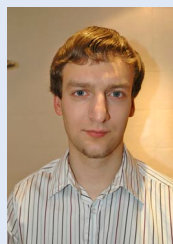


Колонка главного редактора



Так уж вышло, что буквально за несколько часов до написания этой колонки я наконец-то получил свой долгожданный смартфон Neo FreeRunner. За развитием этого Open Source-

устройства я слежу достаточно давно и внимательно. Как и положено, впечатлений много, и чувства смешанные. Само устройство оказалось не таким страшным и неудобным, как все время ощущалось при просмотре его изображений в сети. Аппаратная начинка всем давно известна, и в комментариях не нуждается. Но вот программная часть действительно требует существенной доработки — пожалуй, даже «переработки». Резюме такое: FreeRunner очень далек от массового пользователя, но представляет собой отличную «игрушку» для заинтересованных.

Все это напомнило Linux-десктоп 5-7-летней давности... Теперь стоит взглянуть на будущее, которое постигло Linux на настольных ПК, в контексте нынешних тенденций мобильного мира и последних активностей проекта Openmoko по разработке софта. Картина получается радужной и перспективной — даже несмотря на тот факт, что не стоит путать пользователей смартфонов с компьютерными. Сейчас есть узкая ниша, что было доказано скоростью распространения FreeRunner по всему миру и быстрому формированию сообщества. Кто знает, что будет следующим шагом?

Главный редактор
Дмитрий Шурупов
(osa@samag.ru)

«Open Source»

электронное приложение к журналу

«Системный администратор»

№35, 21 ноября 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

МИД Германии экономит с Open Source-десктопами

Рольф Шустер (Rolf Schuster), немецкий дипломат и в прошлом глава IT-отдела Министерства иностранных дел Германии, заявил, что обслуживание десктопов с программным обеспечением с открытым кодом (Open Source) обходится гораздо дешевле, чем в случае с проприетарным ПО. Об этом он сообщил на конференции Open Source World в Малаге (Испания).

МИД Германии переводит около 11 тысяч десктопов на GNU/Linux и Open Source. По словам Шустера, этот шаг позволяет добиться существенной экономии по сравнению с другими министерствами: «В МИД десктопы функционируют в очень удаленных и малодоступных местах. Но на обслуживание каждого десктопа мы тратим всего 1000 евро в год. Это гораздо дешевле, если сравнивать с другими министерствами, где средние ежегодные расходы за десктоп составляют более 3000 евро».

Завершение миграции Министерства иностранных дел Германии на GNU/Linux и другое ПО с открытым кодом ожидается к лету 2009 года. На данный момент переход осуществлен более чем в половине посольств и представительств (всего их 230). Во время миграции возникают и некоторые сложности: «Переход осуществляется не без проблем. Например, наши разработчики не знают японского языка, поэтому не понимают местных сложностей, и нам пришлось найти специалиста в Японии, чтобы он помог разобратся с шрифтами в OpenOffice.org».

Motorola и Google стали спонсорами GNOME Foundation

Некоммерческая организация GNOME Foundation, регулирующая вопросы по развитию популярной графической среды с открытым исходным кодом GNOME, объявила о том, что к числу ее спонсоров присоединились компании Motorola и Google. Их представители вошли в состав совещательного органа GNOME Advisory Board.

Каких конкретных целей хотят добиться в Google и Motorola этим «присоединением» к GNOME Foundation, пока не совсем понятно. В Motorola, конечно же, сразу недвусмысленно указали на свой интерес к инициативе GNOME Mobile и, в частности, к таким проектам, как GStreamer и Bluez. Участие Google можно объяснить

последними событиями, связывающими интернет-гиганта с Motorola и потенциально GNOME: Motorola сделала ставку на Android как на главную Linux-платформу для своих смартфонов. Поэтому вполне возможно, что обе компании заинтересованы в использовании компонентов GNOME в GUI-приложениях мобильной платформы Android.

Продажи Linux-смартфонов выросли на 49 процентов

Исследовательская фирма Canalsys сообщила, что продажи смартфонов, работающих под управлением операционных систем с ядром Linux, за последний год выросли на 49%.

В целом за последний год рынок смартфонов вырос на 28%, и единственной платформой, которая потеряла долю за этот период, стала операционная система Symbian, исходный код которой обещали открыть в 2010 году. В третьем квартале 2008 года были проданы 39,9 млн смартфонов, благодаря чему устройства этого класса составили 13% от всего рынка мобильных телефонов (по сравнению с 11% во втором квартале 2008 года).

Наиболее внушительный рост наблюдается у платформы компании Apple, достигнутый за счет успешных продаж iPhone: доля мобильной версии Mac OS X с третьего квартала 2007 года выросла на 523%, составив 17,3% от всех смартфонов. На втором месте по росту доли среди мобильных платформ для смартфонов — Research In Mobile (RIM, устройства BlackBerry): ее популярность выросла на 83,5%, составив 15,2%. На третьем — Linux, доля которой выросла на 49% и составила 5,1%. Замыкает этот список Microsoft, популярность платформы которой (Windows Mobile) выросла на 42,9%, составив 13,6%.

Безоговорочным лидером рынка по-прежнему остается Symbian, под управлением которой работает 46,6% смартфонов, даже несмотря на снижение этой доли на 12,4%.

Интересно, что в сентябре компания Gartner заявляла, что продажи Linux-смартфонов, наоборот, упали — по их данным, доля Linux на рынке смартфонов, снизилась с 10% до 7%.

SpringSource купила G2One, авторов Groovy и Grails

Компания SpringSource объявила о приобретении G2One, занимающейся поддерж-

кой и развитием таких популярных Open Source-приложений, как Groovy и Grails.

Сама SpringSource снискала известность благодаря разработке популярного Open Source-фреймворка Spring для платформы Java. Также SpringSource осуществляет поддержку Java веб-сервера Apache Tomcat. G2One специализируется на объектно-ориентированном языке программирования Groovy для Java Platform (он может выступать в качестве альтернативы языку Java) и фреймворке Grails, ранее известному, как Groovy on Rails, поскольку создавался как некий аналог Ruby on Rails (RoR) для Groovy. Интересно, что все перечисленные Open Source-продукты обеих компаний (Spring, Groovy и Grails) распространяются под лицензией Apache License.

Ожидается, что с этим приобретением SpringSource начнет предоставлять поддержку и другие услуги разработчикам и корпоративным пользователям Groovy и Grails.

