

Колонка главного редактора



Осень только началась, а Google уже смогла крепко привлечь к себе внимание не только отраслевых СМИ, но и всех остальных, кто еще не успел окончательно замкнуться в своем ограниченном мире информации, а периодически выходит в свет, чтобы посмотреть, чем живут люди. А живут они теперь новым, формально «открытым» веб-браузером, который с легкостью и за считанные дни потеснит (и уже потеснил) серьезных игроков этого рынка, — Google Chrome. Разработка интернет-гиганта, про которую давно ходили неуверенные слухи, буквально ворвалась в квартиры гиков, продвинутых пользователей и прочих энтузиастов, любящих все новенькое и уж тем более — «трендовое».

Что это означает для индустрии? Лишь в очередной раз недвусмысленно указывает на растущие аппетиты компании, которой многие не первый год пророчат статус главного конкурента Microsoft в обозримом будущем. Что это означает для Open Source? Спорно: проект Chrome еще слабо коррелирует с FOSS-движением и больше выглядит как угроза популярности Firefox, которая была с таким трудом сформирована тысячами энтузиастов со всего мира (к слову, в противовес «внутрикорпоративной» разработке Google, пусть и использующей открытые компоненты). Плюсы и минусы очевидны, а вот результат — пока неясен.

Главный редактор
Дмитрий Шурупов
(osa@samag.ru)

«Open Source»
электронное приложение к журналу
«Системный администратор»
№31, 16 сентября 2008 г.

РЕДАКЦИЯ

Исполнительный директор

Владимир Положевец

Главный редактор

Дмитрий Шурупов

Верстка и оформление

Владимир Лукин

Сайт электронного приложения:

<http://osa.samag.ru>

За содержание статьи ответственность несет автор. Все права на опубликованные материалы защищены.

Новости мира Open Source

Intel покупает Linux-разработчика OpenedHand

Intel объявила о том, что покупает британскую компанию OpenedHand, известную Open Source-миру в качестве разработчика дистрибутива Poky Linux и оконного менеджера Matchbox.

OpenedHand специализируется на программных продуктах для мобильных устройств: ее дистрибутив Poky Linux с интерфейсом X11/GTK+ ориентирован на телефоны, а оконный менеджер Matchbox нашел свое применение, например, в Linux-платформе Maemo для интернет-планшетов Nokia. OpenedHand активно сотрудничала с проектом популярной графической среды GNOME, в основе которой лежит применяемая в разработках компании библиотека GTK+.

Теперь разработчики OpenedHand присоединятся к коллективу технологического центра программного обеспечения с открытым кодом Intel (Open Source Technology Center) и сосредоточат свои усилия на совершенствовании программной Linux-платформы Moblin, предназначенной для мобильных интернет-устройств (MID) вроде нетбуков.

Состоялся долгожданный релиз Django 1.0

3 сентября состоялся первый «крупный» релиз популярного фреймворка с открытым кодом, написанного на Python, — Django 1.0. Как отметили разработчики в пресс-релизе, выпуск Django 1.0 — это очень значимое событие, для достижения которого потребовалось три года разработок. Только с момента последнего релиза (0.96) было сделано более 4000 тысяч коммитов кода, исправлено более 2000 проблем, исправлено/добавлено/удалено около 350 тысяч строк кода, добавлено 40 тысяч строк документации. Если же рассматривать всю историю разработки Django, то релиз 1.0 стал 8961 ревизией публичного репозитория.

Среди новостей в релизе Django 1.0 выделяется: переработанный административный интерфейс, улучшенная обработка Unicode, улучшенный ORM, автоматическое преобразование переменных в шаблонах во избежание XSS-атак, поддержка GIS, поддержка подключаемых базендов для хранения файлов (для FileField и ImageField), совместимость с Jython (реализация Python на Java), обновление встроенной системы комментариев, устранение устаревших возможностей.

У Ubuntu Linux — более 8 миллионов пользователей

Компания Canonical, занимающаяся поддержкой популярного Linux-дистрибутива Ubuntu, заявила, что у ее системы уже более 8 миллионов пользователей.

Интересно, что в середине 2007 года Марк Шаттлворт (Mark Shuttleworth), основатель Ubuntu и Canonical, оценивал число пользователей системы в 6-12 миллионов. И вот теперь в рекламном пресс-релизе Canonical сообщается, что «во всем мире Ubuntu используют более 8 миллионов человек».

Последний всплеск популярности Ubuntu Linux связан с ее широким распространением в качестве предварительной установленной операционной системы на мобильных интернет-устройствах (MID) — нетбуках. Очевидно, оценки числа пользователей Ubuntu довольно относительно и не могут учитывать все особенности свободного распространения дистрибутива между Linux-энтузиастами и их знакомыми.

Red Hat купила компанию Qumranet

Американская Linux-компания Red Hat объявила о покупке Qumranet, известной своими решениями в области виртуализации.

Прославившие Qumranet продукты — это вошедшая в состав Linux-ядра инфраструктура виртуализации KVM (Kernel-based Virtual Machine) и VDI-решение SolidICE (Independent Computing Environment), обеспечивающее виртуализацию для десктопов. Как сообщается в пресс-релизе Red Hat, новое приобретение компании позволит ей предоставить рынку многофункциональные решения, интегрированные с операционной системой, чего не могут сделать поставщики, занимающиеся исключительно продуктами виртуализации.

По соглашению между компаниями Red Hat потратила около 107 миллионов USD на Qumranet. В компании ожидают, что сделка принесет до 20 миллионов USD в следующем финансовом году, который начнется 1 марта 2009 года.

Sun запустила бета-версию хостинга для Open Source-проектов

Компания Sun Microsystems, не делая публичных анонсов, запустила бета-версию нового сервиса для хостинга Open Source-проектов — Project Kenai (<http://kenai.com>).

Тим Брэй (Tim Bray), директор по веб-технологиям в Sun, представил Project Kenai в своем блоге. По его словам, веб-интерфейс нового Open Source-хостинга построен на популярном открытом фреймворке Ruby on Rails. Кроме того, Брэй отметил, что проект Kenai интересен тем, что в него вошли возможности социальных сетей. Как пояснил Ник Сейгер (Nick Seiger), разработчик сервиса, одной из причин, побудивших к созданию Kenai, стала потребность в демонстрации возможностей создания проектов, работающих поверх традиционных стеков LAMP/SAMP. В этом заявлении автор ссылается на разработки Glassfish, JRuby и Ruby on Rails.

Пока Project Kenai находится в стадии бета-тестирования, для регистрации на нем новых проектов требуются приглашения. Ожидается, что этот хостинг станет центром для 750 Open Source-проектов Sun.

У Linux упала доля на рынке смартфонов: с 10% до 7%

Исследователи из компании Gartner опубликовали итоги очередного исследования рынка смартфонов. За последний год общий рост продаж смартфонов составил 15,7%, однако доля Linux значительно уменьшилась.

Общие продажи смартфонов во втором квартале 2008 года превысили 32,22 миллиона (по сравнению с 27,85 по состоянию на второй квартал 2007-го). При этом продажи Linux-устройств упали как в количественном, так и в процентном соотношении. Во втором квартале 2008 года

доля Linux на рынке смартфонов составила 7,3% (2,36 млн. устройств) против 10,1% (2,82 млн.) в прошлом году.

Наибольшего роста за тот же период добилась платформа Research In Motion (смартфоны BlackBerry), захватив 17,4% рынка по сравнению с 8,9% в 2007 году. Платформа Microsoft, Windows Mobile, также сумела немного улучшить свое присутствие на этом рынке: доля Windows-устройств выросла с 11,5% до 12%.

В немецком университете откроется кафедра по Open Source

На техническом факультете Университета Эрланген-Нюрнберг (Германия) открылась вакансия профессора по программному обеспечению с открытым кодом (Open Source). Ожидается, что отобранный кандидат возглавит кафедру по Open Source в январе 2009 года.

В вакансии сообщается, что кандидат будет заниматься как преподаванием, так и исследованиями в области Open Source. При этом он должен обладать опытом работы в крупном проекте ПО с открытым кодом и иметь отличную научную квалификацию в этой области. Кандидат должен иметь опыт исследований хотя бы в одном из следующих направлений: 1) управление технологическими приемами, качеством и командами в контексте Open Source Software; 2) технологические приемы и утилиты разработки для распределенной работы над программным обеспечением; 3) важнейшие характеристики успешных Open Source-архитектур. Также особый интерес представляют вопросы

применения Open Source в бизнесе и легальные аспекты.

Sun представила Open Source-проект виртуализации xVM Server

Sun Microsystems представила свой новый Open Source-проект, нацеленный на улучшение возможностей виртуализации Sun xVM Server, – xVMserver.org. Анонсированный 10 сентября xVMserver.org – это инициатива Sun по созданию нового Open Source-сообщества на базе ее виртуализационного программного обеспечения Sun xVM Server. В пресс-релизе сообщается, что на появившемся веб-сайте разработчики могут скачать первый архив исходного кода Sun xVM Server и внести свой вклад в процесс разработки продукта. На сайте же пока можно найти лишь единственный архив «xVM Server Early Access Open Source Code Bundle», а остальные пакеты (архивы с бинарными версиями xVM Server 1.0 для разных платформ, его исходным кодом и SDK) пока недоступны.

Объявление о запуске xVMserver.org состоялось одновременно с сообщением о доступности программного обеспечения Sun xVM Server и Sun xVM Ops Center 2.0, которые в Sun называют «ключевыми компонентами в ее всеобъемлющей стратегии по виртуализации». А незадолго до этого, в начале месяца, Sun объявила о релизе 2.0 ее другого Open Source-продукта виртуализации – VirtualBox.

*Дмитрий Шурупов,
по материалам www.nixp.ru
(osa@samag.ru)*

Sun Microsystems и Open Source: братья навек?

Компания Sun Microsystems известна на рынке примерно с 1982 года и рекомендовала себя как поставщик рабочих станций и высокопроизводительных серверов, с подходом 4Ss: Servers, Software, Storage, Services (серверы, программное обеспечение, хранилища данных, сервисы).

Как и в случае со многими другими корпорациями, занятыми в области IT, движение Open Source и деятельность Free Software Foundation не могли пройти мимо бизнес-модели Sun. Пример тому –

жесткая конкуренция Solaris и GNU/Linux, которая побудила Sun открыть ряд флагманских проектов и начать поддерживать свободное программное обеспечение.

В этой статье я расскажу о немалом вкладе Sun Microsystems в OSS и постараюсь сделать это беспристрастно.

