, что и в Windows, иногда даже быстрее. Недостаток в том, что Wine может запускаться только на том же оборудовании, что и Windows, поскольку это не эмулятор процессора.

### » Как ее загрузить?

Загрузить Wine во множестве форматов можно с сайта проекта. Пакеты также доступны в репозиториях большинства дистрибутивов. Дополнительно имеются две коммерческих версии Wine.

*CrossOver Office* от Codeweavers предназначена для запуска рабочих приложений Windows. Она также запускает модули расширения для

*Internet Explorer* в некоторых браузерах, применяющихся в Linux. Вся сложная работа по конфигурации выполнена за вас.

На сайте Codeweavers перечислено 50 поддерживаемых приложений, но на самом деле их гораздо больше.

### » С рабочими приложениями ясно, а что у нас с играми?

Существует еще одна коммерческая версия – *Cedega*, ранее известная как *WineX*. *Cedega* от TransGaming Technologies – улучшенная версия Wine с поддержкой DirectX. *Cedega* доступна бесплатно через репозиторий CVS.

Если вам нужны готовые пакеты и поддержка, оформите подписку. Список поддерживаемых программ весьма обширен, и он постоянно пополняется.



# LXF HotPicks



**Ник Вейч**

Ника однажды укусил радиоактивный менеджер скачиваний, и с тех пор у него болезненная страсть к экспериментам с установкой программ. И дурное пищеварение.

DigiKam » PacketFence » Tor » timeline » Gfeedline » cputlimit » Xtide » Dccnightmare » Billiards » Rapid Photo Downloader

## Инструмент обработки фото

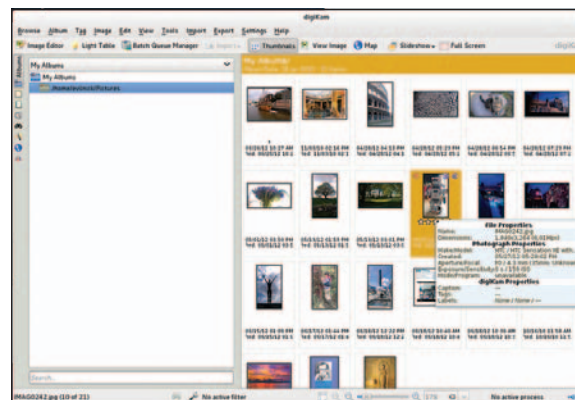
# DigiKam

Версия 2.6.0 Сайт [www.digikam.org](http://www.digikam.org)

**Д**ля работы с фото есть просто инструменты, а есть лучший инструмент. *DigiKam* некогда был самой популярной возможностью для желающих загрузить свои фотографии на машину с Linux, но ситуация изменилась, когда разные дистрибутивы решили, что новым пользователем он не по зубам. В какой-то мере это правда — он умеет намного

больше, чем обычно ожидают от редактора фотографий. Но мне кажется, что подобное суждение сослужило ему плохую службу, поскольку подразумевает некую сложность в работе с ним.

Что ж, давайте кое-что проясним. Конечно, нечто с четырьмя кнопками всегда сложнее, чем нечто однокнопочное, но, я думаю, справиться с этим все же можно. Пользователи Linux чуть более компетентны и смекалисты, чем среднестатистический человек за клавиатурой, и кажется стран-



► **Может быть, это и не самый популярный выбор для создателей дистрибутивов, но DigiKam есть за что рекомендовать.**

ным, что производители дистрибутивов сочли необходимым так их опекать. *Shotwell* (нынешнее приложение по умолчанию в ряде дистрибутивов) — хорошая программа, но я не могу сказать, что пользовался ею больше двух раз: она просто не может сделать со снимком то, что мне надо. *DigiKam* — во многом ее антитеза. С годами она обзавелась поддержкой и функциями для самых разных целей, до такой степени, когда становится возможным сделать практически все, что обычно может понадобиться сделать с полупрофессиональными фотографиями, не прибегая к другому приложению. По большей части это все идет в виде модулей расширения KIPi, которые также способствуют превращению *DigiKam* в расширяемый и удобный для индивидуальной настройки инструмент.

При наличии у него поддержки практически всех типов файлов изображений, быстрого предпросмотра, пакетных операций, отзывчивого и простого в использовании редактора и интерфейса с высокой возможностью настройки, трудно представить, что *DigiKam* не впишется в вашу схему обработки фото. Большим изменением в этом релизе является новый Progress Manager — это намного лучший способ контроля и управления пакетных очередей: вместо того, чтобы обрабатывать одно изображение за раз, вы можете выстроить их в очередь (в том числе и для любых операций KIPi) и спокойно отправиться на обед, пока программа продвигается сквозь 10 000 снимков одного и того же объекта с разных углов...

«Трудно представить, что DigiKam не впишется в вашу схему.»

## Исследуем интерфейс DigiKam

### Browse [Просмотр]

Меню Browse позволяет просматривать недавно добавленные изображения или производить более сложный поиск по тэгам.

### Batch Queue [Пакетная очередь]

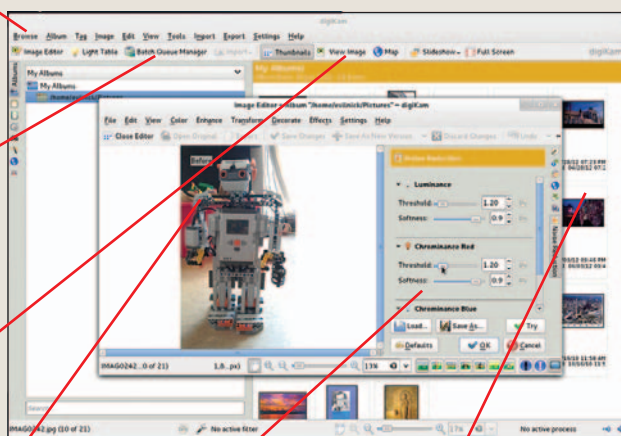
Batch Queue Manager проследит за утомительным процессом трансформации, который вы затеяли.

### View Image [Просмотр изображений]

Просматривайте свои фотографии в самых разнообразных режимах, в зависимости от потребностей.

### Metadata [Метаданные]

Если навести курсор на изображение, будут отображаться метаданные, включая также и тэги *DigiKam*.



### Editor [Редактор]

Открывайте свои изображения в редакторе, чтобы произвести с ними все положенные действия.

### KIPi plugins

Модуль расширения KIPi делают возможными операции по трансформации и использованию ряда фильтров.

## Безопасность

# PacketFence

Версия 3.4.1 Сайт [www.packetfence.org](http://www.packetfence.org)

**Б**езопасность важна! Особенно когда имеешь дело с сетью публичного доступа. Мы имеем в виду такие штуки, как интернет-кафе, университеты, отели, банки, конференц-центры и даже вашу собственную сеть, если у вас роутер Wi-Fi, открытый для всех.

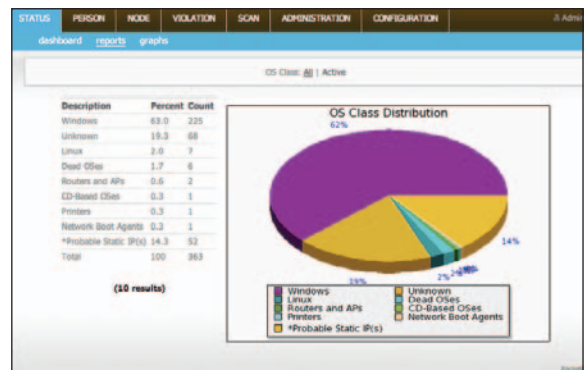
В подобных ситуациях стоит призвать нечто вроде *PacketFence*. Что он может? Список функций включает адаптивный портал для регистрации и решения проблем безопасности, централизованное проводное и беспроводное управление, гостевые опции, поддержку 802.1X и изоляцию уровня-2 [layer-2 isolation] проблемных устройств... если бы вы знали, что они могут в количестве больше одного, вы, вероятно, смогли бы найти им применение.

Например, с адаптивным порталом вы, вероятно, сталкивались, пытаясь где-то на ходу получить доступ к Wi-Fi. Именно здесь запрос HTTP действующего сайта перенаправляется на страницу регист-

рации (или, на самом деле, куда хотите), если отправляющему запрос компьютеру не удастся пройти регистрационную проверку. Очень удобно в тех случаях, когда вам нужно, чтобы гости регистрировались для получения доступа к Сети.

*PacketFence* может работать внутри сети (быть включенным «в разрыв» между беспроводными устройствами и сетью) или Out of Band (сокращенно – OOB, это когда программа работает на сервере, но физически не выступает в роли «бутылочного горла»), и при необходимости применяет небезызвестную программу *Snort* как агента обнаружения проблем. Имеется масса онлайн-документации, помогающей понять, как настраивать программу,

«Базовая установка должна пройти быстрее варки кофе.»



► Помимо оберегания вашей сети, *PacketFence* собирает кое-какие ценные данные.

и базовая установка должна пройти быстрее варки кофе. *PacketFence* предлагает полный набор рабочих и надежных функций, которые, как правило, бывают только в заказном ПО с закрытым кодом от коммерческих компаний. Приятно знать, что не только Linux довольно трудно поддается взлому, но и вы можете усилить свою безопасность, и при этом сохранить открытость кода! *PacketFence* легко компилируется из исходника, приложенного на DVD, но стоит заглянуть на сайт и познакомиться с другими опциями.