Fixstars купила Terra Soft, авторов Yellow Dog Linux

Японская корпорация Fixstars, специализирующаяся на решениях на базе микропроцессорной архитектуры Cell, объявила о покупке компании Terra Soft Solutions, занимающейся разработкой дистрибутива Yellow Dog Linux.

Дистрибутив Yellow Dog Linux (YDL) был создан еще в 1999 году, основывается на RHEL/CentOS и известен как популярное решение для компьютеров с процессорами архитектуры семейства Power. Теперь его поддержкой и разработкой будет заниматься подразделение Fixstars Solutions, в которое перешли сотрудники компании Terra Soft.

Terra Soft Solutions, создавшая Linux-дистрибутив YDL, быстро вошла в список лучших реселлеров Apple и в свое время стала обладательницей уникальной лицензии на установку альтернативной операционной системы на компьютеры Apple PowerPC для таких организаций, как компании Lockheed Martin и Boeing, национальное агентство NASA и ведущие университеты.

Novell начала поддержку пользователей Red Hat и CentOS

Компания Novell объявила о старте программы, нацеленной на плавный перевод пользователей Linux-дистрибутивов Red Hat и CentOS на свое корпоративное решение SUSE Linux Enterprise Server (SLES).

С новой инициативой Novell предлагает обладателям серверов, работающих под управлением систем Red Hat Enterprise Linux (RHEL) или CentOS, поддержку на 2 года, в течение которых клиент должен мигрировать на SLES. Предложение действительно для RHEL версий 3, 4, 5 и соответствующих им релизов CentOS. При этом срок поддержки ограничивается не только пресловутыми двумя годами, но и концом жизненного цикла продуктов (если последний наступит раньше, чем через 2 года, то и поддержка Novell перестанет действовать).

В рамках новой программы Novell будет выпускать для своих «поддерживаемых» клиентов бинарные патчи и обновления, самостоятельно собранные из исходного кода, распространяемого Red Hat. Компания гарантирует бинарную совместимость с RHEL и CentOS. Помимо этого в Novell обещают круглосуточную техни-

ческую поддержку по телефону и электронной почте.

Поскольку конечная цель такой «поддержки» пользователей RHEL/CentOS является все же миграция на решения Novell (самостоятельно клиентами или через сторонних консультантов), компания запустит веб-сайт с исчерпывающей информацией о переходе на SLES. Об этом стало известно от Джастина Штейнмана (Justin Steinman), вице-президента по маркетингу решений и продуктов Novell.

Ubuntu Linux портируют на архитектуру ARM к апрелю 2009

Компания Canonical, обеспечивающая коммерческую поддержку популярного Linux-дистрибутива Ubuntu, объявила о намерении портировать Ubuntu Desktop Linux на архитектуру ARMv7.

Интересно, что долгое время до этого известный финский производитель мобильных телефонов Nokia спонсировал неофициальный проект по портированию Ubuntu на ARM. Но теперь, когда в Linux-ядре появились новые возможности по поддержке ARM, а Canonical «подружилась» с компанией ARM Holdings, порт Ubuntu на ARM выйдет на принципиально новый этап развития. В частности, теперь можно будет скомпилировать для ARM все пакеты программного обеспечения, входящие в состав Ubuntu, – даже Java и OpenOffice.org.

Выход порта Ubuntu Linux на архитектуру ARM запланирован на апрель 2009 года.

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

OpenBSD как десктоп: реально? Вполне!

В недалёком прошлом сложилось мнение, что OpenBSD, будучи ориентированной на высоконагруженные серверы и маршрутизаторы, не подходит для десктопа. Данная статья является живым свидетельством того, что это совсем не так. Сейчас у меня OpenBSD работает исключительно на серверных платформах, но 2,5 года назад, в то время, когда я решил ознакомиться с этой операционной системой, она и заняла место на рабочей машине. Тогда моей задачей было создание среды, в которой было бы удоб-

но работать и отдыхать. Попробую поделиться этим опытом.

Первоначальная установка

Начиная с версии 4.2 OpenBSD обзавелась полным установочным ISO-образом, с которого можно поставить систему без необходимости в подключении к Сети. До этого момента подобный образ нужно было создавать самостоятельно с помощью подручных средств (об этом чуть позже) или использовать ISO-образ network installation, в который входи-

ли само ядро системы и минимальный набор программ для подготовки компьютерного железа к установке новой копии OpenBSD из Сети.

Установка представляет собой несколько последовательных этапов, как и в любой другой UNIX-подобной системе: разметка диска, создание и монтирование разделов, копирование системных файлов, выбор временной зоны и перезагрузка в уже установленную систему. Подробно останавливаться на этом мы не будем – благо в Сети существует достаточно много How-To по инсталляции (например, <http://www.openbsd.org/faq/faq4.html#Install> и <http://www.openbsd101.com/installation.html>).

Единственное, что хотелось бы отметить на данном этапе, – не забудьте выбрать x-sets, когда вам предложат выбрать

пакеты для установки. Те из них, что начинаются с «x», и есть составляющие графической оболочки: сервер X-Window и дополнительные библиотеки, драйверы для видеокарт и других устройств, шрифты, базовые конфигурационные файлы (например, для XDM) и т.п.

Установка дополнительных программ

В OpenBSD существует 2 способа установки программ: из пакетов (packages) и портов (ports). Пакеты представляют собой уже скомпилированные версии программ под различные архитектуры, запакованные специальным образом в архивы и выложенные в Сети на зеркалах проекта OpenBSD. Порты же – это некий набор скриптов, с помощью которого можно скомпилировать программы вручную: при этом скачиваются исходные коды, производится конфигурация под установленную версию системы, компиляция и, наконец, установка. Стоит отметить, что на последнем этапе создаётся пакет. Разработчики OpenBSD рекомендуют устанавливать программы из пакетов, т.к. преимуществ у альтернативного метода установки нет – за исключением, пожалуй, лишнего потерянного (израсходованного на компиляцию) времени и дополнительного дискового пространства.

Управление пакетами в OpenBSD производится главным образом при помощи команд `pkg_add` (для установки и/или обновления), `pkg_delete` (для удаления) и `pkg_info` (для просмотра информации об установленных пакетах и о каждом из них в частности). Для этих команд также существуют псевдонимы: `pkg add`, `pkg delete`, `pkg info`.