Технология Java

Язык программирования и платформа Java (<http://www.java.com/ru>) были выпущены компанией Sun Microsystems в 1995 году. Главными достоинствами но-

вой платформы стали легкая переносимость и удобная разработка кроссплатформенных приложений, что было достигнуто благодаря созданию виртуальной машины, выполняющей байт-код, в который особым образом компилировался исходный код (подробную статью о «сортах» Java можно найти в «Open Source» 029).

Идея Java оказалась настолько хороша, что технология практически сразу приобрела огромную популярность и получила широкое распространение не только в серверной области, но и на обычных настольных компьютерах. Всё шло хорошо до тех пор, пока позиции Java не начали вытесняться бурно развивающейся платформой Microsoft .NET. Одним из шагов, сделанных Sun в конкурентной борьбе, стало открытие своей платформы и привлечение Open Source-сообщества к проекту OpenJDK (<http://openjdk.java.net>).

OpenOffice.org

В последнее время офисный пакет OpenOffice.org (<http://openoffice.org>) завоевал заметную популярность среди пользователей разных категорий. Несмотря на то, что он уступает известному Microsoft Office по функциональности, у него есть и заметное преимущество – он бесплатен и открыт. Изначально же существовал пакет StarOffice, разрабатываемый немецкой компанией StarDivision с 1994 года. В 2000 году Sun Microsystems купила ее и открыла исходники StarOffice: сначала не целиком, но в дальнейшем – исключительно под лицензией LGPL.

После открытия кода по инициативе Sun образовался проект OpenOffice.org, разработкой и поддержкой которого занимается сообщество. Благодаря этому он получил широкое распространение в Linux-дистрибутивах, и даже заменил на некоторых компьютерах Microsoft Office под управлением Windows.

Тем временем Sun продолжают развивать и StarOffice, взяв на вооружение проверенный путь создания коммерческого продукта на базе открытого. Продаваемый StarOffice отличается от OpenOffice.org наличием техподдержки, улучшенной документации, а также наличием «из коробки» ряда сторонних разработок: шрифтов, графики (clipart), мастеров переноса параметров из других офисных пакетов и прочих подобных вещей.

Пакету OpenOffice.org придает значимости тот факт, что его Writer стал первым текстовым процессором, поддерживающим принятый в ISO стандарт на документы – ODF (OpenDocument Format).

Резюмируя, можно сказать, что именно благодаря Sun появился свободный офисный пакет, который может удовлетворить практически любого «среднего» пользователя компьютера.

OpenSolaris

Операционная система Solaris была представлена Sun в 1992 году, а ее предок, SunOS, появился еще в начале 80-х. В дальнейшем Solaris получила мировое признание за хорошую масштабируемость (особенно на процессорах SPARC), высокую стабильность и ряд новшеств, таких как, например, DTrace и ZFS.

Шло время, и во многом благодаря таким компаниям, как Red Hat и Novell, стала набирать обороты операционная система GNU/Linux, которая начала постепенно вытеснять Solaris с рынка серверных решений. Сначала в Sun пытались традиционно развивать Solaris, наращивая функциональные возможности системы и расширяя поддержку оборудования, однако

Common Development and Distribution License (CDDL)

CDDL (Common Development and Distribution License) – это лицензия, по которой распространяются многие Open Source-проекты, спонсируемые Sun Microsystems. Она основана на лицензии для программного обеспечения от Mozilla Foundation – MPL (Mozilla Public License) – и является несовместимой с GNU

со временем компания была вынуждена вновь обратиться к Open Source.

Sun начала планировать проект OpenSolaris в 2004 году, а в 2005-м официально приступила к работе (http://blogs.sun.com/marchamilton/entry/opensolaris_launch). Как известно, OpenSolaris распространяется по лицензии CDDL и по сей день активно развивается сообществом при поддержке Sun Microsystems, а на его основе создаются новые версии Solaris.

VirtualBox

Программный продукт VirtualBox (<http://virtualbox.org>) первоначально разрабатывался немецкой компанией innotek и являлся мощным средством для виртуализации с поддержкой архитектуры x86. Он распространялся в двух версиях: OSE (Open Source Edition) и коммерческая (бесплатная для личного использования). В феврале 2008 года компания Sun объявила в своём пресс-релизе (<http://www.sun.com/aboutsun/pr/2008-02/sunflash.20080212.1.xml>) о приобретении innotek (это событие произошло на следующий день после выхода VirtualBox для OpenSolaris).

Модель распространения VirtualBox не изменилась, зато началось развитие функциональности и более тщательное «вылизывание» кода продукта. С тех пор, как innotek стали частью Sun Microsystems, VirtualBox действительно стал более мощным и стабильным – благодаря множеству изменений, которым подвергся весь проект, начиная от ядра и заканчивая графическим интерфейсом: появилась поддержка Solaris и Mac OS X, API для веб-сервисов, поддержка эмуляции SATA, экспериментальная поддержка компьютеров с более чем 4 Гб оперативной памяти... Вот и в дни написания статьи состоялся важный релиз VirtualBox – 2.0 (<http://virtualbox.org/wiki/Changelog>), главными особенностями которого стали поддержка 64-битных «гостевых» систем (при использовании 64-битной системы), а также долгожданный переход графического интерфейса на Qt 4.

Можно заключить, что VirtualBox стал действительно производительным, ста-

билным и удобным в использовании средством виртуализации. VirtualBox OSE распространяется по лицензии GNU GPL и является полностью свободным программным обеспечением.

Несмотря на все утверждения в Sun, нельзя не отрицать той возможности, что в компании пошли на такой шаг с тем, чтобы их ключевыми Open Source-технологиями не могли воспользоваться разработчики, например, ядра Linux.

MySQL

Популярнейшая свободная система управления базами данных MySQL (<http://www.mysql.com>) была создана компанией MySQL AB, а первый «рабочий» выпуск состоялся в мае 1995 года. Сейчас же стабильной версией является MySQL 5.0, которая может похвастаться поддержкой стандарта SQL 2003, хранимых процедур и функций, Unicode. Своей огромной популярностью СУБД MySQL во многом обязана проекту PHP: связка LAMP (Linux, Apache, MySQL, PHP) стала практически стандартом «де-факто» в среде Web. (Прим. ред.: И даже несмотря на то, что в последнее время PHP постепенно теряет популярность, на используемости MySQL это сказывается мало.) У MySQL принята политика двойного лицензирования, а с октября 2006 года компания выпускает свой ключевой продукт в двух редакциях: Community Edition, распространяемая под GNU GPL, и Enterprise Edition, для корпоративных пользователей, которым важна техническая поддержка, а также возможность скрыть код в своих разработках, что напрямую противоречит GPL.

Sun Microsystems приобрели MySQL AB в феврале 2008 года с заявленной в пресс-релизе (<http://www.mysql.com/news-and-events/sun-to-acquire-mysql.html>) стоимостью сделки в 1 миллиард USD. Модель распространения СУБД MySQL не изменилась. Однако в Интернете начали появляться слухи о том, что MySQL станет закрытой, а её разработчиков постепенно уволят. Причём эти домыслы многократно усилились после ухода из Sun Михаэля Видениуса (Michael Widenius), разработчика MySQL Server. Но эти слухи опроверг Кай Арно (Kaj Arno), занимающийся связями с сообществом Open Source: во-первых, отставка Майкла была возможна ещё до покупки Sun, а во-вторых, компания обещает продолжать

развитие СУБД на должном уровне, поэтому MySQL не станет хуже (<http://blogs.mysql.com/kaj/2008/09/05/the-rumours-on-monty-resigning>).

NetBeans IDE

Интегрированная среда разработки NetBeans (<http://www.netbeans.org>) была создана при Карловом Университете в Праге (Чехия) и называлась Xelfi IDE for Java. Затем разработка была перенесена в одноимённую компанию, которая занималась поддержкой и продажей этой среды, но впоследствии, в 1999 году, была целиком поглощена Sun Microsystems.

Sun открыла код NetBeans в июне 2000 года и начала спонсировать развитие проекта, вокруг которого постепенно формировалось сообщество. Прошли годы, и сейчас среда NetBeans, распространяющаяся по лицензии CDDL, способна конкурировать с таким «монстрами», как Eclipse IDE и IntelliJ IDEA, не уступая им ни по качеству, ни по возможностям. В NetBeans полностью реализованы рефакторинг, профилирование, автодополнение, причём поддерживается не только язык Java, но и C/C++, Ruby и многие другие языки программирования.

От себя замечу, что NetBeans является моей любимой интегрированной средой разработки.

ZFS

Файловая система ZFS (ранее – Zettabyte File System) (<http://opensolaris.org/os/community/zfs>) впервые была показана широкой публике в ноябре 2005 года в составе системы OpenSolaris. Основной целью её разработки была замена файловой системы UFS, которая уже не соответствовала современным требованиям к хранению данных. ZFS – это 128-битная файловая система, позволяющая хранить 248 файлов в одной ФС, а максимальный её размер может достигать 16 эксабайт (то есть 264 байт). ZFS обладает внушительным спектром возможностей, среди которых – поддержка прозрачного сжатия, POSIX-атрибуты для файлов и шифрование.

При этом ZFS является Open Source-проектом, однако у многих сразу возникает вопрос: «Почему же тогда в ядре Linux нет толковой поддержки ZFS, раз эта файловая система настолько хороша?». Ответ крайне прост: ядро Linux распространяется по лицензии GPL, ZFS – по CDDL, а эти лицензии несовместимы. Использование исходников ZFS в Linux-ядре будет нарушением CDDL, а следовательно, портирование кода Sun в ядро Linux просито незаконно.

В Linux-сообществе существует убеждение, что Sun сделала это намеренно – для того, чтобы не дать системе GNU/Linux использовать её открытые разработки. (Прим. ред.: Тем временем сторонники лицензии BSD с ухмылкой смотрят на происходящее: ZFS вошла в состав ОС FreeBSD еще в апреле 2006 года. Подробности – на <http://wiki.freebsd.org/ZFS>.) Существует адаптация ZFS под Linux с использованием FUSE (Filesystem in Userspace), которая основывается на выполнении кода драйвера файловой системы в пользовательском пространстве (userspace). Однако такая реализация вряд ли заслуживает доверия для использования на промышленном сервере.

Open High Availability Cluster

Открытый кластер с высокой доступностью (Open High Availability Cluster – ОНАС) является мощным Open Source-средством для кластеризации, разрабатываемым при поддержке Sun Microsystems. Однако Open HA Cluster не предоставляет продукт для конечных пользователей или полноценное ПО для распределения нагрузки – это, по сути, фреймворк для построения кластерных систем. На основе ОНАС строится коммерческий проект Sun Cluster, входящий в решение Solaris Cluster. Нетрудно догадаться, что Open HA Cluster, как и все разработки Sun, имеющие отношение к системе Solaris, распространяется по лицензии CDDL.