## Инструмент конфиденциальности

# Tor

Версия 0.2.2.37 Сайт <http://bit.ly/L3EXIB>

**T**or изначально задумывался как проект для защиты военных переговоров. О нем даже сложилось особое мнение в ФБР! Ну, то есть, я хочу сказать, что они там его ненавидят и мечтают его запретить, поскольку мир стал бы куда безопаснее, если бы ФБР и прочие силовые агентства могли бы отслеживать веб-трафик каждого.

Вы будете смеяться, но лично мне кажется, что мир стал бы куда безопаснее, если бы все могли читать почту ЦРУ и отслеживать их поиски в Сети, так что сама по себе эта идея, быть может, и неплоха.

Суть *Tor* в следующем: установить сеть промежуточных узлов по всему Интернету и соединять произвольную точку А с точкой В, перескакивая с одного подобного узла на другой.

Фактически, любой поток или даже пакет данных может дойти до цели разными путями. Шифрование вашего общения помешает любому, включая вашего провайдера и всех любопытных Варвар, возмож-

ность сунуть свой нос в ваши сообщения, чтобы выяснить, какие данные вы передаете или с кем вы ими обмениваетесь.

Единственный недостаток здесь в том, что вовлечение такого количества узлов-посредников, к сожалению, делает процесс медленнее, чем вы, возможно, привыкли – это эффективно для электронной почты и передачи файлов, но гораздо менее удобно для VoIP или потоковых устройств.

Сама программа *Tor* шифрует передаваемые данные и управляет путем следования через узлы, но вам все же потребуется дополнительное ПО, чтобы получить полное и завершённое решение по обеспечению конфиденциальности. Вам понадобится прокси-сервер, через который

«Шифрование помешает любому сунуть нос в ваши сообщения.»



► Закупайте лук онлайн в условиях полной конфиденциальности, благодаря превосходной программе *Tor*.

будут перенаправляться ваши запросы, а также ПО конечного пользователя, которое понимает *Tor* и соединяется с ним.

Если вам лень настраивать все это самим, прибегните к предоставленному нами tar-архиву или обращению на сайт *Tor*, где имеется ряд готовых пакетов, включая настроенную для *Tor* версию *Firefox*.

Инструмент производительности

# Timeline

Версия 0.17 Сайт <http://bit.ly/dGqrH>

**В**о время расставить все должным образом по своим местам – важная задача, и очень хорошо, что сей инструмент разработан для выполнения такой задачи, э-ээ, должным образом. Недавно наблюдался всплеск инструментов тайм-менеджмента, но этот непохож на остальных. Он не берется организовать за вас всю вашу жизнь (разве что вы его специально настроите): это просто инструмент создания и управления планами-графиками.

Написанный на Python, *timeline* прекрасен своей простотой. Дисплей отображает линейную временную ось, идущую слева направо, и на нее добавляются события (двойным щелчком и заполнением диалогового окна). События допускаются самых разных типов – особо ценен тип Fuzzy: в нем можно набросать на временную шкалу примерные точки, чтобы оценить масштаб времени.

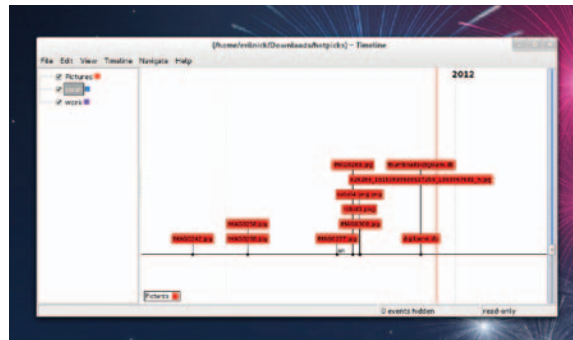
Помимо создания разных типов событий, вы можете добавлять их в категории

(которые сами же и определите) и присваивать им цветовой код, для более легкого прочтения.

Плюс к раскраске разными цветами, можно активировать тумблеры на левой панели для показа или сокрытия определенных категорий; это значительно облегчит работу с извилистыми временными графиками.

Одна из привлекательнейших сторон данной программы – простота управления. Увеличивая и уменьшая масштаб колесиком мыши с клавишей Ctrl, вы можете запросто осуществлять навигацию по мероприятиям, запланированным на послеобеденное время в следующий вторник, и всем остальным событиям нашего столетия.

**«Написанный на Python, timeline прекрасен своей простотой.»**



Мы хотели составить график на все разы, когда Майк покупал выпивку, но оказалось трудно подобрать наглядный масштаб...

Одним только колесом прокрутки мыши вы сдвинете отображение влево или вправо, а уж если совсем заблудитесь, обратитесь к опциям меню или к какой-либо из множества горячих клавиш, чтобы вернуться в известное вам время. Как мы уже упоминали, *timeline* не предназначен для конкретного применения – он с одинаковой легкостью воссоздаст основу вашего рабочего графика или события Гражданской войны в Англии.

Вы даже можете взять за основу временного графика директорию с файлами – попробуйте и посмотрите, что получится!

Социальная сеть

# Gfeedline

Версия 1.6 Сайт <http://code.google.com/p/gfeedline>

**Я**сное дело, когда вы завариваете себе чашку кофе, об этом следует оповестить всю вселенную через *identi.ca* или Twitter. Настоящая проблема не в том, надо ли сообщать тысячам своих поклонников, что вы только что встали или сели, а в том, насколько быстро это удается сделать.

Множество приложений рабочего стола соперничают за звание любимого инструмента для столь важной работы, но тут все зависит от личного вкуса. Одни предпочитают клиент со кучей инструментов для облегчения задачи размещения изображений и слежения за другими, а другим нужно что-то небольшое и ненавязчивое.

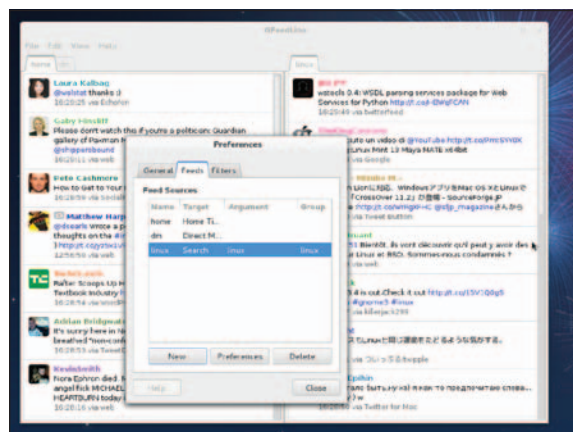
*Gfeedline*, написанный на Python/GTK легковес, относится к последней категории и создает ощущение минимализма. Добавьте учетную запись (вам придется пройти волокиту OAuth), и вы сможете выбирать, что отображать в вашем графике. Сюда входят ленты из разных источников,

и все это отображается во множестве вкладок и множестве панелей, так что вы сможете одновременно отслеживать свои ответы и свой график.

Для изображений имеется поддержка функции Drag-and-drop. Наименее удачная часть – обновление статуса, которое делается во всплывающем окне. Тут есть несколько приятных или полезных моментов, и, возможно, было бы лучше встроить его в основное окно для более быстрого доступа.

*Gfeedline* довольно мило выглядит; при обновлении ленты новые пункты мягко скользят вниз. Кому не нравится стандартный вид Twitter, попробуйте парочку имеющихся тем.

**«При обновлении ленты новые пункты мягко скользят вниз.»**



Дайте знать всему миру о событиях вашей яркой жизни через Gfeedline.

*Gfeedline* требует модулей Python для OAuth и *distutils-extra*. Для установки прилагаемого файла схемы потребуются некоторая ручная работа:

```
sudo cp ./share/com.googlecode.gfeedline.gschema.xml.in /usr/share/glib-2.0/schemas/com.googlecode.gfeedline.gschema.xml
sudo glib-compile-schemas /usr/share/glib-2.0/schemas
sudo python ./setup.py install
```

## Системный инструмент

## Cpulimit

Версия 1.7 Сайт <http://limitcpu.sourceforge.net>

Даже в эпоху многоядерных машин – а может, именно в эту эпоху, поскольку число холостых циклов нередко превышает количество, потребное для рендеринга всего каталога Pixar – неплохо бывает приглядеть за некоторыми процессами и притормозить их.

Естественно, в ядре Linux уже заготовлены для этого инструменты. Вы можете использовать уровень «вежливости [nice]» для уточнения приоритетов и убедиться, что процесс получает свою долю работы процессора, или, намного чаще, что он не блокирует остальные процессы. Nice – дело хорошее, но работает только с приоритетами, а не задает жесткие пределы, как *cpulimit*.

Вкратце, суть *cpulimit* в установке для процесса некоего предела (от 1 до 400), ограничивающего выдаваемое ему время CPU. Предел – 400% (или сколько там позволяет полная нагрузка вашего компьютера) учитывает многоядерность машин, хотя сам по себе предел CPU не ограничи-

вает процессы единственным ядром или чем-то еще – если вы присвоите задаче 100%, весьма вероятно, что эта величина поделится между всеми имеющимися процессами.

Стоит также отметить, что этот инструмент назначает предел, а не гарантирует время CPU – если ваша машина очень занята, отдельной задаче может перепастись куда меньше.