Перед тем как начать работу с пакетами, нужно указать переменную `PKG_PATH`. Это путь к каталогу (расположенному на диске либо в Сети), где находятся откомпилированные и готовые к использованию пакеты, например:

```
/mnt/cdrom/`uname -r`/packages/`machine -a`/
```

или

```
ftp://your.ftp.mirror/pub/OpenBSD/`uname -r`/packages/`machine -a`/
```

Делается это следующей командой:

```
$ export PKG_PATH=ftp://ftp.gammar.ru/pub/OpenBSD/`uname -r`/`machine -a`/
```

Эту строку полезно добавить в `~/.profile`, чтобы каждый раз не вводить её вручную. Теперь для установки пакета воспользуемся командой `pkg_add` (`sudo` требуется для получения прав суперпользователя, т.к. обычный пользователь в OpenBSD не имеет прав для установки):

```
$ sudo pkg_add -v screen-4.0.3p1
```

```
parsing screen-4.0.3p1
installed /etc/screenrc from /usr/local/share/examples/screen/
screenrc | 71%
screen-4.0.3p1: complete
```

После этого пакет готов к работе. Чтобы просмотреть наличие установленных пакетов, достаточно выполнить команду «`pkg info`» без аргументов:

```
$ pkg info
```

```
autoconf-2.61p3      automatically configure source code on many Un*x platforms
automake-1.9.6p2     GNU standards-compliant Makefile generator
bzip2-1.0.5          block-sorting file compressor, unencumbered
curl-7.18.2          get files from FTP, Gopher, HTTP or HTTPS servers
dtach-0.8            tiny program emulating the detach feature of screen
```

Для просмотра подробной информации о каждом пакете до-

статочно выполнить ту же команду с единственным аргументом в качестве имени желаемого пакета:

```
$ pkg info yasm
```

Если требуется показать определённую информацию о пакете, нужно использовать флаги (например, `-R` для вывода списка пакетов, которые требуют для установки указанный):

```
$ pkg info -R libshout
```

```
Information for inst:libshout-2.2.2p0
```

```
Required by:
mpd-0.13.2
```

Для обновления пакетов используем уже известную команду «`pkg add`» с флагом `-u`:

```
$ sudo pkg_add -u unzip
```

```
unzip-5.52 (extracting): complete
unzip-5.51 (deleting): complete
unzip-5.52 (installing): complete
Clean shared items: complete
```

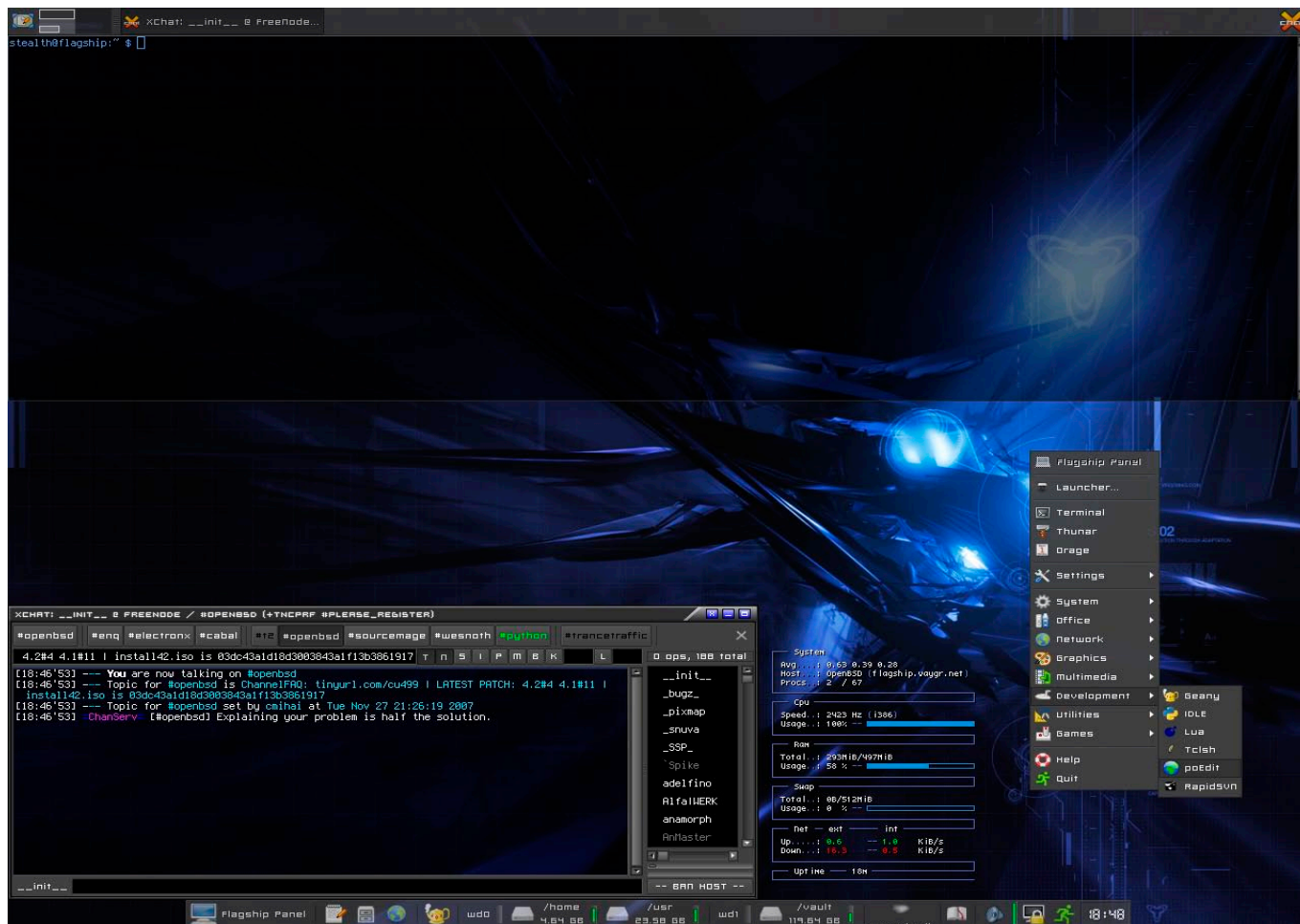
За дополнительной информацией о функциях `pkg_add` можно обратиться к справочной документации `pkg_add(1)` и FAQ на официальном сайте OpenBSD: <http://www.openbsd.org/faq>.

Настройка системы под нужды desktop-пользователя

Конфигурация X-сервера в OpenBSD ничем не отличается от этой процедуры в других операционных системах, где используется X.Org (<http://www.x.org>). Тот же самый `xorg.conf`, с тем же синтаксисом. А вот драйверы для видеокарт NVIDIA – другие. В OpenBSD для них можно использовать только свободный драйвер `nv`. Для полноценных 3D-игр (например, Nexuiz) OpenBSD вкупе с подобными картами, к сожалению, не годится. Производительность работы драйвера, исходные коды которого не доступны, а OpenBSD-сборки не существует, в несколько десятков раз превышает производительность `nv`. Чтобы разрешить возникающие проблемы, возможно, потребуются заглянуть в `/var/log/Xorg.0.log` – всё-таки X-сервер в OpenBSD модифицируется командой разработчиков.