Java Desktop System

Sun Java Desktop System (<http://www.sun.com/software/javadesktopsystem>) – рабочая среда для десктопов (преимущественно – под управлением Solaris, поскольку поддержка GNU/Linux прекращена с середины 2006 года, <http://www.sun.com/software/javadesktopsystem/eol.xml>). Она разработана на основе GNOME и ориентирована на «среднего» пользователя, привыкшего к Windows, а распространяется по лицензии GPL. В состав Java Desktop System включены такие известные в Open Source-сообществе приложения, как StarOffice, Mozilla Suite, Evolution, Gaim (ныне Pidgin). Особого внимания заслуживает официальная тема оформления – Nimbus, которая была разработана Sun Microsystems специально для JDS.

Вообще актуальность JDS можно ставить под большой вопрос. Тот факт, что она основана на GNOME и позиционируется как знакомое для Windows-пользователя окружение, а также актуальность ее версии дают многие достаточные основания считать, что Java Desktop System – рудимент.

Open Firmware

Open Firmware – стандарт, описывающий интерфейсы программ, зашитых в ПЗУ компьютера, и продвигаемый организацией IEEE. Он был создан Sun Microsystems и в настоящее время используется Sun, Apple, IBM и многими другими производителями чипсетов, отличных от x86. Стандарт позволяет системе загружать независимые от платформы драйверы напрямую с платы PCI – для увеличения совместимости. Доступ к Open Firmware происходит через интерфейс на языке Forth. Независимость от платформы достигается благодаря компилированию исходников драйвера в промежуточный код FCode, подобно Java.

Некоторые коммерческие реализации Open Firmware были открыты в 2006 году, включая Sun OpenBoot, Firmworks OpenFirmware и Codegen SmartFirmware. Исходный код доступен под лицензией GPL благодаря проекту OpenBIOS, а реализация от Sun – OpenBoot – распространяется по лицензии BSD.

UltraSPARC T1

UltraSPARC T1 – многоядерный микропроцессор с поддержкой многопоточности, разработанный Sun Microsystems. Также известен как «Niagara». Интересен тем, что его спецификация полностью открыта по лицензии GNU GPL и находится в свободном доступе (<http://www.opensparc.net/opensparc-t2/index.html>).

Братья ли?

Sun Microsystems внесла немалый вклад в движение за свободное программное обеспечение, создала многие достойные продукты – как программные, так и аппаратные. Sun является «патроном» Free Software Foundation, постоянно спонсирует перспективные и инновационные Open Source-проекты, которые вырастают в полноценные решения промышленного уровня (не стоит забывать, что Sun в первую очередь ориентируется на enterprise-рынок)... Но всё ли так просто?

Ни одна коммерческая организация не станет работать себе в убыток, поэтому каждое движение Sun (равно как и других компаний) в сторону программного обеспечения с открытым кодом – тщательно обдуманный шаг. И каждый такой шаг, помимо «благотворительности», решает и другие, более насущные, проблемы: получение максимальной выгоды, реклама или же просто спасение от банкротства.

Но не стоит и односторонне относиться к Open Source от Sun Microsystems. С одной стороны, Sun стремятся максималь-

но расширить сферу своего влияния в IT, получить максимальную прибыль и «выжать» деньги буквально из всего, что можно продать, причём не обращая внимания на чистоту своих методов и порой даже на последствия своих действий для всего

сообщества. А с другой стороны, открытое программное обеспечение, спонсируемое Sun, начинает очень быстро развиваться, и многим компаниям, предлагающим аналогичные, но коммерческие решения, приходится «потесниться»

и работать лучше. Возникает конкуренция. А конкуренция – это всегда полезно для потребителей.

Дмитрий Усталов
(eveel@gmail.com)

FOSS Review 007

Количество программ – не бесконечно. И кажется, что когда-нибудь наступит момент, когда все интересные приложения будут описаны на страницах этой рубрики. Но пока этого не случилось. А учитывая тот факт, что каждый день появляются новые и новые проекты, можно не сомневаться, что в каждом выпуске «Open Source» будут новые обзоры, насколько бы противоречивым это ни казалось.

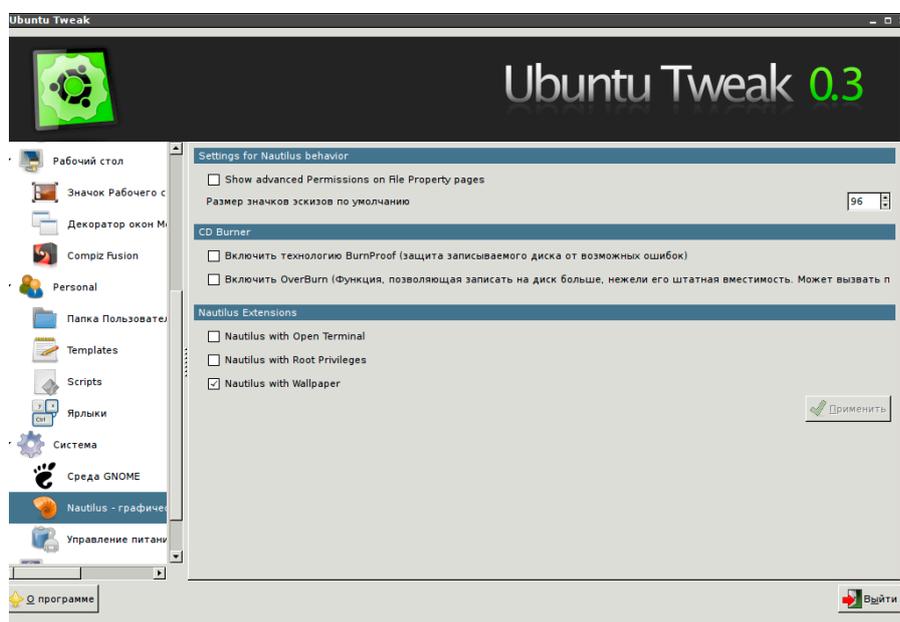
UbuntuTweak

- ✓ Версия: 0.3.5.
- ✓ Лицензия: GNU GPL.
- ✓ Размер: 508 Кб (deb).
- ✓ Сайт: <http://ubuntu-tweak.com>.

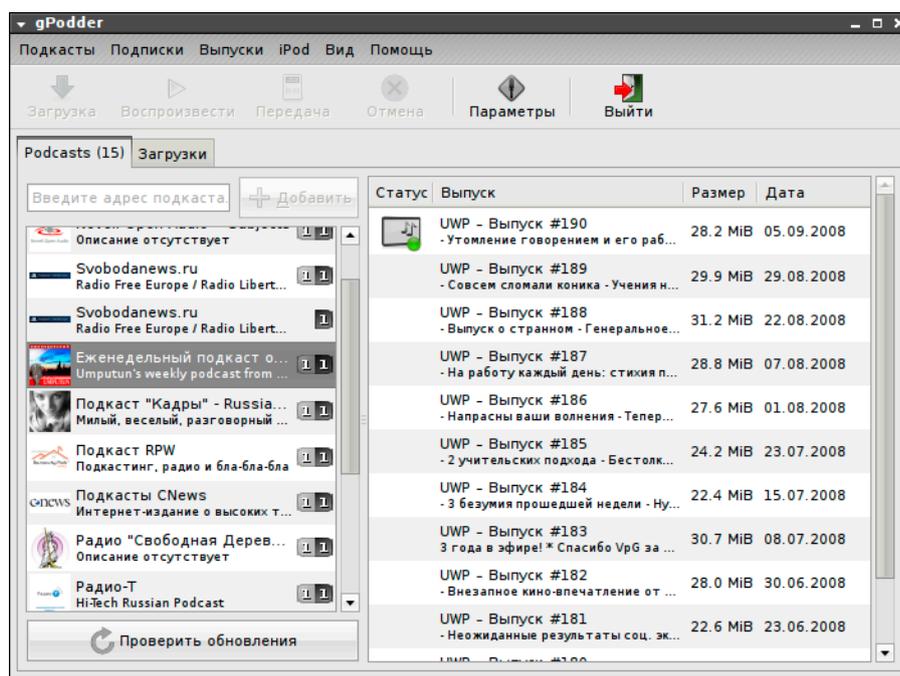
Есть такая проприетарная операционная система, для которой написано очень много программ, изменяющих функциональность и внешний вид компонентов системы. Например, с помощью таких утилит облегчается задача убирирования стрелок с ярлыков или уменьшения времени задержки перед появлением меню. Все они так или иначе меняют ключи реестра и файлы конфигурации. Не каждому пользователю хватит знаний и умений, чтобы самостоятельно отыскать и изменить нужный ключ в реестре или строчку в файле. В Linux же такими программами пользоваться было как-то не принято, но шло время и с появлением дружелюбного простому пользователю дистрибутива Ubuntu, а следовательно – и менее продвинутых пользователей, возник спрос на подобный софт. В основе Ubuntu лежит рабочее окружение GNOME, система настроек которого, Gconf, представляет собой аналог реестра. Некоторые переменные откуда не описаны в настройках GNOME, и их можно изменить только руками с помощью специального редактора (Прим.ред.: gconf-editor). UbuntuTweak – приятная глазу программа, которая легко и просто предоставляет доступ к скрытым настройкам системы. Возможности программы удобно разбиты по категориям. Во вкладке «Приложения» можно найти установку и удаление программ, которые не входят в офи-

циальный репозиторий Ubuntu (например, Google gadgets). Пункт «Автозапуск» в комментариях не нуждается. Категория «Рабочий стол» таит в себе настройки Metacity, Compiz Fusion, а также управление значками рабочего стола. Любопытна вкладка «Персональные», где можно изменить

назначения стандартных каталогов пользователя, управлять шаблонами документов, скриптами Nautilus и клавиатурными сочетаниями. Следующий пункт – «Система», где доступны настройки GNOME, Nautilus и gnome-power-manager. А вот «Безопасность» будет интересна разве что системным администраторам, поскольку в ней можно поставить запрет на некоторые функции рабочего окружения GNOME



UbuntuTweak



Gpodder

Электронное приложение «Open Source»

(например, отключить печать или диалог запуска приложений). Итак, UbuntuTweak – очень полезная программа для начинающих пользователей Ubuntu или тех, кто не хочет копаться в конфигурационных файлах и системе confd.