Работать с *cpulimit* довольно просто – вызовите его в командной строке и дайте ему ID и должный предел процесса. Например:

```
cpulimit -p 1738 -l 50
```

В данном случае мы назвали PID и выдали ему предел 50%. Если вы не точно знаете PID процесса, который решили ог-

▶ Пресечь обнаглешившие процессы и повысить КПД работы процессора до того бы это работало еще и на людях...

раничить, он отображается командой вроде `top`, или можно использовать `ps aux | grep 'firefox'`

для выяснения PID определенного процесса.

Это напоминает мне исторический эпизод, когда мы с Майком поехали в Уэстон-супер-Маре [Weston-super-Mare – курортный город в Великобритании, – прим. пер.] и целый день делали ставки на скачках у букмекеров. Выиграть-то мы ни гроша не выиграли, но поломали им компьютеры, навалив непосильную нагрузку на их сервера. А вот был бы у них достойный менеджер CPU, такой проблемы бы не возникло.

«Суть *cpulimit* в установке для процесса некоего предела.»

## Прогнозирование приливов

## Xtide

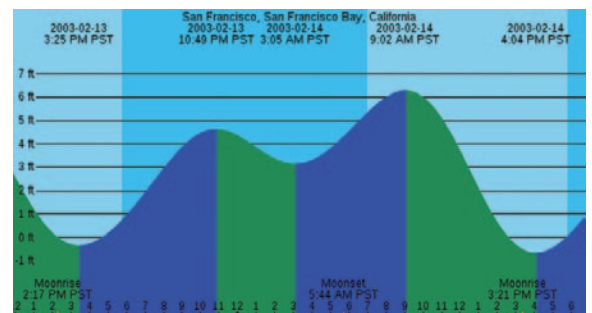
Версия 13 Сайт [www.flaterco.com/xtide](http://www.flaterco.com/xtide)

Иногда поиск программ для HotPicks доводит и до бешенства. Но я говорю сейчас не о всех тех программах, которые не удалось нормально скомпилировать или которые вообще не работают. В данном случае меня взбесили права ИП на публикацию данных.

*Xtide* – отличный инструмент, очень полезный всем вовлеченным в мореплавание. С его помощью легко вычислить приливы для любого места на Земле, при наличии правильных гармонических таблиц данных. Он делает эту работу намного лучше, чем “tide clock [часы, показывающие время прилива и отлива]”, которые и не столь точны, и годятся только для того места, где они установлены. Нужна ему лишь возможность произвести кое-какие вычисления и ряд справочных данные по месту, где вы находитесь. «Гармонические» данные – это записи колебаний уровня приливов в данной местности за определенный период.

Из-за неправильной формы береговой линии точно назвать уровень прилива без этих данных невозможно. К счастью, мы, британцы, будучи островной нацией, собирали информацию о приливах сотни лет – но данные, к несчастью, попали в лапы Гидрографического управления Объединенного Королевства [United Kingdom Hydrographic Office] (<http://www.ukho.gov.uk>), которое сочло нужным предупредить автора *Xtide* о том, что ретроспектива всех собранных гармонических данных не публиковалась и у него нет права их использовать.

В США данные, собранные общественными органами, являются, э-ээ, общественным достоянием. А мы в Вели-



▶ Нам больше не поймут на контрабандном провозе шотландского виски!

британии по неизвестным причинам платим организациям за заботу о данных, которые они нам же и не выдают. Но хватит брюзжать.

Хотя здесь имеются только данные, предоставленные для использования в США, автор объясняет, как сгенерировать собственные гармонические файлы из имеющихся данных (есть ссылки).

Подводя итог, можно сказать, что *Xtide* весьма полезен, если вы живете у моря или собираетесь предпринять морскую прогулку. Запустите его, увеличьте масштаб карты, и перед вами появятся качественный и точный график приливов.

«Xtide – инструмент, полезный всем вовлеченным в мореплавание.»

## HotGames Развлекательные приложения

Ролевая игра

# Dccnightmare

Версия 0.9 Сайт <http://bit.ly/MXOon2>

**Н**екотрые игры, попадающие в HotPicks, исключительно, восхитительно прекрасны. Они полностью завершены и протестированы, и гарантированно захватят вас. *Dccnightmare* (да, на названии язык сломаете) вовсе не обязательно попадает в данную категорию, но все же это очень интересный проект, достойный ознакомления.

Игра принадлежит к жанру RPG. Действие ролевки происходит в вымышленном университете, но прежде чем туда попасть, придется пройти через длительную и веселую процедуру создания вашего персонажа. Кроме солидного набора атрибутов, выбираемых броском костей, можно также выбрать навыки и умения, подобных которым нет в обычных RPG – например, умение найти выпивку, а также более привычные навыки, типа кражи со взломом. Когда дело

доходит до выбора «класса», мы готовы поспорить, что активно используется опция Science Geek.

Сделана игра в 3D – с вращающимся обзором, показывающим вашего персонажа и окружающую его обстановку. Хотя игра пошаговая, все действия анимированы – пока не предусмотрено ничего выдающегося, но, по крайней мере, фигуры могут ходить.

Управление осуществляется мышью и клавиатурой, совместно с панелью инструментов на экране для определенных действий.

Вам, безусловно, не потребуется много времени на осознание, что игра

**«Действие ролевки происходит в вымышленном университете.»**



➤ Ну вылитый университет – разве что нету пустых пивных банок.

несколько недоделана в определенных моментах. Ладно, согласен, сильно недоделана. Обладатели музыкального слуха могут пожелать немедля выключить звук, а если вы ожидаете безупречной графики, то, возможно, вам стоит попробовать предложить свою помощь разработчикам проекта...

*dccnightmare* широко использует SDL, который должен быть установлен, но его очень легко скомпилировать и установить.

Головоломка-пазл

# Billiards

Версия 0.4.1 Сайт [www.nongnu.org/billiards](http://www.nongnu.org/billiards)

**С**тук шаров, чтобы они затем двигались в заданном направлении по относительно гладкой поверхности, вряд ли такая уж захватывающая или хитроумная концепция для основы игры. Но, как ни странно, простого применения законов физики не вполне достаточно, чтобы овладеть *Billiards*.

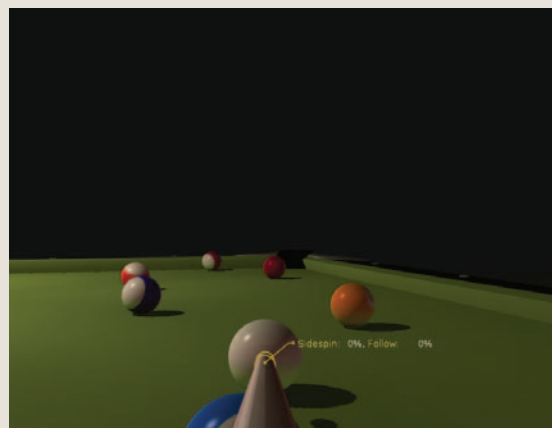
*Billiards* – не просто игра, а попытка стать весьма реалистичным имитатором, с максимально точным следованием физике. Здесь даже можно повозиться с параметрами, чтобы немного преобразовать игру.

Так же как и в классическом бильярде, спорте джентльменов, можно сыграть в производные этой игры, например, пул (8 или 9 шаров), с аналогичным вниманием к деталям. К изумлению, *Billiards* написана на Lua – а точнее, задействует

техне через скрипты Lua. Techne – отличный имитатор физики (<http://www.nongnu.org/techne>), также обладающий прекрасными способностями к рендерингу. Различные режимы затемнений добавляют весь возможный для имитаторов реализм, какой ни пожелать.

Оборотная сторона всего этого – необходимость хорошего 3D-оборудования, чтобы игра шла гладко. Если у вас нет хорошей (то есть выпущенной в последние несколько лет) 3D-видеокарты, повозитесь с различными моделями шейдинга – некоторые из них отнимают значительно меньше времени GPU. Techne

**«Как в бильярде, спорте для джентльменов, можно сыграть в пул.»**



➤ Для тех, у кого только черно-белое отображение: оранжевый шар находится позади голубого...

здорово умеет раскручивать видеокарты по максимуму.

В *Billiards* включен стандартный скрипт настройки. Учтите, что вначале вам нужно установить techne, если вы еще не сделали этого; затем убедитесь, что вы настроили директорию {PREFIX} так, что файлы *Billiards* оказались в нужном месте.

## Инструмент мультимедиа

## Gmerlin

Версия 1.2 Сайт <http://gmerlin.sourceforge.net>

**G**merlin – это медиа-плеер, но, вот диво, он не использует *gstreamer*. На самом деле, *Gmerlin* – витрина для библиотек *gavl*, которые в нем сокрыты. Они позволяют легко справляться с любыми видами медиа-файлов и потоков, и поддерживают громоздкие и сложные способы запаковки упомянутых в цифровые данные. Реализуя разные способы выполнения одной и той же задачи, *gavl* облегчает программистам шансы оптимизации их приложения на предмет, например, скорости или качества.

Видеоизменяемый плеер *Gmerlin* является свидетельством их отличной работы, да и сам по себе неплохой инструмент – «дерево» медиа-источников весьма приемлемо, и поддерживаются сетевые источники, такие как Samba и WMS. И хотя пока что *Gmerlin*, вероятно, скорее выбор разработчиков, что-то полезное из него может извлечь любой. Если вы гонитесь за внешней привлекательностью, обратите также внимание на *lemuria*, которая занимается визуализацией для *Gmerlin*.