Теперь можно перейти непосредственно к установке дополнительных программ для desktop-использования. Официальный веб-интерфейс для портов OpenBSD находится по адресу <http://openports.se>. Там можно найти желаемую программу по слову в её описании или названию, а также проследить за обновлением программ, идущих в ветку `-current`.

На данный момент количество портов для платформы i386 весьма небольшое – чуть более 5 тысяч. Связано это с тем, что, во-первых, большинство Open Source-программ в настоящее время пишется под Linux (нельзя же не замечать очевидного), а во-вторых, с особенностями библиотек (и C, и многих других), которые несколько отличаются от аналогов GNU и постоянно изменяются. Судя по довольно узкой аудитории пользователей OpenBSD, процесс «портирования» (адаптации программ под работу в OpenBSD) сильно затянут, но прогресс все же есть. С каждым годом количество портов увеличивается (а следовательно, увеличивается и количество пакетов), да и разработчики программ становятся всё прилежнее: создают унифицированные программы (работающие одинаково как в NetBSD, так и в Linux с Mac OS X) и стараются писать код по стандартам POSIX для UNIX/UNIX-подобных операционных систем, улучшая не только количество, но и качество кода.



Рабочий стол OpenBSD: XFCE с темой Enclave

В первую очередь следует подобрать рабочую среду (DE, Desktop Environment). В OpenBSD выбор довольно широк: GNOME, KDE, XFCE, а также пара десятков оконных менеджеров, включая мой любимый Fluxbox и fvwm, входящий в стандартную поставку OpenBSD. Я предпочитаю минимализм и в то же время достаточную гибкость настройки, поэтому выбрал XFCE. Подобная лёгкая среда хорошо уживается с системой и не требует многих зависимостей, а это в свою очередь упрощает поддержку системы (с минимальным возможным количеством установленных пакетов).

Что же ещё может понадобиться обычному пользователю рабочей станции? Офисный пакет OpenOffice.org (версий 2 и 3) прекрасно работает под OpenBSD. И поскольку для его работы требуется JRE (Java Runtime Environment), в процессе установки будет необходимо принять условия дополнительного лицензионного соглашения.

Для работы с PDF-документами ePDFView и Evince будут весьма кстати. Настройка принтера в OpenBSD – процесс безболезненный, но это, конечно, зависит от модели устройства. Обычно не возникает никаких проблем, если принтер использует драйвер gutenprint.

Печать осуществляется посредством CUPS (Common Unix Printing System).

Для записи CD/DVD можно использовать программы cdio (входит в поставку OpenBSD), а также cdrtools и dvd+rw-tools, для работы с которыми более года назад был портирован графический frontend TkDVD, упрощающий запись дисков. (Прим. ред.: Кстати, с подробностями о записи CD/DVD с помощью указанных консольных утилит можно ознакомиться в предпоследней статье этого выпуска «Open Source».) Графических почтовых клиентов в OpenBSD довольно много: Thunderbird, Sylpheed, Claws Mail, Evolution. Впрочем, как и браузеров: Firefox, Midori, Eriphany, Galeon, Konqueror... Из удобных torrent-клиентов можно отметить Transmission.

Кстати, совсем недавно прошла маленькая революция: общими усилиями был наконец-то портирован редактор векторной графики Inkscape (версия 0.46). А вот GIMP был в составе портов OpenBSD практически всегда, так что порисовать удастся вволю. Для 3D-разработчиков – Blender тоже присутствует в портах, хотя высокой производительности на маломощных графических картах при рендеринге добиться не получится.

Из системных программ обычно нуж-

ны файловые менеджеры: Thunar из состава XFCE и старый добрый mc (Midnight Commander) будут хорошим дополнением к рабочей системе. MPlayer для просмотра видео постоянно обновляется в дереве портов (на данный момент актуальна версия 1.0rc2) и доступен для установки. Аудиоплееров же немного, но работают они довольно стабильно; из известных – MPD (с различными GUI вроде GMPD и Sonata) и Audacious.

И наконец, выбором редакторов для разработки команда OpenBSD не обделила: Geany, gvim, Nedit, SciTE, Tea и многие другие.

Закключение

В этой статье OpenBSD была рассмотрена в не самом привычном для нее ракурсе – в качестве desktop-ориентированного окружения. Как указано выше, количество программ с каждым днём увеличивается, и кто знает, возможно, через пару лет OpenBSD ни в чём не будет уступать, скажем, FreeBSD в «дружелюбности» и при этом сохранит качество кода, параноидальную безопасность, не станет Linux-подобной BSD-системой.

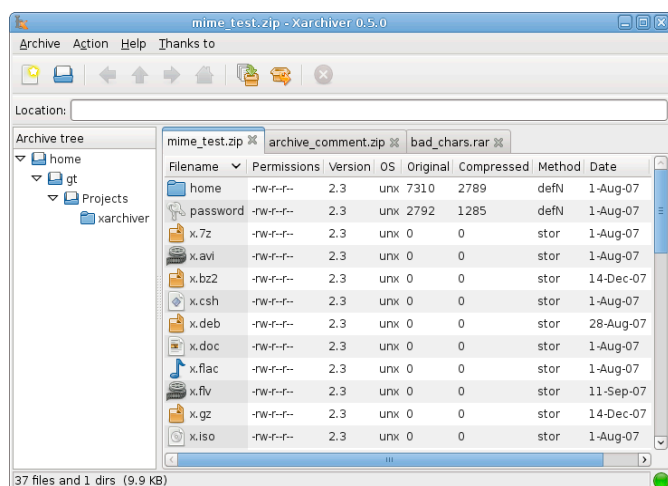
Влад Глаголев
(stealth@sourcemage.org)

FOSS Review 011

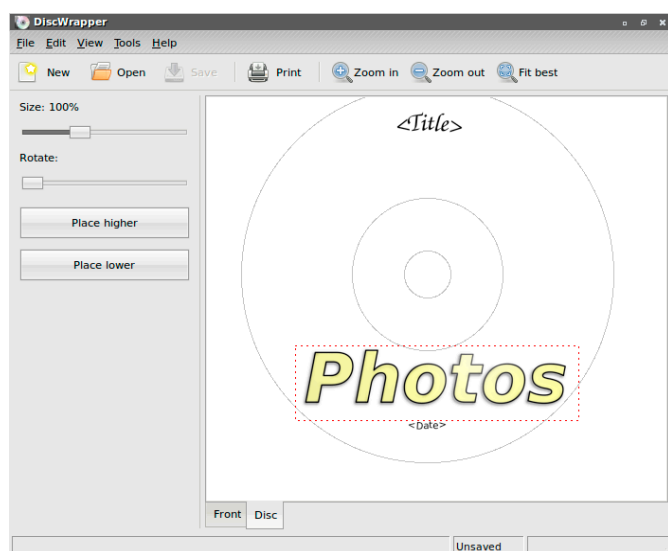
Xarchiver

- ✓ **Версия:** 0.5.2.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 180 Кб (tar.bz2).
- ✓ **Сайт:** <http://xarchiver.xfce.org>.