Gpodder

- ✓ **Версия:** 0.12.3.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 316 Кб (tar.gz).
- ✓ **Сайт:** <http://gpodder.berlios.de>.

Подкастинг – процесс записи и распространения посредством RSS, как правило, тематических аудио- или видеопередач. Подкасты публикуют на специальных сайтах – подкаст-терминалах (например, www.rpod.ru и www.podfm.ru). Подкаст-ленты же напоминают блоги. Популярность подкастов была обусловлена плеерами iPod, где впервые появилась их поддержка. Gpodder – программа, которой я пользуюсь практически со времени ее появления. Она представляет собой подкаст-клиента, т.е. приложение для скачивания аудио- и видеопередач. Gpodder написан на PyGTK и наделен удобным, красивым интерфейсом. В левой части окна – список лент (с названиями, числом новых записей в ленте и картинкой), на которые вы подписаны, а справа показаны доступные для скачивания и прослушивания передачи. В колонках списка публикуются размер файла и дата его появления. При наведении мышью на выпуск отображается дополнительная информация (обычно это краткое содержание). Внизу слева – большая кнопка проверки обновлений. Gpodder поддерживает загрузку файлов по BitTorrent, но самая важная особенность программы – синхронизация с iPod и другими mp3-плеерами. Прослу-

шанные выпуски программа может удалять с диска автоматически. Любопытной покажется возможность передачи файлов по Bluetooth, что будет полезно, если вам больше нравится слушать подкасты на мобильном телефоне. Разработчики обещают делать новые релизы каждый месяц и пока обещания держат, причем зачастую даже опережая график.

GScrot

- ✓ **Версия:** 0.51.1.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 198 Кб (tar.gz).
- ✓ **Сайт:** <http://gscrot.ubuntu-projekte.de>.

Мало кому из нас не приходилось хотя бы однажды делать скриншоты. И риску предположить, что многие делают такие снимки гораздо чаще. GScrot предназначена как раз для создания скриншотов. Возможностей у программы по-настоящему много, но обо всем по порядку. GScrot умеет снимать выделенную область экрана, полный экран, конкретное окно (из списка можно выбрать, с какого из открытых окон делать скриншот) и делать снимок страницы в Интернете. Сохранить изображение можно либо в формате PNG, либо в JPEG. После того как снимок сделан, можно обработать полученное изображение – например, нарисовать на нем что-то или применить к нему один из плагинов, коих к текущему релизу насчиталось 11 штук. Плагины позволяют накладывать фильтры (черно-белый, сепия), рисовать рамки. Разработчики обещали в следующих релизах увеличить их число. Разумеется, поддерживается и печать снятого. GScrot может сворачиваться в область уведомлений. В настройках, помимо всего прочего, задаются собственные комбинации клавиш для снимка. Ко всем снятым изображениям можно получить доступ с помощью вкладок.

Last Exit

- ✓ **Версия:** 6.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 437 Кб.
- ✓ **Сайт:** <http://lastexit-player.org>.

Last.fm – замечательный, на мой взгляд, социальный сервис. Незаменим, когда хочется новой музыки, но нет желания искать что-либо и покупать/скачивать. У этой социальной сети есть свой собственный клиент, который основан на Qt и вообще до-



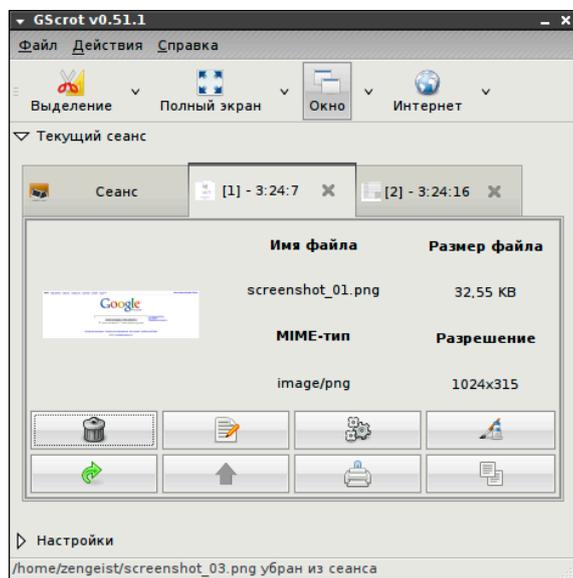
вольно громоздок. Last Exit написан с помощью Mono, имеет простой интерфейс, который, по заявлению разработчиков, был взят у широко известного в узких кругах музыкального плеера Muine. Окно программы выглядит весьма аскетично. Элементы управления можно мысленно разбить на три горизонтальные строки. В первой верхней – кнопки «играть/пауза», «переход к следующему треку», «нравится/не нравится», «управление громкостью» и кнопка, вызывающая маленькое окно настроек. Ниже – строка с выпадающим списком радиостанций (все стандартные станции и настройка своей), а еще ниже – информация о воспроизводимой в настоящий момент композиции. Last Exit – предельно простая программа, ничего лишнего.

BlueProximity

- ✓ **Версия:** 1.2.5.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 301 Кб (tar.gz).
- ✓ **Сайт:** <http://blueproximity.sourceforge.net>.

Представьте, что вы работаете в многолюдном офисе и вам вдруг необходимо куда-то отойти, а на экране компьютера – важная информация, открыта страница Одноклассников, да и вообще, вы просто не любите, когда в ваше отсутствие компьютером пользуется кто-то еще. Ноутбук, например, можно забрать с собой, а экран – заблокировать с помощью скринсейвера. Но есть программа BlueProximity, которая использует Bluetooth для определения расстояния между вами и компьютером, а по достижении критической отметки блокирует экран или выполняет любое другое (предварительно заданное) действие. Все очень просто. Bluetooth-устройство (например, мобильный телефон или гарнитура) лежит у вас в кармане. BlueProximity, визуально размещающаяся в системном трее, регулярно его опрашивает и измеряет расстояние между компьютером и устройством. Если вы оказываетесь за допустимым радиусом, экран блокируется, но как только попадаете в допустимую зону – блокировка автоматически снимается, так что можно продолжать работать.

Роман Комков
(r.komkov@gmail.com)



GScrot

Обзор Open Source-инструментов для разработки программного обеспечения. Часть 1

Первая статья – это быстрый обзор доступных Open Source-инструментов для решения всех возникающих при разработке ПО задач. В дальнейшем планируется продолжение статьи, где будут представлены более подробные обзоры тех или иных средств.

Прим ред.: Эта статья написана специалистом компании, занимающейся разработкой программного обеспечения. Она отражает взгляд профессионала на современный рынок Open Source-решений в этой области, который заинтересован в решении конкретных прикладных задач. Не стоит путать такой подход с тем, что проповедует среднестатистический Open Source-энтузиаст, и нужно различать эти взгляды. Как видно из материала, речь идет об использовании средств с открытым кодом на протяжении всего жизненного цикла разработки межплатформенного ПО, создаваемого на таких языках, как C++, C# и Java, при условии, что код пишется программистами, работающими под разными ОС.

В этом цикле статей я, как разработчик ряда программных продуктов, хочу поделиться опытом использования Open Source-инструментов при разработке программного обеспечения. Эти средства абсолютно легальны, общедоступны и в полной мере удовлетворяют большинство требований разработчиков. Хочу с самого начала обратить внимание читателей на то, что целью статьи не ставится срав-

нение коммерческих и Open Source-продуктов, так как это две разные ниши, отличающиеся в том числе и сервисом, который вам готов предоставить поставщик средств разработки.

Что можно, а что нельзя?

Именно с этого вопроса должен начинаться выбор инструментов разработки программного обеспечения, так как он определяет возможность использования инструмента. Почему так? Потому что от того, коммерческий продукт вы производите или открытый, будет зависеть, что можно использовать, а что – нельзя. Например, если вы собираетесь использовать свободные библиотеки, то должны обратить внимание на лицензию, под которой они распространяются. Так, например, лицензия GPL обязывает открывать исходные коды своего продукта под той же лицензией. Есть и более мягкие лицензии, которые позволяют не открывать свой код – к ним относятся BSD, Apache Public License, EPL (Eclipse Public License), LGPL (Lesser General Public License) и другие.

Также необходимо учесть ограничения использования инструментов, так как некоторые производители требуют их покупки в случае применения при разработке коммерческого продукта. Чаще всего такие продукты имеют две версии: одну – Community, а другую – Standart, Professional и т.д.

При выборе инструментов не забудьте определить, что можно для вашего проекта.

Стандартный набор приложений

После решения этого вопроса можно приступать к определению, какие базовые инструменты понадобятся. Для примера рассмотрим абстрактный проект среднего масштаба – разработка клиент-серверной системы управления каким-либо сетевым оборудованием. Что нам понадобится для этого, кроме программистов? Конечно же, инструменты, которые эти программисты будут использовать. Определим, что они будут делать:

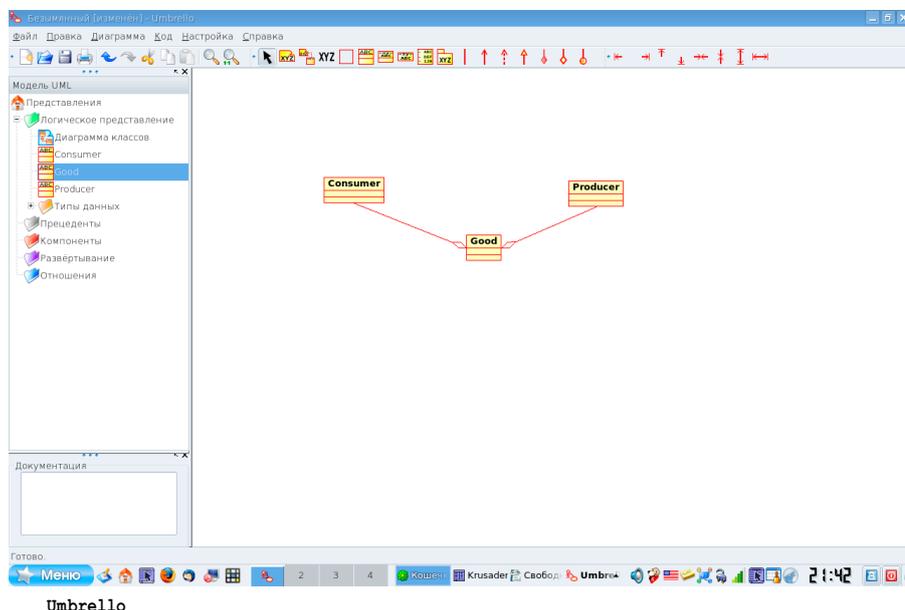
- Создавать проект программного продукта: набор пакетов и классов и их взаимодействие; взаимодействие с агентом/сервером и с пользователем.
- Писать код программы на каком-то языке.
- Тестировать код программы.
- Отлаживать код.
- Писать документацию на протестированный и отлаженный код.
- Хранить код в едином, общедоступном хранилище.
- Писать инструкцию по использованию программного продукта.
- Создавать пакет развертывания.
- Собирать информацию об ошибках в программном продукте.