Но – пара слов о компиляции этой программы из исходника: возможно, вам стоит найти самый последний пакет «все-в-одном» на сайте, поскольку он включает все зависимости, которых у вас, возможно, нет, и скрипт для компиляции всего чего угодно – просто скачайте его и запустите:

```
sudo ./buildall.sh
```

Хотя скрипт намекает, что пакет *Doxygen* вам не нужен, без него ничего не скомпилируется, даже если проставить правильные параметры в скипте настройки.



» *Gmerlin* – вероятно, единственный медиа-плеер из нами рассмотренных, который не использует *gstreamer*.

## Инструмент для фото

## Rapid PhotoDownloader

Версия 0.4.4 Сайт <http://damonlynch.net/rapid>

**П**рактически любой владелец цифровой камеры, даже если он не относится к типам с полнофункциональной цифровой зеркалкой со многочисленными объективами и кучей фильтров, наталкивается на одну проблему: фотографий слишком много. Использование цифровых технологий означает, что в большинстве случаев люди сделают 50 снимков вместо одного.

*Rapid Photo Downloader* не претендует на решение всех жизненных проблем, но поможет вам разобраться с вашими фотками. Как видно по его названию, основная идея в том, чтобы вам не ждать миллион лет, пока скопируется содержимое вашей карты памяти; но правда такова, что он копирует файлы из одного источника не быстрее, чем практически любая другая программа. Зато он может получать доступ и копировать файлы из разных мест одновременно,

и если у вас более одной читалки карт, он действительно намного обгонит среднее фотоприложение. Еще большее преимущество – то время, которое он вам сэкономит в дальнейшем. Эта утилита разделяет разные типы медиа по разным папкам, и может создавать подпапки с пометкой времени создания с любым именем. Практически мгновенно все ваши файлы размещаются там, где их легко найти. Простой в использовании и простой для понимания, *Rapid Photo Downloader* – сущий ас. **LXF**



» Копируйте из многих источников сразу и тут же размещайте все в нужном месте с помощью *Rapid Photo Downloader*.

## Также вышли

Новые и обновленные программы, тоже достойные внимания...

## » globonote 1.2.4

Красочное приложение Java для заметок сделает ваш рабочий стол ярче. <http://globonote.info>



» *Globonote*: для случаев, когда ваш список дел жаждет яркой раскраски.

## » Domination 1.1.0.9

Risk-подобная стратегия со множеством опций. <http://domination.sourceforge.net/>



» Атаку Ближний Восток за три броска костей с... упс, минуточку...

## » Vox 0.3.4

Довольно симпатичный графически-ориентированный язык, который весело изучать. <http://sourceforge.net/projects/boxc>

## » Fabula 0.8.3

Система генерирования игр-приключений на основе Python. <http://fabula-engine.org>

## » Expect-lite 4.3.1

Истинное благословение для любителей создавать автоматические скрипты оболочки. <http://expect-lite.sourceforge.net/>

## » LBreakout 2.6.4

Если вам надоело пинать одни и те же кирпичи, создавайте свои собственные уровни. <http://lqgames.sourceforge.net>

## » phpMyAdmin 3.5.2rc

Свежайшая версия самого популярного в мире программного обеспечения для администраторов MySQL. <http://www.phpmyadmin.net>

# На диске

Пробуйте новую операционную систему уже сегодня!



**В** этом месяце не было крупных релизов дистрибутивов, что дает нам возможность обратиться к тем дистрибутивам, о которых мы, в силу разных причин, некоторое время не рассказывали. Мы также создали собственный дистрибутив, чтобы помочь вам сравнить вашу машину с теми, которые мы тестировали в нашем материале. DVD прошлого месяца был рассчитан на 64-битные ма-

шины, потому что все дистрибутивы создавались для современного оборудования. Теперь мы сместили баланс: на DVD этого месяца 64-битным является только наш ремикс Ubuntu, предназначенный для тестирования производительности; FreeNAS представлен в обоих вариантах; а все остальные – 32-битные. Vector и FreeNAS включены как ISO-образы, и их можно прожечь на CD.

Лучший экспорт из России – причем не алкоголь

## Rosa

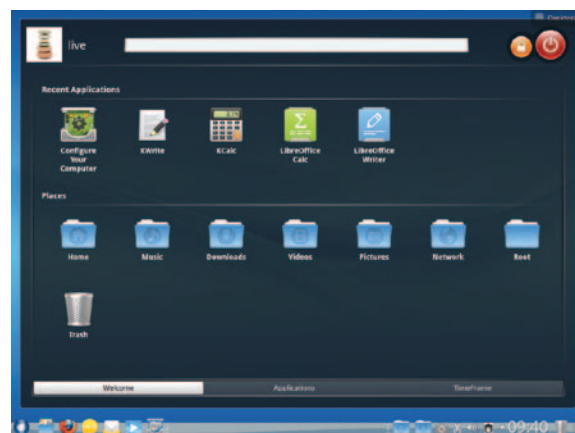
**R**osa – дистрибутив, основанный на KDE, но развернувший KDE в новом направлении: серия индивидуальных настроек придала ему вид скорее OS X, нежели обычного KDE. Как и все изменения рабочего стола, это, вероятно, взбесит и восхитит равное количество пользователей. За такой стиль интерфейса пытались взяться несколько разных проектов, и это – одна из лучших попыток, хотя и весьма ресурсоемкая.

Два из дополнений KDE – *Klook* и *Stack Folders* – отосланы прародителю, и, ве-

роятнее всего, войдут в будущие релизы KDE. *Klook* выдает предпросмотр многих типов мультимедиа прямо из *Dolphin*, а *Stack Folders* предоставляет способ увидеть содержимое папки без ее открытия.

Релиз *Marathon* – более стабильный, ориентированный на бизнес. Rosa также выпускает вариант *Desktop*, с самыми авангардными версиями программ. Оба варианта выпускаются ежегод-

«Klook выдает предпросмотр мультимедиа прямо из Dolphin.»



» Rosa: возможно, это – будущее KDE.

но, друг от друга их отделяет полугодичный перерыв, так что ждите появления на серверах Rosa Desktop приблизительно к Рождеству.

Проверьте темп своей системы

## Ubuntu LXF Benchmarking

Итак, вы прочли тему номера и хотите сравнить свой компьютер с нашими. Что ж, это возможно, благодаря нашему live-дистрибутиву, который позволит вам запустить те же самые тесты в собственной системе. После загрузки вы должны увидеть на рабочем столе два файла: **LXF Large Benchmark** и **LXF Small Benchmark**. Последний запустит пакет тестов, содержащихся на DVD. Время, затраченное на них, зависит от вашей системы, но в среднем должно составлять около 15 минут для относительно

мощной машины. Большой пакет запустит больше тестов и требует данных – скачать придется около 1,5 Гб. Отдельные тесты запускаются командой

```
phoronix-test-suite benchmark pts/test
```

где *test* – это требуемый тест. Чтобы просмотреть все тесты, скомандуйте

```
phoronix-test-suite lists-tests
```

Пакеты тестов производительности **LXF benchmark** запускаются с помощью

«Большой пакет запустит больше тестов и требует данных.»



» Путь это не победа, но участие тоже немаловажно.

```
phoronix-test-suite benchmark lxf-small
```

```
phoronix-test-suite benchmark lxf-large
```

Первое, в отличие от второго, не требует интернет-соединения. Наш дистрибутив 64-битный, потому что это влияет на работу тестов, но тесты можно запустить и на 32-битном ПК. Подробности см. на странице Phoronix HTML на DVD.



Маленький и шустрый

# Vector

**V**ector Linux разрабатывался под девизом «Сохраняй простоту и малый размер, и пусть пользователь решит, какой будет его операционная система» [аллюзия на слова Гэндалфа из романа «Властелин колец», — прим. пер.]. Эта философия привела к разработке компактной и легковесной ОС. Но не думайте, что система сложна в работе или предназначена для продвинутых пользователей. Отказавшись от ненужных эффектов, разработчики создали дружелюбный рабочий стол, непритязательный к ресурсам.

Приложения по умолчанию тоже придерживаются философии экономии ресурсов, и вместо *Libre Office* сюда включен *Gnome Office*.

Версий Vector Linux существует несколько, и мы включили Standard

Live. Если вам понравилась сама идея и захотелось пройти еще дальше, есть Light edition, а Soho edition идет с полным рабочим столом KDE и *Libre Office*. Поскольку мы включили на DVD версию Live, опция установки отсутствует.



➤ Наши учителя математики определяли Вектор как направленную величину, но мы-то знаем, что это дистрибутив Linux.

## Важно ВНИМАНИЕ!

➤ Прежде чем вставить DVD в дисковод, пожалуйста, убедитесь, что вы прочитали, поняли и согласились с нижеследующим.

Диски *Linux Format* DVD тщательно проверяются на предмет отсутствия на них всех известных вирусов. Тем не менее, мы рекомендуем вам всегда проверять любые новые программы надежным и современным антивирусом.

Хотя процесс отбора, тестирования и установки программ на DVD проводится со всем тщанием, редакция *Linux Format* не несет никакой ответственности за повреждение и/или утрату данных или системы, могущее произойти при использовании данного диска, программ или данных на нем. Настоятельно рекомендуем вам создавать своевременные и надежные резервные копии всех важных файлов.