Некоторые из читателей уже могли догадаться, что я особенно трепетно отношусь к автономным приложениям, которым для работы не требуется огромных библиотек и дополнительных программ. Практика показывает, что они работают быстрее своих аналогов. Пусть это лишь тенденция, но она сработала и на сей раз, когда объектом моего внимания стала программа для работы с архивами – Xarchiver. Несмотря на то что она является частью проекта среды рабочего стола (Xfce), libxfce для ее функционирования не требуется. Xarchiver поддерживает все популярные типы архивов: 7-zip, arj, bzip2, gzip, rar, lha, iso, lzma, lzop, deb, rpm, tar и zip. Обеспечена и работа с архивами, защищенными паролем, причем автоматическая – для arj, zip и rar. Из rar, 7-zip и zip можно создавать самораспаковывающиеся бинарные файлы. Xarchiver приятно радует простотой интерфейса и удобством работы. В частности, это обусловлено поддержкой Drag & Drop и вкладок. Благодаря последнему в одном окне Xarchiver можно одновременно работать с несколькими архивами. Другое удобство – Archive tree – дерево файловой сис-



Xarchiver



DiscWrapper

темы, которое позволяет работать с файлами на диске и открывать архивы прямо из окна программы. Кроме того, с Xarchiver можно работать и в командной строке.

DiscWrapper

- ✓ **Версия:** 1.0.1.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 650 Кб (tar.gz).
- ✓ **Сайт:** <http://sourceforge.net/projects/discwrapper>.

DiscWrapper пригодится тем, кто регулярно записывает CD/DVD-диски. Программа рисует обложки и наклейки для таких компакт-дисков. В ней представлено несколько шаблонов, а также есть возможность нарисовать свой. Обложку можно создать как для CD-коробок, так и для DVD-боксов (и обычных, и тонких – slim). В «документах» DiscWrapper допускается текст и изображения.

Glista

- ✓ **Версия:** 0.4.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 134 Кб (tar.gz).
- ✓ **Сайт:** <http://prematureoptimization.org/glista>.

Glista – это вовсе не приложение-паразит, обитающее в полости животного или человека, как можно подумать, увидев его название. Это программа для организации задач с очень простым интерфейсом. В главном окне расположен сам список задач, разделенный (по усмотрению пользователя) на разные категории. Для удобства каждую категорию можно свернуть. Если поставите галочку на выполненной задаче, то она окрасится в серый цвет, а текст станет зачеркнутым. Каждой задаче можно задать свое описание. Задачи быстро добавляются через нижнее поле. Glista удобно держать в системном трее



Glista



Music Applet

в виде иконки. По моему мнению, программе не хватает только синхронизации с online-менеджерами задач (например, с RememberTheMilk.com). А в целом – отличное, легкое приложение с большим потенциалом и забавным для русского человека названием.

Music Applet

- ✓ **Версия:** 2.4.2.
- ✓ **Лицензия:** GNU GPL.
- ✓ **Размер:** 468 Кб (tar.gz).
- ✓ **Сайт:** <http://www.kuliniewicz.org/music-applet>.

В Linux существует великое множество музыкальных плееров, которыми зачастую надо так или иначе управлять. Под последним понимается переключение треков, пауза, регулировка громкости, а по возможности – еще и получение информации о том, какая песня сейчас играет, сколько осталось до конца ее воспроизведения, какой у песни рейтинг. И в идеале – все эти возможности должны быть легкодоступны без необходимости держать окно плеера постоянно открытым на рабочем пространстве. Видимо, примерно так и размышляли авторы этого приложе-

ния. Music Applet – небольшой апплет для панели GNOME, с помощью которого можно управлять музыкальным плеером, причем как через D-Bus, так и с помощью команд самого плеера. В нем реализованы все те возможности, о которых давно могли мечтать некоторые меломаны.

Роман Комков
(r.komkov@gmail.com)

Ubuntu для новичков. Часть 5: управление ПО

Эта пятая часть из цикла статей «Ubuntu для новичков». Ранее был сделан общий обзор дистрибутива, рассмотрены вопросы выбора ISO-образа, установки, использования консоли и настройки репозитория. Перейдем к управлению программным обеспечением. В статье, как и в предыдущей, будут использоваться два подхода: консольный и с применением графических утилит.

Установка/удаление

Манипулировать программным обеспечением в Ubuntu очень легко. И после того как вы получите навыки в управлении ПО, отказаться от привычки использовать пакетный менеджер будет тяжело. Итак, рассмотрим два способа.

Первый способ

В пункте меню «Приложения» самой последней строчкой можно найти программу

для управления ПО – «Установка/Удаление». Она вызывает утилиту `gnome-app-install`, которая в свою очередь работает на основе APT (Advanced Package Tool). Разобраться в этой программе не составляет труда: в левом столбце доступное ПО разделено по категориям, справа в верхней части – название программ, а снизу – их описания (на английском языке). После того как нужное ПО выбрано, нажмите «Применить изменения» – будут скачаны и установлены все необходимые пакеты с программами. Если устанавливались приложения с графическим интерфейсом, их можно найти в меню «Приложения» (в тех группах, из которых они были установлены): например, если установлен аудиоплеер Amarok, то искать его следует в категории «Аудио и видео».