Примерно из этих стадий состоит среднестатистический проект, который ставит целью поставку пользователю полезного и качественного продукта. Пора определять инструменты, которые будут использоваться.

Проект программного продукта

Проектом программного продукта (впрочем, как и любого проекта) называется временное мероприятие, в ходе которого реализуется определенное решение или продукт, имеющие четкие цели, критерии оценки их достижения, программу действий и ограниченное время.

Однако нас интересует проект с инженерной точки зрения, а это диаграммы классов, диаграммы взаимодействия, описания использования, описания входных и выходных данных, описание функционирования и прототипы графических форм (если используются). Традиционно для этой задачи используются язык UML, текстовый редактор и графический редактор. Первый – для описания классов, диаграмм взаимодействия и вариантов использования, второй – для описания



проекта в словесной форме, а третий – для оформления иллюстраций и прототипов графических форм.

UML-редактор

Если начать поиски UML-редактора в Google, то выяснится, что чуть ли не полсотни фирм и разработчиков предлагают свои продукты и называют их особенными. Некоторые из них действительно выдающиеся, такие как Telelogic Rhapsody или Rational. Но даже если отбросить все коммерческие продукты, остается еще много решений. Однако лишь единицы из них могут быть использованы в проектах, потому что, во-первых, работают без ошибок, а во-вторых, имеют весь необходимый набор типов диаграмм.

Вот те единицы, что я отобрал после длительного изучения разных программ:

- ✓ **NetBeans IDE** (<http://www.netbeans.org>) – в эту среду включен отличный UML-редактор, который успешно используется в одном из моих текущих проектов. Редактор – удобный, качественный, быстрый, несмотря на то что написан на Java. К сожалению, не умеет генерировать исходный код заголовочных файлов на языке C++, а только прототипы классов для Java.
- ✓ **ArgoUML** (<http://argouml.tigris.org>) – неплохой и устойчивый редактор, написанный на Java, но при этом также работает быстро. Позволяет создавать все виды диаграмм UML и даже проектировать структуры таблиц для баз данных. Минусом является не очень удобный интерфейс (впрочем, это субъективно).
- ✓ **StarUML** (<http://staruml.sourceforge.net>) – отличный UML-редактор для Windows, позволяющий создавать все виды диаграмм. Работает очень быстро,

стабильно, а интерфейс удобный. Минус в том, что проект не развивается с 2005 года.

- ✓ **Umbrello** (<http://uml.sourceforge.net>) – хороший UML-редактор, входящий в большинство дистрибутивов Linux. Надежный, с удобным интерфейсом, работает быстро и генерирует код для языков C++ и Java.

Таким образом, если вы создаете проект для нескольких операционных систем или разработчики работают в разных ОС, то UML-редакторы на Java будут для вас более предпочтительны.

К сожалению, у всех рассмотренных редакторов различаются и являются несовместимыми форматы файлов для хранения диаграмм – при выборе следует учесть этот немаловажный фактор.

Текстовый редактор

Нужен редактор, который является простым и удобным, позволяет встраивать иллюстрации. Дополнительным требованием является возможность генерировать документы PDF и PostScript. Таких редакторов набралось немного:

- ✓ **OpenOffice.org Writer** (<http://www.openoffice.org>) – многоплатформенный, мощный и удобный текстовый редактор, обладающий возможностью создания документов в формате PDF. Важным достоинством является поддержка множества форматов документов, в том числе и Microsoft Word.
- ✓ **AbiWord** (<http://www.abisource.com>) – мощный редактор, который работает как в среде Linux, так и в среде Windows. В рассматриваемом контексте так же хорош, как и OpenOffice Writer.

Графический редактор

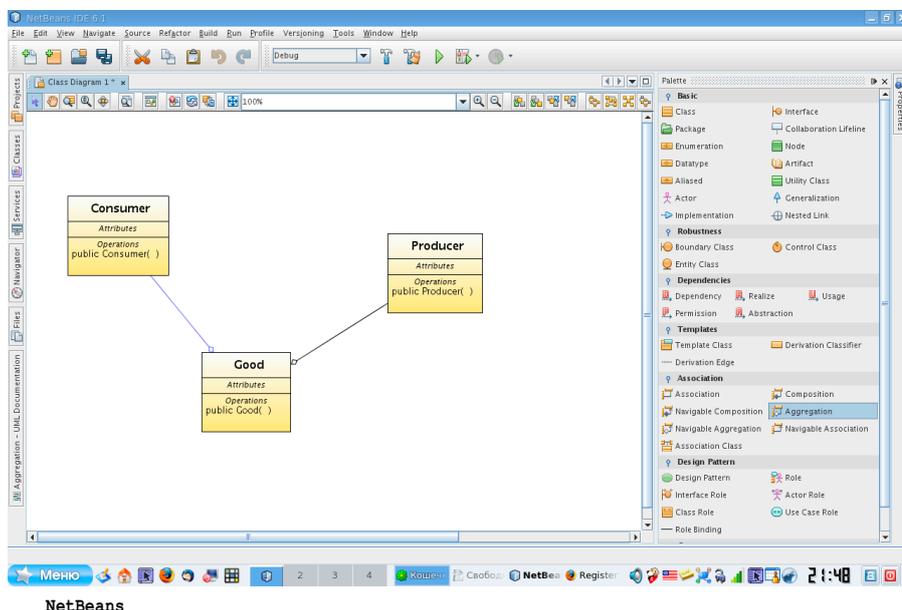
Требования к графическому редактору будут небольшие, а основным является редактирование графики в векторном виде. Таких редакторов достаточно, но из всех выделим два, которые позволяют нарисовать нужные нам схемы:

- ✓ **OpenOffice Draw** (<http://www.openoffice.org>) – достаточно функциональный редактор векторной графики, который позволяет рисовать блоки, стрелки и прочие элементы, используемые в схемах.
- ✓ **Inkscape** (<http://www.inkscape.org>) – мощный редактор векторной графики, позволяющий создать абсолютно любую мыслимую схему.

Написание кода программы

Что требуется от инструмента, помогающего непосредственно писать код программы? Как минимум, это улучшение восприятия кода путем подсветки ключевых слов и разных лексем языка разными цветами, идентификация парных скобок и автоматическая расстановка отступов. Это умеют большинство редакторов в системе Linux и многие в Windows. Дополнительные требования к редактору кода – работа с файлами проекта и возможность анализа потока вывода программы сборки проектов (Make), возможность автоматизированного создания проекта, шаблоны файлов исходных кодов и автоматическое дополнение кода. Для дизайнеров GUI могут быть полезны возможности создания графического интерфейса в редакторе интерфейса. С учетом всех основных требований и ряда дополнительных выбраны следующие среды разработки:

- ✓ **Kate** (<http://kate-editor.org>) – простой инструмент, который обеспечивает базовые требования на отличном уровне. Простой и без излишеств – за это его любят многие разработчики.
- ✓ **Eclipse** (<http://www.eclipse.org>) – мощный инструмент для разработки программ на нескольких языках программирования. Обеспечивает автодополнение кода, историю изменений, проекты и взаимодействие с системами контроля версий CVS и SVN. За счет расширений функциональность легко наращивается. Предусмотрено взаимодействие с отладчиком.
- ✓ **NetBeans** (<http://www.netbeans.org>) – мощная среда разработки, которая, как и Eclipse, поддерживает множество языков программирования. Из дополнительных свойств – автодополнение кода, проекты, работа с системами контроля версий CVS и SVN, расширения. Имеет встроенный UML-ре-



дактор и редактор графического интерфейса для языка Java. Предусмотрено взаимодействие с отладчиком.

☑ **Kdevelop** (<http://www.kdevelop.org>) – среда разработки, обеспечивающая базовый набор функций. Из дополнительных – взаимодействие с отладчиком, проекты (automake + autoseconf), поддержка множества языков программирования, взаимодействие с редактором графических интерфейсов Qt.

☑ **#Develop** (<http://www.icsharpcode.net/OpenSource/SD>) – среда программирования на языке C# для Windows, которая может работать и под Linux с Mono. Обладает всеми базовыми возможностями; из дополнительных – автодополнение кода, взаимодействие с отладчиком, редактор графического интерфейса, работа с системами контроля версий CVS и SVN, проекты, расширения.

В следующей части обзора

В завершающей обзор статье будут рассмотрены инструменты для тестирования, отладки, контроля версий, документации, сборки пакетов и сбора информации об ошибках.

Александр Шайхразеев
(alexander.shaykhrzeev@gmail.com)

Ubuntu для новичков. Часть 2: установка

Первая статья цикла, опубликованная в «Open Source» 030, была посвящена знакомству с основными понятиями Linux, базовой информацией про Ubuntu и о том, как получить этот дистрибутив. Настало время установить Ubuntu.

Критика, отзывы и предложения по циклу статей «Ubuntu для новичков» по-прежнему принимаются на почту или в форум (<http://www.samag.ru/forum/index.php/topic.804.0.html>).

Выбор Ubuntu

В прошлой статье уже упоминалось о различных вариантах Ubuntu, а также о том, как получить этот дистрибутив. Однако, возможно, многих эта информация лишь запутала, и они просто хотят знать, с чего же им начать. Попробуем выбрать нужный вариант Ubuntu.

Первое, на что обращаю внимание, – не стоит останавливать свой выбор на 64-битном варианте Ubuntu. Это связано с потенциальными проблемами «на ровном месте» (например, иногда можно встретить драйверы для сканеров, принтеров и аналогичных устройств, не имеющих поддержку 64-битных архитектур, а уж совсем редко можно столкнуться и с ситуациями, когда какой-то софт просто не собран под 64-битные процессоры).

Второе – нужно выбирать Ubuntu со словом «desktop» в названии ISO-образа. Так вы получите возможности воспользоваться графической установкой дистрибутива и опробовать операционную систему без установки на свой компьютер.