Чтобы узнать об условиях использования, просим вас прочесть лицензии.

### Бракованные диски

В маловероятном случае обнаружения бракованного диска *Linux Format*, просим связаться с нашей группой поддержки по адресу [disks@linuxformat.ru](mailto:disks@linuxformat.ru), для получения содействия.

А вы полегче!

# Fedora LXDE

**F**edora — это замечательно, если вы подыскиваете дистрибутив, разрабатываемый сообществом, и с самым свежим ПО. Однако его установка по умолчанию с Gnome 3 может оказаться тяжеловата для некоторых компьютеров и простовата для некоторых пользователей.

К счастью, помимо основной версии, команда Fedora поддерживает ряд от-

ветвлений. Они, как и ремиксы Ubuntu, используют те же репозитории пакетов, что и основная версия, но снабжены другими программами по умолчанию.

Мы включили вариант с LXDE, который отлично подходит для менее мощных систем или для тех случаев, когда незачем тратить системные ресурсы на отработку излишней графики. Но если вам это не подходит, есть множество других от-

ветвлений — с KDE, играми и дизайном — и вы сможете выбрать то, что придется по вкусу лично вам. **LXF**

«К счастью, команда Fedora поддерживает ряд ответвлений.»



➤ LXDE нетребователен к ресурсам, и набирает все больше пользователей, потому что прост в использовании.

## На диске

Помимо четырех отличных дистрибутивов, мы включили весь код из учебников, все новинки и все программы из всех разделов журнала, где вы видели плашку **На диске**. При установке DVD в ваш компьютер он должен по умолчанию открыть HTML-страницу; но если ваши настройки безопасности этого не разрешают, перейдите в [index.html](#), чтобы открыть для себя все 4,4 ГБ прелестей Linux. Для ознакомления с нашими дистрибутивами уста-

новите диск в дисковод, затем перезагрузите компьютер. Откроется экран, где вы сможете выбрать то, что вам нужно. Если ваш компьютер загружается как обычно, а не с диска, вам придется вручную изменить настройки BIOS, чтобы активировать загрузку DVD. Если вы предпочитаете перенести содержимое диска на USB-брелок, мы включили для этого удобный скрипт. Загляните в раздел [dvd2usb.html](#) на диске, чтобы узнать, как он рабо-

тает. Подписчики цифровой версии, обратите внимание, что вы можете просто использовать **dd** для переноса скачанного ISO на USB, если вам это потребуется. Используйте `sudo dd if=lx157.iso of=/dev/sdX` при необходимости включив путь до скачанного ISO, где **X** изменяется соответственно для USB-брелка. **Это уничтожит все данные на диске. Если вы введете не тот диск, то сотрете весь жесткий диск.**

## Пропустили номер?

» Мир свободного ПО богат и разнообразен, а потому далеко не все можно вместить в рамки одной статьи. Linux Format обходит эту проблему, публикуя серии статей по самым актуальным вопросам. Но что делать, если вы поймали интересующий вас материал на середине? Обратитесь в «Линуксцентр» по адресу [www.linuxcenter.ru](http://www.linuxcenter.ru) и закажите желаемый номер журнала! Он доставляется как в печатной, так и в электронной форме, поэтому с момента открытия браузера и до получения нужного вам выпуска LXF может пройти не более нескольких минут!

Прямо сейчас для заказа доступны следующие номера:



**LXFI58**  
Июнь 2012

250 руб.

- » **Надем ЦРУ** Как не попасться во Всемирную паутину
- » **Братья наши меньшие** Альтернативные ОС
- » **The Journal** Новый метод ведения системных журналов
- » **FontForge** Рисуем собственный шрифт

**LXFDVD:** Tails и еще 3 дистрибутива, Audacity, FontForge, KtechLab, Qemu 1.0.1 и прочее...

Печатная версия: [shop.linuxformat.ru/lxf\\_158/](http://shop.linuxformat.ru/lxf_158/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_158/](http://shop.linuxformat.ru/elxf_158/)



**LXFI59**  
Июль 2012

250 руб.

- » **Linux против Windows** Если слон на кита налезет, кто кого соберет?
- » **Рулим обновлениями** Менеджеры пакетов – без них никуда
- » **OpenCL** Открытый стандарт для параллельного программирования
- » **Презентации** Как обаять аудиторию и заразить своей идеей

**LXFDVD:** Ubuntu 12.04 LXF Remix и еще 3 дистрибутива, FWBuilder, LibreOffice 3.5.2 и прочее...

Печатная версия: [shop.linuxformat.ru/lxf\\_159/](http://shop.linuxformat.ru/lxf_159/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_159/](http://shop.linuxformat.ru/elxf_159/)



**LXFI60**  
Август 2012

250 руб.

- » **Red Hat** Как заработать миллиард долларов
- » **Общаемся в открытую** Социальные сети, уважающие приватность
- » **OpenCL** Открытый стандарт для параллельного программирования
- » **Кластер всем миром** Подключаем свой ПК к прогрессу науки

**LXFDVD:** Mageia 2 и еще 3 дистрибутива, BCCD, Fotoxx, YAGF, руководство администратора Debian и прочее...

Печатная версия: [shop.linuxformat.ru/lxf\\_160/](http://shop.linuxformat.ru/lxf_160/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_160/](http://shop.linuxformat.ru/elxf_160/)

Ну, а если вы хотите быть уверенными, что не пропустите ни один номер журнала – оформите подписку! Помните, что все подписавшиеся на печатную версию журнала через [www.linuxcenter.ru](http://www.linuxcenter.ru) или [shop.linuxformat.ru](http://shop.linuxformat.ru) получают электронную версию в подарок!

Подписывайтесь на журнал на [www.linuxformat.ru/subscribe/](http://www.linuxformat.ru/subscribe/)

Телефоны отдела подписки: Санкт-Петербург (812) 309-06-86, Москва (499) 271-49-54

## Специальное предложение

### Купите подборку журнала!

К нам в редакцию периодически приходят письма с вопросами, где можно купить предыдущие выпуски LXF. Если вы тоже этим озадачены, то в интернет-магазине «ГНУ/Линуксцентра» продолжается продажа журналов за 2011 год. Вы можете приобрести как отдельные номера изданий, так и подписки на 6 или 12 месяцев.

Спешите – журналов осталось не так уж много!

[shop.linuxformat.ru](http://shop.linuxformat.ru)



## Информация о диске

### Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

### Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или 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](mailto:disks@linuxformat.ru)

**Внимательно прочтите это перед тем, как использовать LXF DVD!**

**А ТАКЖЕ:** Видеоурок по работе с микрокомпьютером MIK802

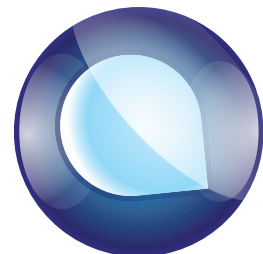
## Linux Mint 13 “Maya”



» Ядро 3.2 » Mate 1.2 или Gnome 1.4 » MDM 1.0.4 » Firefox 12 и Thunderbird 12

Сборки с Mate, Gnome, KDE и Xfce

**А ТАКЖЕ:** Phoenix Test Suite, Digikam, Packet Fence и многое другое...



Marathon 2012  
**ROSA**

Российский «потомок» Mandriva

ПЛЮС ЕЩЕ ТРИ ДИСТРИБУТИВА:

» Fedora 17 LXDE spin » FreeNAS 8.0.4 » Vector Linux 7.0

Сентябрь 2012  
**LXF DVD 161**

**LINUX**  
FORMAT

Сентябрь 2012  
**LXF DVD 161**

**LINUX**  
FORMAT

# Содержание

# LINUX FORMAT

## Сторона 1

### ДИСТРИБУТИВЫ

**Bench Live CD** для загрузки Photnix Test Suite,  
64-разрядная сборка (загрузка с LXFDVD)  
**FreeNAS 8.0.4** Сетевая система хранения данных на базе  
FreeBSD, 32- и 64-разрядная сборка (ISO-образы)  
**Fedora 17 LXDE spin** Вариант популярного дистрибутива  
с рабочим столом LXDE, 32-разрядная сборка  
(загрузка с LXFDVD)  
**ROSA Marathon 2012** Российский дистрибутив  
на основе Mandriva, 32-разрядная сборка  
(загрузка с LXFDVD)  
**Vector Linux 7.0** Облегченный дистрибутив для старых  
компьютеров, 32-разрядная сборка (ISO-образ)

### ПРОГРАММЫ

**Photnix Test Suite 3.8.0** Набор тестов для оценки  
производительности аппаратуры

### НОТРИСКС

**Billiards 0.41** Симулятор игры на бильярде  
**CPU Limit 1.7** Средство управления  
диспетчеризацией процессов  
**DCC Nightmare 0.9** Однопользовательская  
ролевая 3D-игра  
**DigiKam 2.6.0** Популярная программа для управления  
коллекциями фотографий  
**GFeedline 1.6** Клиент для общения в социальных сетях  
**Gmetlin 1.2.0** Набор библиотек и приложений,  
образующих мультимедиа-архитектуру Linux  
**Packet Fence 3.4.1** Система обеспечения безопасности  
беспроводных сетей  
**Rapid Photo Downloader 0.4.4** Графическое приложение  
для загрузки фотографий из фотокамеры  
**Timeline 0.17** Кросс-платформенная программа  
для управления расписаниями

## Сторона 2

**Mint 13** Установочный CD с рабочим столом Cinnamon,  
64-разрядная сборка (ISO-образ)  
**Mint 13** Установочный CD с рабочим столом Mate,  
64-разрядная сборка (ISO-образ)  
**Mint 13** Установочный CD с рабочим столом KDE,  
64-разрядная сборка (ISO-образ)  
**Mint 13** Установочный CD с рабочим столом Xfce,  
32-разрядная сборка (ISO-образ)

### ДОКУМЕНТАЦИЯ

Видеурок по работе с микрокомпьютером **MK802**

**Пожалуйста, перед использованием  
данного диска ознакомьтесь  
с опубликованной в журнале  
инструкцией!**

**КОММЕНТАРИЙ** Присылайте ваши пожелания и предложения по электронной  
почте: [info@linuxformat.ru](mailto:info@linuxformat.ru)

**Дорогие диски!** В маловероятном случае обнаружения дефектов  
на данном диске, обращайтесь, пожалуйста, по адресу [disk@linuxformat.ru](mailto:disk@linuxformat.ru)

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам  
использовать авторизованный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашей 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, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер,  
подготовьте резервную копию данных и сделайте резервную копию.