Для того чтобы быстро найти необходимую программу по названию, в самой верхней строчке есть форма поис-

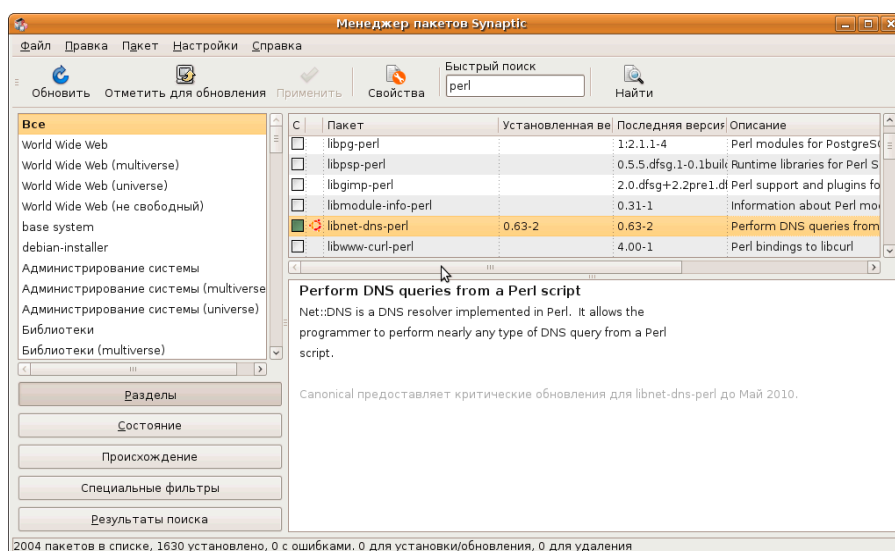
ка. Рядом с ней – выбор возможных вариантов ПО для установки: поддерживаемое, Open Source (полностью открытое) и др. (**Прим. ред.:** В общем случае лучше раз и навсегда выбрать там «Все доступные приложения».)

Более «продвинутым» вариантом управления ПО является утилита Synaptic («Система → Администрирование → Менеджер пакетов Synaptic»; запуск требует прав администратора компьютера). С ее помощью можно обновлять пакеты, удалять их и переустанавливать. Кроме того, вашему вниманию предоставляется такая полезная информация, как версия программы и ее зависимости. Здесь намного шире возможности фильтрации и охвата пакетов: Synaptic позволяет сортировать их по репозиторию, состоянию (установлены, не установлены, можно автоматически удалить...) и другим параметрам. Кроме того, в программе предусмотрены настройки прокси-сервера и используемых репозитория (для последнего вызывается уже знакомое приложение «Источники приложений»), в ней поддерживаются сценарии и ведется история действий. Управлять программами с ее помощью тоже несложно: надо найти пакет, отметить его для установки/удаления (или другой необходимой функции) и нажать «Применить» в панели инструментов.

Второй способ

Все описанные операции можно выполнить и из консоли. Для этого существуют такие приложения, как `apt` и `aptitude`. Точнее, APT – это целый набор утилит, на котором, кстати, основывается и Synaptic, и Aptitude. Первая, наиболее часто используемая и важная утилита из набора APT, – это `apt-get`. С ее помощью можно установить, удалить и обновить пакеты. Часть операций была рассмотрена в предыдущей части цикла, напомним:

- ✓ **`sudo apt-get update`** – обновить список репозитория;



Менеджер пакетов Synaptic

☑ **sudo apt-get dist-upgrade** – безопасное обновление всего ПО в системе.

Другие операции тоже интуитивно понятны по синтаксису. Например, чтобы установить Pidgin (клиент обмена мгновенными сообщениями), нужно воспользоваться следующей командой:

```
$ sudo apt-get install pidgin
```

Для его удаления:

```
$ sudo apt-get remove pidgin
```

Для удаления со всеми системными настройками (не пользовательскими, которые лежат в домашнем каталоге):

```
$ sudo apt-get purge pidgin
```

Помимо этого можно воспользоваться дополнительными ключами. Например, чтобы закачать пакеты, но не устанавливать их, существует ключ `-d` (--download-only в полной форме), а для того чтобы каждый раз не отвечать на запросы – ключ `-y` (--yes, --assume-yes). В качестве примера – обновление компонентов системы без запросов на подтверждение:

```
$ sudo apt-get dist-upgrade -y
```

Более подробно описано в официальном руководстве (man apt-get). Для управления кэшированными данными предусмотрена утилита `apt-cache`, которая, в частности, позволяет найти нужный пакет по имени по команде `search`:

```
$ apt-cache search pidgin
```

Получить статистику по пакетам, присутствующим в кэше, можно так:

```
$ apt-cache stats
```

Для подключения репозитория на компакт-диске есть утилита `apt-cdrom`. Строка добавления репозитория на CD выглядит следующим образом:

```
$ sudo apt-cdrom add
```

Вторая утилита, Aptitude, имеет текстовый интерфейс подобный графическому (основанный на ncurses). Верхняя строка – меню. Основная область состоит из двух частей: первая, как и в графических аналогах, отображает пакеты, а нижняя – их описание. Кроме того, управлять ПО через Aptitude можно исключительно через командную строку.

Вот наиболее востребованные команды:

- ☑ **sudo aptitude install pidgin** (на месте `install` могут быть `remove`, `purge`, `reinstall`);
- ☑ **sudo aptitude update**;
- ☑ **sudo aptitude full-upgrade**;
- ☑ **sudo aptitude show pidgin** (информация о пакете).

Информация о других возможностях aptitude доступна в map-странице (man aptitude).

Все вышеперечисленные программы так или иначе используют `dpkg` – менеджер пакетов в Debian и Ubuntu. Он в отличие от остальных не следит за зависимостями – их надо устанавливать вручную. Пользоваться `dpkg` довольно просто. Например, установка пакета из файла `file.deb`:

```
$ sudo dpkg -i file.deb
```

Для удаления пакета используется ключ `-r`. Для установки множества `deb`-пакетов, которые доступны в одном каталоге, можно воспользоваться удобством консоли, что пригодится, например, после распаковки архива с `OpenOffice.org` Pro от компании Инфра-Ресурс:

```
$ sudo dpkg -i *.deb
```

Эта команда установит все `.deb`-пакеты, доступные в текущем каталоге.

Некоторая часть ПО в интернете распространяется в `deb`-пакетах – например, Skype. Его можно установить, скачав необходимый пакет с сайта и либо вызвав из консоли программу `dpkg`, либо воспользовавшись еще одной графической утилитой – GDebi. Процесс установки при использовании последней практически не отличается от установки ПО в Windows-системах. После клика на устанавливаемый пакет потребуется ввести пароль суперпользователя, после чего пакет будет установлен. Если пакету потребуются файлы, которые не установлены в системе, они также будут скачаны и установлены из Сети.