Третий вопрос, который поможет определиться, – это наличие или отсутствие дешевого Интернета. Если у вас есть возможность выкачивать большие объемы информации, будь то любой ресурс в сети или же локальные ресурсы провайдера (провайдеры иногда организуют под-

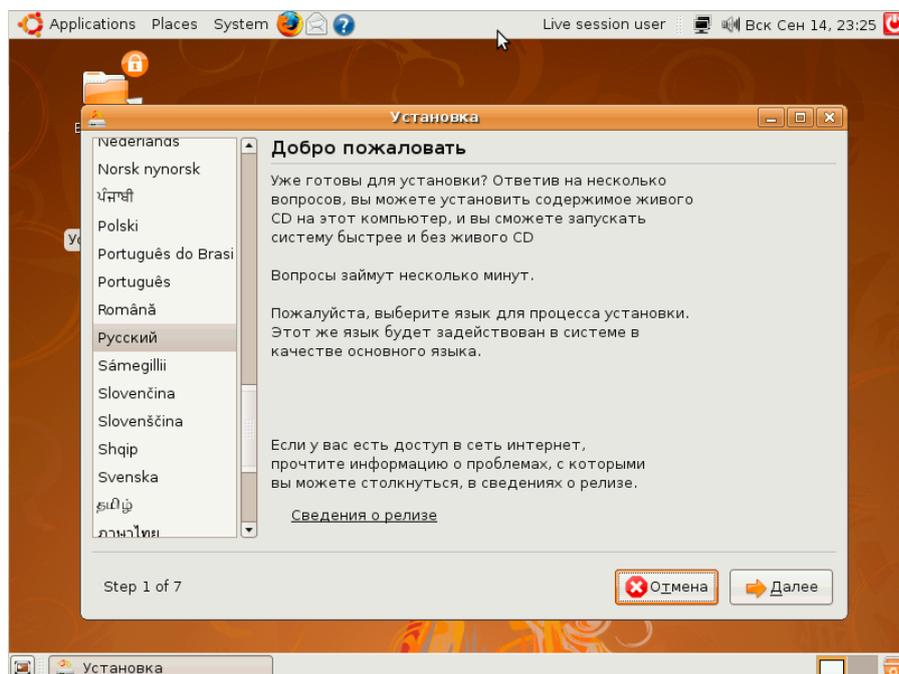
держку репозитория некоторых дистрибутивов), стоит выбрать CD-версию. Объясняется это просто: из Интернета устанавливается более свежее ПО, а в CD-редакции дистрибутива не так много компонентов, которые могут показаться ненужными. Если такого сетевого доступа нет, советую воспользоваться DVD-версией Ubuntu, поскольку в ней можно найти много компонентов без потребности в интернет-доступе. Но при этом стоит учитывать, что в таком варианте вы не будете работать с самыми последними версиями программ.

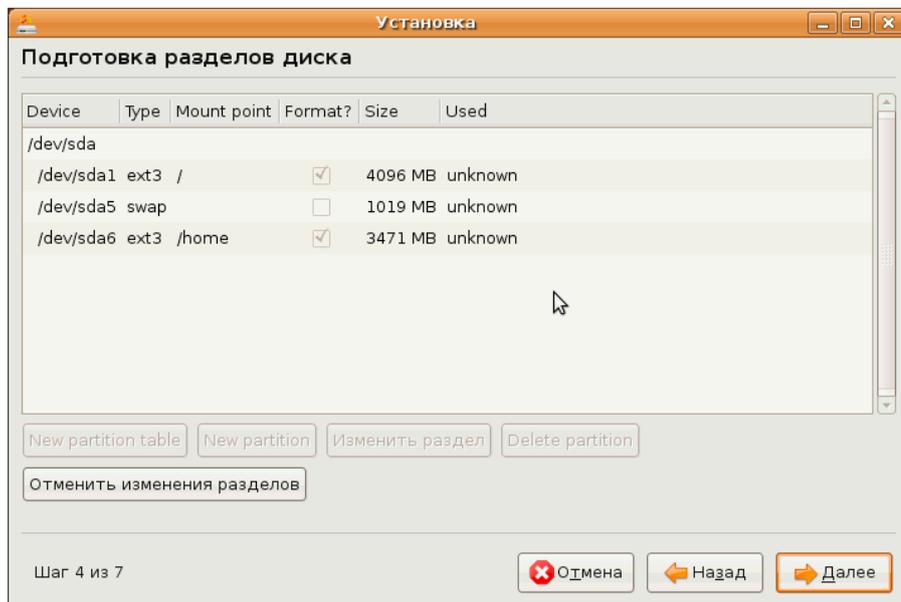
Итак, исходя из всего вышеперечисленного, ISO-образ последней версии дистрибутива на основе LiveCD на одном CD-диске называется: ubuntu-8.04.1-desktop-i386.iso. Его можно найти, например, на FTP-сервере Корбины (<ftp://ftp.corbina.net/pub/Linux/ubuntu-cd/8.04.1>). Далее в статье будет рассматриваться эта версия дистрибутива.

Установка

После того, как дистрибутив выбран и получен, можно приступать к его установке на жесткий диск. Если разметка диска будет выполнена правильно, то с вашими данными и Windows ничего не случится. Если включена загрузка с привода (CD/DVD), то достаточно вставить диск в устройство, загрузиться с него и выбрать русский язык в открывшемся меню. Теперь для «живого» запуска операционной системы следует выбрать пункт «Запустить Ubuntu без установки на компьютер», а чтобы немедленно перейти к установке, следует выбрать пункт «Установить Ubuntu». В первом случае загрузится рабочий стол GNOME и его компоненты, и можно начинать знакомство с дистрибутивом, используемыми программами и доступным меню. Это и есть режим LiveCD. В любой момент можно начать процесс установки, просто кликнув на рабочем столе по иконке «Установка».

На первом шаге нужно выбрать язык, который будет использоваться в будущем по умолчанию. Выбираем «Русский» и нажимаем «Далее». На втором шаге предложат выбрать часовой пояс вручную или





с помощью интерактивной карты мира. Для этого достаточно найти крупный город, находящийся в вашем часовом поясе, и нажать на него на карте. Затем указываем раскладку вашей клавиатуры – смело выставляйте «Russia – Winkey» и не забудьте воспользоваться полем снизу для проверки. Такой выбор («Winkeys» вместо обычной «Russia») связан с тем, что раскладка клавиатуры, которую мы привыкли видеть, не является единственной, и в случае выбора другой раскладки (которую разработчики предлагают по умолчанию) могут возникнуть неудобства: например, знак вопроса не окажется на той же клавише, где цифра «7».

На четвертом шаге стоит остановиться поподробней. Выполнить разметку диска можно несколькими способами. Если у вас есть целый жесткий диск для установки Ubuntu, то вполне подойдет вариант автоматической разбивки. Если же целого диска нет, то можно разбить существующий на необходимое количество разделов вручную. После перехода в режим ручной разбивки выбираем устройство и создаем один раздел (всего их будет три) – нажимаем на кнопку «New Partition» (новый раздел). В поле «Размер нового раздела» указываем не менее 4100 (4 Гб) – это необходимый минимум для системы. В качестве точки монтирования ставим «/» – она указывает на корневой раздел файловой системы. Следующий раздел – это swap (раздел подкачки). Он имеет отличную от ext3 файловую систему – для того, чтобы ее выбрать, в выпадающем меню «Использовать как:» выбираем «Раздел подкачки». Размер файла подкачки зависит от объема оперативной памяти: для 1 Гб и более он не должен быть большим (512 Мб будет достаточно), а для меньших объемов – логично делать его в несколь-

ко раз больше, чем RAM (например, для случая с 512 Мб можно сделать 1 Гб). Кроме того, полезно (хотя и необязательно!) сделать еще один раздел – с точкой монтирования «/home». Он будет предназначен для хранения пользовательских данных: настроек, информации, общих каталогов. Для него также нужно установить файловую систему «ext3», а размер выбрать такой, который понадобится вам: в зависимости от активности использования системы и личных привычек (например, хранить фильмы на рабочем столе) этот объем может сильно варьироваться, но делать его менее 0,5-1 Гб однозначно не стоит. Плюсами такого подхода (создания дополнительного раздела для /home) является изоляция пользовательских данных за счет их сохранения при переустановке или смене дистрибутива, а также тот факт, что система не «делит» общий размер с пользовательскими данными (в противном случае, т.е. если они займут

все дисковое пространство, работоспособность системы нарушится).

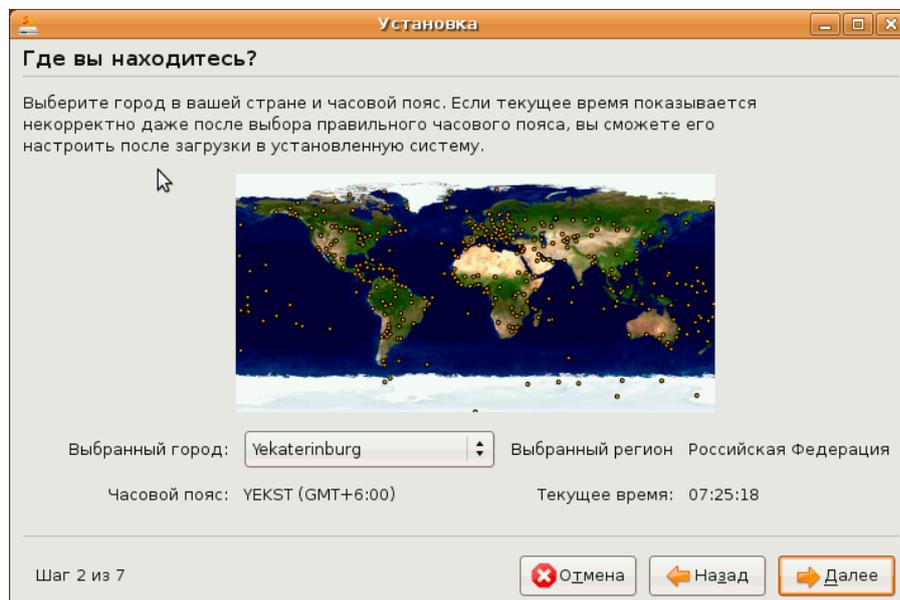
На пятом шаге все достаточно ясно: указывается ваше имя, имя пользователя для входа в систему (логин), затем два раза пароль и имя компьютера. Далее предлагают сделать миграцию из Windows, если эта ОС установлена. Если ее задействовать, то в новую установку Ubuntu будут перенесены пользовательские закладки, обои рабочего стола и другие пользовательские настройки. И завершающий этап – окно со списком всех выбранных во время предыдущих шагов установок настроек. Нажимаем кнопку «Установить», и процесс установки начнется.