Тираж изготовлен ООО «Уральский электронный завод», 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПР ВАР 77-15.

## Создание установочных дисков при помощи cdbrecord

Самый быстрый способ записать ISO-образ на чистую матрицу – это через *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке – например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле `/etc/default/cdrecord`. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor=0,3,0 12 16M
```

Первое слово в этой строке – метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root 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?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере с Windows, Mac OS X, AmigaOS или любой другой ОС.


## Нет устройства для записи дисков?

А что если у вас нет устройства, с помощью которого можно было бы записать образ на диск? Вы знаете кого-нибудь с таким устройством? Вам не обязательно использовать Linux для записи дисков: подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.

# ФОРУМ №1

ДЛЯ ВСЕХ ПОЛЬЗОВАТЕЛЕЙ LINUX



**LINUXFORUM.RU**

**LinSoft.info**  
Путеводитель по программному обеспечению для GNU/Linux



**WWW.LINSOFT.INFO**

**Linux по-русски**

**Библиотека книг, статей и переводов о Linux**

**WWW.RUS-LINUX.NET**

Аппаратно-программный комплекс

# DR.WEB OFFICE SHIELD

Комплексное решение задачи антивирусной и антиспам-защиты для малых и средних предприятий



88 679 руб.

**DR.WEB OFFICE SHIELD TWISTER**  
Рассчитан на 250 ПК



65 240 руб.

**DR.WEB OFFICE SHIELD NEO**  
Рассчитан на 50 ПК

**Dr.WEB®**

**Linux center**  
www.linuxcenter.ru

[WWW.LINUXCENTER.RU/SHOP/ANTIVIR/DR\\_WEB/](http://WWW.LINUXCENTER.RU/SHOP/ANTIVIR/DR_WEB/)

# LINUX FORMAT

Главное в мире Linux

## Как разместить рекламу в разделе Classifieds?

¼ полоса (210 × 297 мм)	165 200 руб.
½ полосы горизонтально (197 × 144 мм)	88 500 руб.
½ полосы вертикально (102 × 278)	88 500 руб.
¼ 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

**Linux center**  
www.linuxcenter.ru

*Отдел дистрибьюции ГНУ/Линуксцентра приглашает дилеров и дистрибьюторов к сотрудничеству!*

**Широкая сеть представительств в разных городах позволит вам оптимизировать процессы логистики и доставки товара.**

Подробнее о партнерской программе:  
[www.linuxcenter.ru/partner/](http://www.linuxcenter.ru/partner/)

Международная **XXXII КОНФЕРЕНЦИЯ ВЫСТАВКА**  
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

# ИТО

Место проведения:  
Москва  
Ленинские горы  
Московский государственный университет имени М. В. Ломоносова  
2-й учебный корпус  
дом 1, строение 52

**7-9 ноября**

<http://ito.su> email: [info@ito.su](mailto:info@ito.su)

Конференция проходит при поддержке:



**Unix Education Center**  
Россия, 190000, Санкт-Петербург  
Черноморский переулок, дом 4  
Тел.: +7 (812) 611-1575



**ГНУ/Линуксцентр приглашает на работу!**

**Linux center**  
www.linuxcenter.ru

**ВАКАНСИЯ: PHP-программист/web-мастер**

**ОБЯЗАННОСТИ:**

- » Создание сайтов с нуля на базе CMS Drupal: верстка, программирование, разработка баз данных, конвертация данных.
- » Разработка дополнительных модулей для CMS Drupal.
- » Внесение изменений в готовые скрипты.
- » Поддержка сайтов компании.

**ПОДРОБНЕЕ:** [www.linuxcenter.ru/vacancy/](http://www.linuxcenter.ru/vacancy/)

# Школа LXF

Спонсор рубрики  
**Mandriva.ru**  
разработчик  
дистрибутива  
EduMandriva  
[www.mandriva.ru](http://www.mandriva.ru)

Обмен опытом и передовые идеи по использованию свободного ПО в образовании

## СПО в школе: Linux или Windows?

Еще один репортаж с места событий: **Артем Зорин** побеседовал с учителем, внедрявшим Linux в сельской школе.

**Н**аш собеседник — Райво Штулберг, учитель информатики Ольховской средней общеобразовательной школы Рязанской области, по совместительству писатель и компьютерный энтузиаст. Окончил педагогический факультет Рязанского государственного университета по специальности «преподаватель иностранного языка», но волею судьбы преподает информатику.

Райво согласился поделиться с нашим журналом своим опытом внедрения СПО в маленькой сельской школе. Диалог проходил в форме интервью посредством электронной почты. Это не история успеха: это история внедрения, борьбы с проблемами, личное мнение среднестатистического продвинутого пользователя об СПО и идее его внедрения в российских школах. Также, автор статьи давно и тесно общается с Райво и считает его своим другом, но не считает, что Райво во всем прав. Наши читатели помнят проект по внедрению Linux и СПО в школах, который завершился полным провалом, но кое-где все же удался. Райво сделал все от него зависящее, чтобы внедрить СПО в своей школе; что из этого вышло — читайте...

**LXF:** Как и при каких обстоятельствах ты познакомился с СПО в целом и с Linux в частности?

**Райво Штулберг:** Год уже не вспомню точно, но где-то в начале 2000-х состоялось мое первое заочное знакомство с Linux. Как-то в журнале «Мир ПК» увидел статью о новой для меня операционной системе. Сейчас даже не помню, про что именно там говорилось, помню лишь пингвина на иллюстрации. Потом подобные статьи начали встречаться все чаще и чаще. Не сказать, чтобы они вызывали какой-либо особый интерес с моей стороны, я их читал разве что в порядке самообразования. Через некоторое время в книжном магазине Рязани увидел коробочную версию какого-то дистрибутива (скорее всего, это был Red Hat, но не уверен). На школьном кружке информатики в качестве ознакомительного материала я давал ученикам информацию о Linux, но лишь обзорно. И уж ни в коем случае не подозревал, что пройдет совсем немного времени — и придется столкнуться с этой ОС вплотную...

**LXF:** Как проходил процесс перевода школы на Linux? Какие при этом возникали трудности?

**РШ:** Сумбурно, бессистемно и самотеком — вот вкратце как можно описать этот переход. Прислали коробку с дистрибутивами ALT Linux, никакой сопутствующей инструкции в помощь не прилагалось. Это были ALT Linux Легкий, Юниор, Терминал и Мастер — стандартный набор от «Армады». Сразу после этого я попытался установить на домашнем компьютере ALT Linux Мастер, но, разобравшись в особенностях организации файловой системы, едва не отформатировал весь жесткий диск. Типичная ошибка новичков, в общем-то. Решил, что лучше не рисковать, и запустил Live-версию какого-то дистрибутива из того набора. Честно говоря, впечатления не произвело. Ну, Линукс... Ну, те же окна...

А примерно через полгода учителей информатики нашего района записали на какие-то дистанционные курсы, в РУНО выдали диск со статьями и сказали: «Готовьтесь. Через три месяца будете сдавать тестирование». Мы и готовились. Не знаю, как бы «подготовился» я, не будь под рукой Интернета и линуксоида со стажем — друга Артема. В частности, он помог мне с установкой Мастера на мой рабочий нетбук (примечательно, что весь инструктаж происходил по ICQ). Тогда же я установил на одну из школьных машин (тогда в обязательном порядке организовывались так называемые «опытные зоны») ALT Linux Легкий.

А потом мы сдавали обещанное тестирование. Фарс — и больше ничего. Очаровательная девушка сразу сказала, что мы можем списывать, откуда хотим. Знай я, что будет такая возможность, непременно прихватил бы шаргалки.

Но пришлось сдавать «настоящему». В результате, через несколько недель на руки получил сертификат, подтверждающий, что я являюсь специалистом по администрированию Linux.

Еще через полгода РУНО организовало выездные курсы, где сухонький старичок-живчик бодро показывал нам, как устанавливать Linux. Меня, естественно, откомандировали на те курсы, но абсолютно ничего нового оттуда я не вынес, так как к тому времени переинсталлировал и перепробовал уже не один дистрибутив самостоятельно. Интересно же, что многие из прибывших —

### ПРО ПЕРЕХОД ШКОЛ НА LINUX

«Сумбурно, бессистемно и самотеком — так можно описать этот переход.»

учителя информатики, кстати – не имели понятия о том, как... войти в BIOS для изменения порядка загрузки ОС. Некоторые из них, вместе со мной, в свое время получили сертификаты...