Необходимое ПО

Часть программного обеспечения из-за лицензионных ограничений не может свободно входить в состав дистрибутива Ubuntu. Среди них – Flash-плагин для браузера, mp3-кодеки и прочее. Однако при желании пользователя он может сам установить необходимые компоненты из репозитория. Для облегчения поиска и установки требуемых компонентов

в Ubuntu ввели метапакет `ubuntu-restricted-extras`. Для Kubuntu он называется `kubuntu-restricted-extras`, а для Xubuntu – `xubuntu-restricted-extras`. Сам метапакет не содержит никаких программ, однако в его зависимостях указаны требующиеся пакеты, которые будут скачаны и установлены. После установки метапакет не нужен – его можно удалить вручную (он располагается в каталоге `/var/cache/apt/archive`, где находятся все другие скачанные из Интернета пакеты). А лучше для этого воспользоваться одной из уже известных команд, что удаляет все пакеты в кэше:

```
$ sudo apt-get clean
```

или:

```
$ sudo aptitude clean
```

По тем же (лицензионным) причинам часть программ не была включена в четыре основных репозитория Ubuntu. Их можно установить из так называемого «Партнерского» репозитория. Это различные наборы кодеков, Орега и другое ПО. Для его установки выберите «Система → Администрирование → Источники приложений». На вкладке «ПО сторонних производителей» нужно галкой отметить строку, в конце которой написано «partner». Если требуется, можно подключить и репозиторий с исходными кодами, и любой дополнительный. Например, чтобы добавить ежедневно обновляемый репозиторий с KDE 4, достаточно нажать на кнопку «Добавить» и в «Строку APT» вписать следующее:

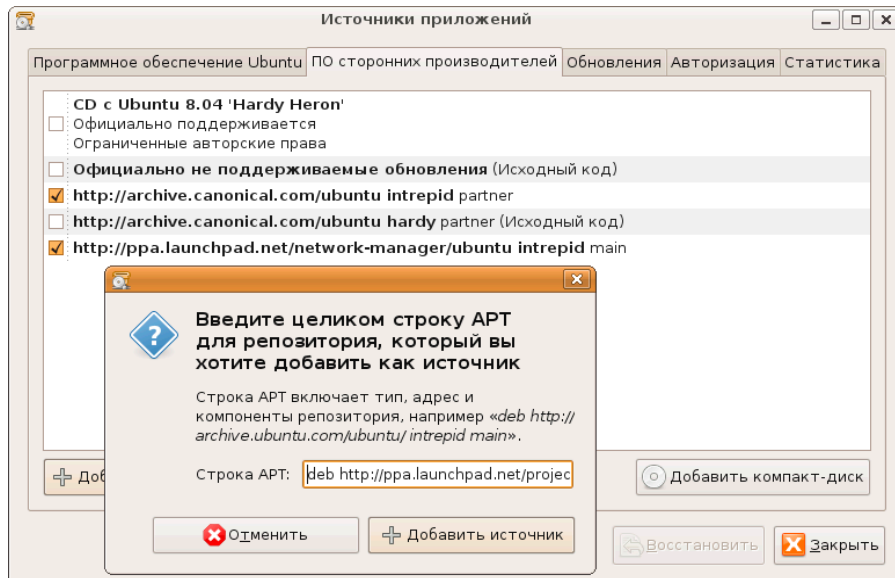
```
deb http://ppa.launchpad.net/~project-neon/ubuntu intrepid main
```

Теперь программу можно закрывать, за чем последует предложение обновить список репозитория.

Все эти операции можно выполнить правкой файла `/etc/apt/sources.list`, о чем было написано в предыдущей статье.

Драйверы устройств

Проприетарные (сделанные компанией-производителем устройства и выпущенные под собственнической лицензией) драйверы по лицензионным ограничениям не могут свободно распространяться на установочных дисках, однако при желании пользователь может их установить. В Ubuntu 8.10 была обновлена утилита инсталляции таких драйверов (при их наличии возникает соответствующее уведомление). Драйверы на видеокарту можно установить с помощью программы `Envyng` (она доступна в основном репозитории).



Источники приложений

Выбор ПО

Выбор ПО для GNU/Linux обычно не ограничивается одним вариантом: достаточно много программ есть для решения большинства повседневных задач. Поэтому я решил привести короткий список используемого в Ubuntu ПО и его основных альтернатив.

Клиент обмена мгновенными сообщениями. По умолчанию это Pidgin, который поддерживает большинство современных протоколов, однако может показаться непривычным пользователям Windows-систем. Его альтернативы – это Kopete (стандартный для Kubuntu мессенджер из KDE), с недавних пор поддерживающий X-статусы ICQ, а также qutim, наполняющий QIP.

Веб-браузер по умолчанию – не нуждающийся в представлении Firefox. Любители Opera могут установить ее из партнерского репозитория, а в Kubuntu используется браузер Konqueror.

Почтовый клиент: Evolution, стандартный для среды GNOME, можно заменить на KMail из KDE или Thunderbird от Mozilla.

Transmission – новый BitTorrent-клиент, у которого есть более известные аналоги вроде Vuze (ранее известен как Azureus) на Java и KTorrent из KDE, а также Deluge.

Для работы в ed2k и Kad по умолчанию установленной программы в Ubuntu не предусмотрено, однако aMule, аналог eMule, легко проинсталлировать са-

мостоятельно. Аналогичная ситуация с LinuxDC++ для работы в DirectConnect (DC).

Помимо стандартного офисного пакета OpenOffice.org доступны приложения Gnumeric и Abiword (для работы с таблицами и текстами соответственно). Они не предоставляют аналогичных возможностей, но не требуют много места и вычислительных ресурсов. Другой альтернативой выступает стандартный для KDE офис KOffice.

Растровая графика в Ubuntu обрабатывается кроссплатформенным редактором GIMP – его возможностей вполне хватает обычному пользователю. При желании можно попробовать Krita из офисного пакета KOffice.

Альтернативами для программы просмотра изображений F-Spot могут выступить GQview и Gwenview (из состава KDE).

Для записи дисков используется Brasero. Заменить его можно стандартной утилитой из Kubuntu – k3b, а также более простой Xfburn.

Достойная замена мультимедийному плееру Totem – SMplayer, фронтенд к MPlayer, а также VLC со своим набором кодеков.

Для прослушивания музыки отличной альтернативой будет Amarok (по умолчанию в Kubuntu) и Audacious – для любителей минимализма и WinAMP.

Никита Лялин
(tinman321@gmail.com)

Запись дисков в Linux-консоли

Немного истории

Не открою Америку, заявив, что большинство Linux-программ для записи CD и DVD являются просто графическими оболочками для ряда консольных утилит. Исключение составляет, пожалуй, только версия Nero для Linux.