Совет: если вы не желаете тратить время на установку некоторых компонентов (обновление списка репозитория и установка локализации) из Интернета, отключите его на время установки. В среднем установка занимает 15-20 минут, так что долго ждать не придется. К тому же режим LiveCD позволит в это время, например, изучить представленные в системе простые игры.

После установки появится окно с просьбой о перезагрузке: можно отложить ее или сразу перезагрузиться. Когда будете перезагружаться, не забудьте вынуть диск из привода, чтобы компьютер вновь не загрузил установку с CD/DVD, – впрочем, инсталлятор Ubuntu заботливо напомнит вам об этом.

Осматриваемся

После перезагрузки у вас появится новый системный загрузчик, который называется GRUB. Он позволит выбирать ОС на стадии загрузки компьютера, обеспечивая удобную возможность использования на компьютере двух ОС: Windows и GNU/Linux. Если у вас уже была уста-



новлена Windows (или ее разные версии), то уже сразу после инсталляции Ubuntu в меню GRUB вы с легкостью найдете и записи, соответствующие представленным в вашей системе Windows-системам.

Загружаемся в Ubuntu Linux, вводим логин и пароль. Вас поприветствует уже знакомый по LiveCD-режиму рабочий стол GNOME. Он представляет собой три области. Первая – верхняя панель, на которой отображаются ярлыки запуска (по умолчанию это веб-браузер Firefox, почтовый клиент Evolution и справка), трей с часами и кнопка выхода, а также меню, о котором позже. На второй панели, находящейся в самом низу, расположен переключатель рабочих столов (справа) и док, в который сворачиваются окна запущенных приложений. Третья область – непосредственно рабочий стол, на котором хранятся ярлыки и прочие файлы, в его области открываются окна запускаемых приложений.

Меню в левом верхнем углу, как видно, тоже разделено на три части: Applications (приложения), Places (места), System (система). В меню «Приложения» все ПО разделено на шесть категорий (при последующей установке новых приложений их число может расти): аудио и видео, графика, игры, Интернет, офис, стандартные и отдельно «Установка/удаление» (об управлении пакетами будет в следующей статье). Как видно из категорий, которые охватывают различные области использования домашнего компьютера, по умолчанию в Ubuntu есть приложения почти на все случаи жизни – при этом здесь предусмотрено по одному приложению на решение каждой задачи. «Из коробки» можно слушать музыку, проигрывать видео (при условии использования открытых форматов), записывать диски, использовать ресурсы Интернета и офисные программы для редактирования текстов, таблиц и т.д. Через меню «Места»

можно получить доступ к любому «месту» на компьютере – в частности, к домашнему каталогу, подключенным устройствам или сети. Гораздо больший интерес представляет меню «Система» – через него можно получить доступ ко всем настройкам в операционной системе, а также получить справку и некоторую интересующую вас информацию.

Заключение

В этой статье было рассказано о том, как установить выбранную версию Ubuntu на свой компьютер, а также были озвучены некоторые мысли по поводу разбивки жесткого диска и описание рабочей среды GNOME. В следующей статье будет подробно рассказано об управлении пакетами и кое-что о настройке системы. Жду отзывов и предложений.

Никита Лялин
(tinman321@gmail.com)

Ее PC и Ubuntu Linux. Часть 2: увеличение времени автономной работы

В этой части цикла «Ее PC и Ubuntu Linux» рассмотрены способы продления времени автономной работы. Статья будет полезна для случаев, когда общие рекомендации не дают желаемого эффекта. Под последними я понимаю такие хитрости, как уменьшение яркости подсветки LCD, когда она не столь нужна, выключение радиомодулей, когда они не используются, и т.д.

Сразу же оговорюсь, что время автономной работы – это, на мой взгляд, самый важный показатель мобильного устройства (а самый надежный способ его увеличить – купить более емкую батарею). Производительность – дело второе, если не третье, ибо вторым я считаю удобство (хотя существует некая грань, когда производительность влияет на удобство).

Исследование энергопотребления ноутбука

Инженеры Intel подарили Linux-сообществу неплохую и весьма информативную программу, наглядно демонстрирующую расход энергии аккумулятора ноутбука теми или иными компонентами системы, – PowerTOP (<http://www.lesswatts.org/projects/powerTOP>). Ею можно воспользоваться для контролирования потребления

энергии процессором (в основном) и шиной USB. Пакет устанавливается обычным образом:

```
$ sudo apt-get install powertop
```

После запуска программы из-под суперпользователя (sudo powertop) вы сможете наблюдать, куда и как расходуется запасенная энергия. Как видно, процессор почти не находится в спящем состоя-

нии (C3) во время бездействия. В основном его состояние оценивается как C0 или, в лучшем случае, как C1-C2 – т.е. рабочее без существенного энергосбережения. Это происходит из-за того, что количество пробуждений в секунду слишком высоко, а для экономичного режима должно составлять около 10.

Итак, наша задача – добиться, чтобы в idle-режиме C3 составляло 99% от все остальных, а время C3 – около или больше 100 мс. Теперь мы будем руководствоваться информацией, полученной от PowerTOP, для увеличения времени автономии.

Уменьшение «тиков» процессора и пара слов о USB

Сразу оговорюсь, что гарантировать какого-то конкретного результата я не могу.

```

PowerTOP version 1.9 (C) 2007 Intel Corporation

Cn          Avg residency      P-states (frequencies)
C0 (ЦП работает)      ( 0,7%)
C1              0,0мс ( 0,0%)
C2              3,3мс ( 0,1%)
C3            118,0мс (99,2%)

wakeups-from-idle per second : 8,7 interval: 30,0s
Оценка потребления энергии через ACPI недоступна

Top causes for wakeups:
 19,3% ( 1,7)  gnome-terminal : schedule_timeout (process_timeout)
 12,4% ( 1,1)  <interrupt> : acpi
 11,6% ( 1,0)  pulseaudio : schedule_timeout (process_timeout)
 11,6% ( 1,0)  Xorg : schedule_timeout (process_timeout)
    
```

```
PowerTOP version 1.9 (C) 2007 Intel Corporation
Cn Avg residency P-states (frequencies)
C0 (ЦП работает) ( 1,0%) 450 МГц 0,0%
C1 0,0мс ( 0,0%) 338 МГц 100,0%
C2 10,7мс ( 1,1%) 225 МГц 0,0%
C3 413,4мс (98,0%) 113 МГц 0,0%

Wakeups from idle per second : 3,5 interval: 30,0s
Оценка потребления энергии через ACPI недоступна

Top causes for wakeups:
28,7% ( 1,0) Xorg : schedule_timeout (process_timeout)
25,0% ( 0,9) <interrupt> : acpi
12,0% ( 0,4) <interrupt> : libata
12,0% ( 0,4) gnome-panel : schedule_timeout (process_timeout)
6,5% ( 0,2) <kernel core> : neigh_table_init_no_netlink (neigh_periodic_timer)
3,7% ( 0,1) <kernel module> : neigh_table_init_no_netlink (neigh_periodic_timer)
2,8% ( 0,1) gconfd-2 : schedule_timeout (process_timeout)
1,9% ( 0,1) gnome-volume-na : schedule_timeout (process_timeout)
1,9% ( 0,1) <kernel core> : page_switchback_init (ub_timer_fn)
1,9% ( 0,1) pdflush : do_journal_end (delayed_work_timer_fn)
1,9% ( 0,1) hald : schedule_timeout (process_timeout)
0,9% ( 0,0) <kernel module> : acpi_thermal_check (acpi_thermal_run)
0,9% ( 0,0) <kernel core> : end_that_request_last (laptop_timer_fn)

[O] - Выйти [R] - Обновить
```

Результат после длительных экспериментов. Всё работает!

И любые «достижения» на поприще подобных оптимизаций крайне индивидуальны и субъективны: у одного пользователя время работы устройства может существенно увеличиться, а для другого – прирост в 1-2 часа (и уж тем более в 30-60 минут) может ничего не значить.

Если вы все же намерены добиться некоторого результата, необходимо сделать несколько вещей:

1. Отключите эффекты композитного менеджера Compiz. Это делается через меню: «Система → Параметры → Внешний вид → Визуальные эффекты → Без эффектов».

2. Избегайте бесполезных обращений к USB-портам и включите режим Persist для USB-подсистемы в целом. Это позволит тратить меньше энергии на обслуживание USB-портов как во время их работы, так и во время бездействия (подробнее – <http://www.mjmwired.net/kernel/Documentation/usb/persist.txt>).

Основной виновник излишнего потребления USB – hald (HAL daemon). Эта программа, работающая в фоне, создает слой аппаратной абстракции, то есть организует единый и универсальный интерфейс для работы с устройствами, скрывая их различия (подробнее – на [http://ru.wikipedia.org/wiki/HAL_\(freedesktop.org\)](http://ru.wikipedia.org/wiki/HAL_(freedesktop.org)) и http://ru.wikipedia.org/wiki/Hardware_abstraction_layer). Если у вас ничего не подключено к USB-порту ноутбука, то беспокоиться почти не о чем – единственным исключением является встроенный кард-ридер. Из-за него часть подсистемы USB не может работать в persist mode, так как hald каждые 2 секунды опрашивает ее на предмет наличия карты памяти. Чтобы отключить ридер, воспользуйтесь командой:

```
$ sudo echo 0 > /proc/acpi/asus/cardr
```

3. Отключайте интерфейс eth0 при отсутствии необходимости в таковом:

```
$ sudo ifconfig eth0 down
```

4. Включите hpet, чтобы процессор не генерировал лишние тики (подробнее – <http://en.wikipedia.org/wiki/HPET>). Добавьте параметр clocksource=hpet в строку параметров ядра. Для этого откройте в любом текстовом редакторе файл /boot/grub/menu.lst и измените строку:

```
# defoptions=quiet splash
```

на:

```
# defoptions=quiet splash clocksource=hpet
```

Сохраните файл (не забудьте убедиться, что файловая система смонтирована на чтение-запись), а затем обновите загрузчик GRUB:

```
$ sudo update-grub
```

После перезагрузки изменения вступят в силу.

5. Отключите все службы, которые так или иначе не нужны. Например, все логины, armd, trackerd и т.п.

6. Уберите с панели GNOME все лишнее – особенно это касается «интерактивных» элементов вроде «Системного монитора».

Подведем итоги. Если процессор находится в C3 около 100 мс, то у вас появилось около 30-40 дополнительных «автономных» минут. Возможно, даже больше: на сайте программы PowerTOP говорится о целом часе, но я такого прироста не замечал (у меня получается добиться лучших результатов только при использовании внешнего дисплея вместо встроенного).