И было 1 января, и всем школам приказали либо закупать собственные лицензионные Windows, либо устанавливать Linux. Многие же десятки дисков из «Первой Помощи» от Microsoft оказались вне закона, так что по инструкции их требовалось физически ликвидировать. Что мы и сделали. А совсем недавно начали поднимать вопрос о продлении лицензии тех самых дисков. Возникает вполне естественный вопрос: «Лицензия будет продлеваться на разбитые болванки?..»

Но школьные Альты (ALT Linux) я убрал подальше и вернулся к ним всего пару раз за все время. У себя же в классе перепробовал достаточно много дистрибутивов (купленных, кстати, за собственные деньги). В разное время через мои компьютеры прошли: Debian, Ubuntu, ALT Linux Ковчег, Slackware, Solaris (!), Knoppix. Пока остановился на Debian, Ubuntu и Ковчеге. Лучшим из них признаю все-таки Debian. Ubuntu тоже показал себя неплохо, но возникли проблемы с кодеками, а как следствие – с воспроизведением аудио и видео. Подключение же класса к Интернету мне только снится. А тратить собственные деньги на связь от мобильных провайдеров я не намерен.

Кроме вышеперечисленных, были еще трудности. Отмечу лишь несколько. Отсутствие должной квалификации – это раз. Второе – отсутствие быстрого стабильного Интернета в сельской школе, когда ко Всемирной Сети подключена всего одна машина, да и та находится в учительской. Как только добропорядочные учителя информатики установили Linux, их школы моментально остались без Интернета. Провайдеры также отказывались оказывать помощь. То ли в силу лени, то ли в силу технической неосведомленности. Но не мне вам объяснять, как важен Интернет при работе с Linux.

Насколько мне также известно, добрая половина принтеров до сих пор отказывается печатать из-под Linux. Лично в моем классе, изо всех машин с принтером «подружилась» лишь одна, работающая под Debian. Также постоянно возникают проблемы со сложным форматированием во время перехода документов из среды Linux в ОС Windows и наоборот. Текст «плывет», таблицы порой вовсе куда-то «проваливаются» за границы листа, и вернуть их назад нет никакой возможности. Да и вставка рисунков в документ – тема больная.

Кроме того, до сих пор ни одна интерактивная доска не работает ни под одним Linux. Вот и пылятся они в подсобках без дела...

А что прикажете делать со многими образовательными программами, которые запускаются исключительно под Windows? Иные из них весьма полезны и на самом деле могли бы оказать огромное подспорье в работе учителя. И *Wine* здесь не выход. Вопреки заверениям разработчиков, из-под *Wine* не запускается даже Аттестат-Экспресс (программа для печати дипломов). В этом году взял грех на душу – на два дня поставил «пиратскую Винду», иначе выпускники остались бы без дипломов. И дома печатать – не вариант: Аттестат-Экспресс лицензирован лишь под одну конкретную машину.

И уж точно надо посочувствовать учителям информатики. Все мы так или иначе выкручиваемся и ищем точки соприкосновения и аналогии с Windows в Linux. Получается где-то лучше, где-то хуже. Но то, что Linux-учебников для уроков информатики нет и не предвидится – факт. Сказывали, будто существует один такой в природе, в Питере. Так ли это – доподлинно мне не известно.

В этом году (2011) запущена также программа по переходу школ на ФГОС (Федеральные государственные образовательные стандарты). Компьютеры теперь должны стать неотъемлемой частью урока и сопровождать ученика едва ли не со второго сентября первого класса. Да вот незадача: все, абсолютно все об-



► Райво Раймондович Шгулберг — человек, несмотря ни на что, внедривший Linux в школе.

разовательное ПО работает в Windows. О каком таком переходе на какие такие новые образовательные стандарты смеет говорить Министерство образования?! Видимо, они посчитали «линуксизацию» школ давно и успешно свершившимся фактом. На деле же до этого еще ой-ой как далеко!

**LXF:** Что, по твоему мнению, было сделано не так при реализации программы перевода российских школ на Linux и СПО?

**РШ:** Ох, да все не так! Сам перевод был одной большой ошибкой. Президент, видимо желая прославить свое имя в веках, совершил ряд абсолютно пустых и ненужных реформ «на ровном месте»; перевод школ на СПО относится сюда же. Вполне понятно, что Дмитрий Анатольевич (речь идет о 2011 годе) не в курсе идеологии Открытых систем, Открытого кода и т.п. – он услышал слово «бесплатно» и дал отмашку. Чиновники же не замедлили, как всегда, воспользоваться данным моментом для набивания собственных карманов.

Уж сколько миллионов «отмыли» и попросту наворовали на этом деле – вопрос в Прокуратуру. Но в итоге все получилось не то что не бесплатно, а во многие разы дороже. Гораздо дешевле было бы закупить для всех школ лицензии на Windows. А уж сколько сэкономили бы учительских сил и нервов...

Вопрос «что было сделано не так» теперь звучит праздно. Все уже сделано. Криво, безобразно, безответственно, бессистемно. Самая главная ошибка была в спешке, в которой проводился перевод. В непродуманности. Подозреваю, что не было никакой программы. Либо она была, но лишь на бумаге, как это все в России и происходит с незапамятных времен.

**LXF:** Как ты относишься к идее перевода государственных органов России на СПО?

»



Ученики Ольховской средней школы на уроке информатики. Заметно недоумение на их лицах...

**РШ:** Отрицательно. Последуют многие технические сбои, недо-разумения и прочие неприятные и порой просто опасные вещи, связанные с технической неподготовленностью чиновников. Многие и в Windows «ни в зуб ногой», так куда ж таким в Linux со-ваться? Впрочем, тому, кому пришла первому в его светлую го-лову идея о внедрении СПО в школы, я бы лично установил один из школьных ALT Linux на компьютер и за-ставил бы поработать с ним годик-другой.

## ПРО LINUX ВОООЩЕ

**«Linux — прекрасная систе-  
ма, но... лишь для своих нужд.  
Впрочем, как и всякая вещь.»**

**LXF:** Что тебе не удалось, а что удалось сделать?

**РШ:** Да ничего не удалось. Дистрибутивы установлены. А толку?..

**LXF:** Доволен ли ты проделанной работой?

**РШ:** Отвечу так: все, что от меня зависело, я добросовестно выполнил.

**LXF:** Есть ли польза от этого?

**РШ:** В школах СПО — лишнее.

На некоторых сайтах приходилось читать восторженные от-зывы от учителей и учеников в связи с переходом школ на Linux. Уж не знаю, настоящие ли те отзывы были, либо... Только хоте-лось бы задать вопрос: у кого из этих восторженных на домашних компьютерах стоит Linux?..

Linux — система прекрасная. Но она никоим боком не при-менима к школам. По крайней мере, уж точно не таким образом, как это было наспех слеплено в нашей стране. Не открою Амери-ки, но на самом деле все очевидно: там, где нужен Linux — надо ставить Linux. В школах же он не нужен. Это мое мнение, вы воль-ны с ним согласиться или нет. Линуксизация же школ в России провалилась с глухим треском. Падать начнем совсем скоро.

**LXF:** Как отнеслось руководство школы и твои коллеги к пере-ходу на Linux? Как отреагировали ученики?

**РШ:** Руководство, как и многое руководство, отнеслось с недо-умением. Человеку, далекому от мира ПК, довольно трудно разъ-яснить, отчего вдруг стало невозможным отправить важное пись-мо по электронной почте, или хотя бы просто распечатать тот или иной документ. Или — как рассказывать директору школы о том,

что завтра выпускники не получат свои аттестаты об окончании школы?..

То же самое касательно многих коллег. Те же немногие из них, кто имеет представление о компьютере и даже активно его ис-пользует в образовательном процессе, переход восприняли, как и я, крайне болезненно и с нескрываемым раздражением. В силу всех вышеперечисленных причин.

Ученики же особенно никак не отреагировали. У большинства из них дома стоят Windows-системы, и до Linux им нет совер-шенно никакого дела. Разве что на уроке информатики то, что на-писано в учебнике, идет вразрез с тем, что установлено на их ра-бочей машине в классе. Приходится постоянно перестраиваться, перепланировать ход урока как в частности, так и в целом. Но это — проблема для учителя. Для ученика же это все должно произойти как можно менее болезненно. Впрочем, куда потом подадутся бу-дущие выпускники со знаниями Linux и кто их будет переучивать на Windows — те еще вопросы...

Мне возражат: пусть-де дома изучают Windows самостоятель-но, а в школе — Linux как дополнительную систему. В таком слу-чае, пусть, скажем, сведения по истории России черпают дома из детских исторических рассказов (ведь не секрет, что у многих ПК дома используется исключительно в качестве игровой маши-ны), а в школе — ну, как дополнительную дисциплину — подробно изучают историю африканского племени Мумбо-Юмбо...