Торможение FSB

Внимание! Все дальнейшие действия вы совершаете на свой страх и риск! Ни автор, ни редакция не несет ответственности за возможную порчу ноутбука.

Теперь о частотах. Включите модуль разгона:

```
$ sudo modprobe i2c_i801
$ sudo modprobe eee
```

С его помощью мы будем замедлять работу, причем FSB (http://ru.wikipedia.org/wiki/Front_side_bus), а не процессора. Это вызвано тем, что Intel Celeron M лишен возможности сохранять энергию путем изменения своей частоты (этой особенностью обладает Intel Pentium M – см. <http://www.bay-wolf.com/speedstep.htm#10>). Если менять только частоту процессора, то она будет успешно уменьшаться, все будет работать медленнее и вообще возникнет ощущение, что компьютер получил еще часа 3-4 автономной работы. Однако, как показала практика, выиграть удастся лишь минут 20, и работать при этом становится затруднительно (даже меню открывается медленно).

Вернемся к практике понижения частоты шины. Выполним:

```
$ cat /proc/eee/fsb
```

```
70 24 0
```

Что означают эти числа? 70 – первый множитель, 24 – первый делитель, 0 – флаг управления напряжением на процессоре (0 – пониженное, т.е. то, что нам нужно, а 1 – повышенное, требуется для разгона FSB до 100 МГц и выше, если повезет). При данных параметрах первый множитель будет равен значению частоты шины. Чтобы не перебирать значения вручную (после каждого значения с шагом в 1 МГц придется делать задержку на 1-2 секунды перед следующим шагом), можно воспользоваться скриптом: (<http://wiki.eeuser.com/media/howto:fsb.txt.gz?id=howto%3Aoverclockfsb>). Распакуйте архив, например в /usr/bin, и сделайте файл исполняемым.

```
$ sudo chmod +x fsb.txt
```

Наша цель – понизить частоту FSB до значения в пределах 45-50 (если вы используете дешевую плашку оперативной памяти на 1 Гб, как я) или 40-45 (а может быть и ниже – если вы используете дорогую плашку на 1 Гб или на 512 Мб, которая идет в комплекте). Отредактируйте fsb.sh (в него я переименовал fsb.txt из архива, полный путь до этого файла-скрипта стал таким: /usr/bin/fsb.sh) так,

чтобы MAX_STEP был не больше 2 (лучше не рисковать – при больших значениях возможны проблемы с оперативной памятью и рябь на LCD), а SLEEP_INTERVAL установите в 1 (здесь число может быть любым целым, однако в 0 устанавливать нежелательно – слишком быстрой будет смена частоты FSB, поэтому также могут возникнуть проблемы с ОЗУ и встроенным дисплеем). Теперь запустим скрипт с числом 50 в качестве аргумента (я настоятельно не рекомендую устанавливать значения вне границ 35-110):

```
# /usr/bin/fsb.sh 50
```

В результате исполнения скрипта будет получена частота FSB в 50 МГц. При этом частота процессора будет равна 450 МГц (заметьте, скорость отклика «на глаз» осталась прежней). Вот и все относительно торможения FSB. Осталось эту строчку добавить в /etc/rc.local до «exit 0», но, разумеется, после modprobe i2c_i801 и modprobe eee:

```
-----<другие автозапускаемые сервисы>-----  
modprobe i2c_i801  
modprobe eee  
-----<другие автозапускаемые сервисы>-----  
/usr/bin/fsb.sh 50  
exit 0
```

Торможение процессора

Теперь об относительно безопасном изменении частоты процессора. Хотя этот метод и не столь полезен, однако все же стоит его рассмотреть. Сначала подключите соответствующий модуль командой:

```
$ sudo modprobe p4-clockmod
```

Добавьте эту строчку (без sudo) в /etc/rc.local до «exit 0» или название модуля в /etc/modules, если вы хотите, чтобы возможность контроля частоты процессора была включена с момента загрузки системы (имейте в виду, что у некоторых наблюдаются проблемы при работе, когда этот модуль загружен). Теперь

нужно отключить все службы, которые заведуют динамическим изменением частоты процессора. Воспользуемся `ondemand governor` (подробнее о нем – на <http://www.mjmwired.net/kernel/Documentation/cpu-freq/governors.txt>) по одной причине: его использование не вызывает прерываний, а следовательно, процессор сможет еще чуть дольше находиться в состоянии C3. Следующий шаг – установка минимального значения. Я рекомендую поставить 337,5 МГц. Для этого достаточно внести в /etc/rc.local до «exit 0»:

```
echo 337500 > /sys/devices/system/cpu/cpu0/cpufreq/scaling_min_freq
```

И после этой строчки добавить:

```
echo ondemand > /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

Теперь частота процессора будет варьироваться от 337,5 МГц до той, какую мы задали при помощи торможения FSB (однако для ядра оно будет вплоть до 900 МГц, что, конечно, не отражает действительной картины).

Послесловие

Несмотря на все эти советы по поводу продления жизни устройства от аккумулятора, помните, что лучшее решение увеличения автономной работы – покупка более емкой батареи. Хочу лишь заметить, что после вышесказанных оптимизаций мне удалось увеличить время автономии на 30-40 минут: сразу после установки Ubuntu и настройки драйверов было около 2:40-3:00, а после оптимизации стало 3:20-3:40. Емкость моей батареи – 5200 мАч, однако, как показывает сравнение с другими батареями, моя, видимо, слегка неисправна. Все тесты проводились в idle-режиме, включение Wi-Fi без постоянной передачи данных повлияло на результаты не столь критично: время автономии уменьшилось всего на 10-15 минут. На этом я завершаю свое повествование об установке Ubuntu 8.04.1 на Eee PC 701.

Сергей Гулин
(sugia@yandex.ru)

«Open Source» приглашает к сотрудничеству!

Электронное приложение «Open Source» всегда открыто для сотрудничества с новыми авторами, с читателями и их конструктивными предложениями по улучшению издания, обоснованной критикой и любыми отзывами, с компаниями, занимающимися разработкой и продвижением программного обеспечения с открытым кодом. Приветствуются все энтузиасты, желающие опубликовать у нас свои статьи.

Тематика нужных материалов очевидна из предназначения приложения, то есть FOSS (Free and Open Source Software): теория и практическое применение; исторические сведения, анализ сегодняш-

него положения, прогнозы на будущее и другие аспекты, связанные с открытым ПО.

Среди наиболее интересных на данный момент общих тем можно выделить:

- ☑ общие обзоры новых и/или интересных проектов Open Source и конкретных приложений, свежих версий дистрибутивов Linux, *BSD и других систем;
- ☑ советы и рекомендации новичкам в мире GNU;
- ☑ истории успеха применения/распространения ПО с открытым кодом;
- ☑ философия и идеология Free Software;
- ☑ разработка приложений с применением средств Open Source.

Желательный объем статей: 6000 или 12000 символов (с пробелами). Примеры актуальных сейчас тем для статей публикуются на <http://osa.samag.ru/todo>. Но не стоит строго ограничиваться приведенными выше рамками!

Если у вас есть свои темы и предложения, присылайте – рассмотрим все вопросы.

Редакция постоянно стремится к развитию и совершенствованию своего издания, поэтому мы рады активным читателям и их комментариям как относительно публикуемых материалов, так и самой структуры приложения, его оформления и приоритетов.

Публичное обсуждение «Open Source» проводится на форуме сайта журнала «Системный администратор» по адресу: <http://osa.samag.ru/forum>.

Связаться с редакцией можно по электронной почте osa@samag.ru.

Подписные индексы:

20780*

+ диск с архивом статей
2008 года

81655**

без диска

по каталогу агентства
«Роспечать»

88099*

+ диск с архивом статей
2008 года

87836**

без диска

по каталогу агентства
«Пресса России»

* Годовой
** Полугодовой
*** Диск вкладывается
в февральский
номер журнала,
распространяется только
на территории России

Подписка на журнал «Системный администратор»

Российская Федерация

- ☑ Подписной индекс: годовой – **20780**, полугодовой – **81655**
Каталог агентства «Роспечать»
- ☑ Подписной индекс: годовой – **88099**, полугодовой – **87836**
Объединенный каталог «Пресса России»
Адресный каталог «Подписка за рабочим столом»
Адресный каталог «Библиотечный каталог»
- ☑ Альтернативные подписные агентства: агентство «Интер-Почта» (495) 500-00-60, курьерская доставка по Москве
агентство «Вся Пресса» (495) 787-34-47
агентство «Курьер-Пресссервис»
агентство «ООО Урал-Пресс» (343) 375-62-74
- ☑ Подписка On-line
<http://www.arzi.ru>
<http://www.gazety.ru>
<http://www.presscafe.ru>

СНГ

В странах СНГ подписка принимается в почтовых отделениях по национальным каталогам или по списку номенклатуры «АРЗИ»:

- ☑ **Азербайджан** – по объединенному каталогу российских изданий через предприятие по распространению печати «Гасид» (370102, г. Баку, ул. Джавадхана, 21)

- ☑ **Казахстан** – по каталогу «Российская пресса» через ОАО «Казпочта» и ЗАО «Евразия пресс»
- ☑ **Беларусь** – по каталогу изданий стран СНГ через РГО «Белпочта» (220050, г. Минск, пр-т Ф. Скорины, 10)
- ☑ **Узбекистан** – по каталогу «Davriy nashrlar», российские издания через агентство по распространению печати «Davriy nashrlar» (7000029, г. Ташкент, пл. Мустакиллик, 5/3, офис 33)
- ☑ **Армения** – по списку номенклатуры «АРЗИ» через ГЗАО «Армпечать» (375005, г. Ереван, пл. Сасунци Давида, д. 2) и ЗАО «Контакт-Мамул» (375002, г. Ереван, ул. Сарьяна, 22)
- ☑ **Грузия** – по списку номенклатуры «АРЗИ» через АО «Сакпресса» (380019, г. Тбилиси, ул. Хошараульская, 29) и АО «Мацне» (380060, г. Тбилиси, пр-т Гамсахурдия, 42)
- ☑ **Молдавия** – по каталогу через ГП «Пошта Молдовой» (МД-2012, г. Кишинев, бул. Штефан чел Маре, 134)
по списку через ГУП «Почта Приднестровья» (МД-3300, г. Тирасполь, ул. Ленина, 17)
по прайс-листу через ООО агентство «Editil Periodice» (МД-2012, г. Кишинев, бул. Штефан чел Маре, 134)
- ☑ Подписка для **Украины**:
Киевский главпочтамт
Подписное агентство «KSS»
Телефон/факс (044)464-0220