**LXF:** Твое отношение к Linux и СПО в целом?

**РШ:** Положительно отношусь. Linux — прекрасная система, но... лишь для своих нужд. Впрочем, как и всякая вещь в этом мире.

Я знаком с историей становления и разви-тия Открытого ПО в ми-ре и глубоко уважаю людей, занимающих-ся разработками в этой области. Беда же российского образо-

вания (и не только касательно введения Linux, а вообще — образо-вания) в том, что у руля стоят люди, страшно далекие от истинных нужд школы. Наспех проведенные реформы, необходимости в ко-торых, по большому счету, и не было, незаконное «освоение», как теперь говорится, бюджетных средств — это также боль всей Рос-сии. В сфере же образования это, как всегда, сказывается наибо-лее остро. А компьютеризация школ — и подавно боль. Впрочем, это отдельная тема для отдельного большого разговора.

И последнее. Я лишь попрошу тех, от кого это зависит: «Верни-те нам Windows!» **LXF**

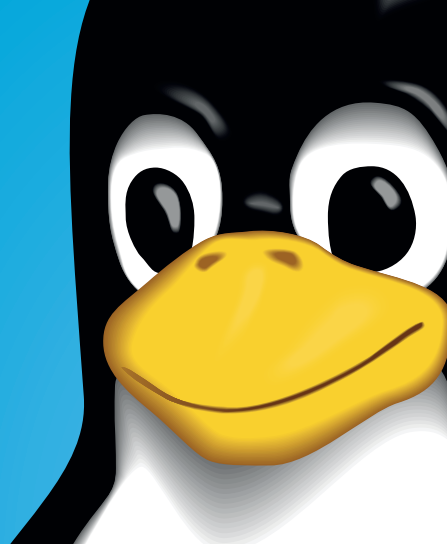
## Мнение редакции

Отметим, что редакция Linux Format не всегда разделяет мнение гостей журнала, и не может согласиться с некоторыми тези-сами, высказанными Райво. Однако стоит учесть, что ему уда-лось-таки перевести пусть и маленький, но все же парк машин на Linux, и сейчас идет процесс обучения остальных сотрудни-ков школы работы с новой ОС. Кстати, если вы хотите подробнее узнать о Райво, почитать его произведения, добро пожаловать на один из его личных сайтов: <http://litdao.narod.ru/>, а также стра-ничку ВКонтакте: <http://vk.com/id59568570>, или пишите Райво на электронную почту, он с удовольствием ответит на все ваши вопросы: [sht-rajvo@yandex.ru](mailto:sht-rajvo@yandex.ru).

Все торговые марки, адреса сайтов и прочие важные штуки, фигурировавшие в данном интервью, ни в коем случае не явля-ются рекламой и служат лишь в познавательных целях.



# 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:

- 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](http://www.linuxcenter.ru)

Linux  center



В октябрьском номере

# Чемпионат дистрибутивов

Берем античную Олимпиаду, заменяем людей на дистрибутивы Linux и удаляем всякие непродуктивные корпоративные бренды – и вот вам результат!

## Давайте поспорим

Vim против Emacs, KDE против Gnome, Linux против GNU/Linux... полемика в Linux-сити (и сети) так и кипит.

## Идеальная документация

Без хорошей документации никто к вам на плечи не встанет, каким бы вы ни были гигантом.

## LXC

Виртуализуемся по способу BSD и Solaris, пользуясь Linux-контейнерами. Страшно – аж жуть; но мы все-таки рискуем.

Содержание последующих выпусков может меняться без уведомления – вдруг мы заблудимся, летя домой с OSCON!  
Да тут еще нарушение суточного ритма...

# LINUX FORMAT

Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-21973 от 14 сентября 2005 года. Выходит ежемесячно. Тираж 5000 экз.

### РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

#### Главный редактор

Кирилл Степанов [info@linuxformat.ru](mailto:info@linuxformat.ru)

#### Литературный и выпускающий редактор

Елена Толстякова

#### Переводчики

Елена Ессяк, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова

#### Редактор диска

Кирилл Степанов

#### Верстка, допечатная подготовка

Сергей Рогожников

#### Технический директор

Денис Филиппов

#### Директор по рекламе

Владимир Савельев [advert@linuxformat.ru](mailto:advert@linuxformat.ru)

#### Генеральный директор

Павел Фролов

#### Учредители

Частные лица

#### Издатели

Виктор Федосеев, Павел Фролов

#### Отпечатано в ООО «Ланинь»

188330, Ленинградская обл., Гатчинский р-н, пос. Сиверский, Вокзальная ул., 4  
Заказ 6011

### РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Грэм Моррисон [Graham Morrison] [graham\\_morrison@futurenet.com](mailto:graham_morrison@futurenet.com)

Заместитель редактора Эндрю Грегори [Andrew Gregory]

[agregory@futurenet.com](mailto:agregory@futurenet.com)

Штатный автор Джонатан Робертс [Jonathan Roberts]

[jonathan.roberts@futurenet.com](mailto:jonathan.roberts@futurenet.com)

Редактор диска Бен Эверард [Ben Everard] [ben\\_everard@futurenet.com](mailto:ben_everard@futurenet.com)

Художественный редактор Эфраин Эрнандес-Мендоза

[Efrain Hernandez-Mendoza] [efrain.hernandez-mendoza@futurenet.com](mailto:efrain.hernandez-mendoza@futurenet.com)

Выпускающий редактор Гэри Уокер [Gary Walker] [gary.walker@futurenet.com](mailto:gary.walker@futurenet.com)

### ПОДГОТОВКА МАТЕРИАЛОВ

Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Бен Эверард [Ben Everard], Грэм Моррисон [Graham Morrison], Джонатан Робертс [Jonathan Roberts], Шашанк Шарма [Shashank Sharma], Майк Сондерс [Mike Saunders], Ник Вейч [Nick Veitch], Евгений Балдин, Артем Зорин, Кирилл Степанов, Юрий Татарский, Андрей Ушаков, Алексей Федорчук, Игорь Штомпель

Художественный ассистент Кейт Мак-Доннелл [Kate McDonnell]

Иллюстрации Шейн Коллиндж [Shane Collinge], Ely Walton Illustrations, iStockPhoto, Саймон Мидлвик [Simon Middleweek]

### КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Tel. +44 01225 442244 Email: [linuxformat@futurenet.com](mailto:linuxformat@futurenet.com)

#### РОССИЯ:

##### Санкт-Петербург (редакция):

Лиговский пр., 50, корп. 15

Тел. +7 (812) 309-06-86

##### Представительство в Москве:

Красноказарменная ул., 17, м. «Авиамоторная» (в помещении АТС МЭИ)

Тел./факс +7 (499) 271-49-54

По вопросам сотрудничества, партнерства, оптовых закупок:

[partner@linuxcenter.ru](mailto:partner@linuxcenter.ru)

**Авторские права:** Статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на CD или DVD-диски, поставляемые вместе с журналом, если не было оговорено иное.

**Ограничение ответственности:** используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за 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](http://www.gnu.org/copyleft/gpl.html)

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт <http://www.futurepic.com>



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO



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.
<b>Корпоративные продукты Mandriva</b>	
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
<b>Продукты Mandriva для образования</b>	
Комплект программного обеспечения Mandriva Linux и EduMandriva для школ	3 500
<b>Сертифицированные ФСТЭК продукты Mandriva</b>	
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 ИСПОЛЬЗУЮТ:**  
 МВД РФ, Минздравсоцразвития РФ,  
 Минфин р. 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](http://www.linuxcenter.ru)



# ЦЕНЫ В СВОБОДНОМ ПАДЕНИИ

HETZNER DEDICATED  
ROOT SERVER **EX 4**

~~5500~~ руб.

**1900** руб.

УСТАНОВКА

HETZNER DEDICATED  
ROOT SERVER **EX 4S**

~~5500~~ руб.

**2200** руб.

УСТАНОВКА

HETZNER DEDICATED  
ROOT SERVER **EX 6**

~~5500~~ руб.

**2900** руб.

УСТАНОВКА

HETZNER DEDICATED  
ROOT SERVER **EX 5**

~~1900~~ руб.

**0** руб.

УСТАНОВКА

## HETZNER DEDICATED ROOT SERVER **EX 4**

- Intel®Core™ i7-2600 Quadcore с поддержкой технологии Hyper-Threading
- 16 ГБ DDR3 RAM
- 2 x 3 ТБ SATA 6 Гбит/с 7200 об/м! (Software-RAID 1)
- Операционная система Linux
- Неограниченный трафик\*
- IPv6 подсеть (/64)
- Установка из образов
- Без минимального контракта
- Установка 1900 рублей

**1900**  
рублей в месяц

## HETZNER DEDICATED ROOT SERVER **EX 5**

- Intel®Core™ i7-920 Quadcore с поддержкой технологии Hyper-Threading
- 24 ГБ DDR3 RAM
- 2 x 750 ГБ SATA 3 Гбит/с (Software-RAID 1)
- Операционная система Linux
- Неограниченный трафик\*
- IPv6 подсеть (/64)
- Установка из образов
- Без минимального контракта
- Установка 0 рублей

**2500**  
рублей в месяц



GreenIT  
Best Practice Award

2011

Hetzner Online активно поддерживает защиту окружающей среды, используя исключительно 100% возобновляемые источники энергии. Выберите более чистое будущее вместе с Hetzner Online!



**RU.HETZNER.COM**

\*Трафик предоставляется бесплатно. При превышении порога 10000 ГБ/месяц скорость соединения будет ограничена до 10 MBit/s. Стоимость аренды постоянного канала с пропускной способностью 100 MBit/s составляет 290 рублей за каждый дополнительный ТВ.