Долгое время утилиты записи на CD/DVD выпускались в пакете cdrtools, разрабатываемом Йоргом Шиллингом (Joerg Schilling). В 2006 году Шиллинг решил сменить лицензию с GPL на CDDL (Common Development and Distribution License), несовместимую с GPL. Точнее, сначала была изменена лицензия нескольких make-файлов. И этим сразу озаботились люди из Debian, которые не могли создавать пакеты, где GPL-код зависим от не-GPL-кода. И люди из Debian после долгих дебатов с Шиллингом (насколько я знаю, во время которых код cdrtools стал уже на 80% под CDDL), сделали ответвление от оригинального проекта и назвали его cdrkit. Утилиты, входящие в его состав, получили другие названия, однако параметры командных строк сохранились прежними. Так, программа cdrecord стала именова-

ться wodim (сокращение от write optical disk media), cdda2wav превратилась в icedax (incredible digital audio extractor), а mkisofs переименовали в genisoimage (generate ISO image).

Создание образа

Чтобы записать информацию на болванку, нужно сначала создать образ записываемого диска. Есть несколько способов сделать это. Первый применим, когда вы хотите скопировать существующий диск. Поместите диск в привод, смонтируйте диск, перейдите в каталог, где намереваетесь создать образ, и дайте там примерно такую команду:

```
$ dd if=/dev/cdrom of=mycd.iso
```

Используемое здесь «dd» – это название стандартной программы, которая найдется в любом дистрибутиве Linux. Она копирует содержимое одного файла в другой, а поскольку в UNIX любое устройство есть файл, то dd можно использовать для создания образов любых дисков: от доисторических дискет до DVD (и более того, dd способен записать образ на тот же флоппи).

Разберем параметры:

- ☑ **if** – означает input file, т.е. «входной файл»,
- ☑ **of** – output file (выходной файл).

☑ **/dev/cdrom** – это стандартная ссылка на устройство CD/DVD-привода, вместо нее можно подставить и настоящее имя устройства (например, /dev/hdd).

В параметре **of** указывается имя файла, под которым будет создан образ с помещенного в привод диска. Расширение **iso** – дань традиции. По умолчанию **dd** читает с устройства блоками по 512 байт. Это значение можно изменить параметром **bs** (например, **bs=1024**). Для принудительного копирования даже после ошибок чтения можно добавить ключ **noerror**. Создание копии при помощи **dd** может не сработать при использовании двухслойных DVD. В остальных случаях обычно работает.

В результате на выходе получается точная копия диска. Но таким образом не получится скопировать ни аудиодиск, ни какой-нибудь защищенный диск с игрой (последний переписывается, но распознаваться как «правильный», лицензионный, не будет). Кроме того, **dd** не показывает данные о процессе выполняемого копирования – для этого можно лишь наблюдать за светодиодом привода и слушать гудение вращающегося диска (**Прим. ред.:** Или проверять изменения в размере записываемого файла-образа.).

Есть и другой способ создания образа диска. Он особенно актуален для случаев, когда нужно записать на диск каталог с файлами или множество таких каталогов. Поможет замечательная программа **genisoimage**. Она умеет делать образ диска из директории вот так:

```
$ genisoimage -f -v -J -o mycd.iso /mnt/somedisk/temp
```

В этом примере в текущем каталоге создается образ каталога /mnt/somedisk/temp и записывается в файл mycd.iso. Параметр **-v** указывает, что **genisoimage** должна информировать о ходе процесса в консоль, **-J** задает формат файловой системы (Joliet), а **-f** указывает **genisoimage** следовать по символическим ссылкам, помещенным в копируемый каталог.

Итак, чтобы сделать «виртуальный проект» диска, надо создать некий временный каталог и поместить в него символические ссылки на директории или файлы, которые вы желаете видеть в своем проекте. В чистой консоли это делать весьма скучно, поэтому воспользуемся для создания этих ссылок консольным файловым менеджером Midnight Commander. Запустим **mc**, на одной панели перейдем в каталог, где «проектируем» диск, а на другой панели – туда, где находится файл или каталог, символическую ссылку на который вы хотите добавить в проект. Помещаем курсор на этот файл или каталог, нажимаем **<Ctrl> + <X>**, отпускаем. Теперь нажимаем клавишу **<S>**. Появляется диалоговое окно с уже заполненными полями (на что делать ссылку и куда ее помещать). Подтверждаем нажатием на **<Enter>**. Все, готово. Таким образом наполняем проект файлами и каталогами. Ничуть не медленнее, чем в **k3b**.

Теперь остается передать каталог проекта упомянутой утилите **genisoimage** и создать образ. Но хватит ли приведенных мною выше параметров на все случаи жизни? Конечно же, нет. Вникнем в некоторые тонкости. Linux может откровенно плохо читать с носителя большие видеофайлы, если на диске нет файловой системы UDF (Universal Disc Format). Напомню, что на болванке могут соседствовать разные файловые системы. По умолчанию это ISO-9660: она понимается даже в MS-DOS и «железные» DVD-плееры ее тоже очень любят. Есть также Joliet – надстройка над ISO-9660, разрешающая длинные имена файлов, причем даже в юникоде. И Linux, и Windows хорошо понимают эту файловую систему, и многие «железные» плееры тоже.

Но вернемся к UDF: ее поддержку надо обязательно вклю-

чать, если вы записываете «стандартный» DVD с видео, т.е. с четкой структурой каталогов VIDEO_TS (с фильмом, состоящим из VOB-файлов) и иногда AUDIO_TS. Впрочем, лично я использую UDF при записи любых DVD. Есть два способа включить UDF для образа диска. Первый – использовать ключ **-udf**, а второй – это «ориентированный» на DVD Video параметр **-dvd-video**. В последнем случае надо, чтобы имена файлов и каталогов (VIDEO_TS и прочих) были в верхнем регистре. При создании образа файлы (их содержимое) будут отсортированы и расположены в образе диска. Сортировка не произойдет, если все названия не в верхнем регистре. Все это должно «способствовать» чтению подключаемых дисков на всех «железных» DVD-плеерах. Однако я эту сортировку не включаю, и по имеющейся у меня информации записанные без нее диски с DVD Video все равно нормально читаются на «железных» плеерах.

Другое расширение к ISO-9660 – это Rock Ridge. Оно по умолчанию включено в **k3b** и дает возможность использовать более длинные, чем в Joliet, имена файлов, а также большую глубину вложенности каталогов и атрибуты файлов UNIX (обратите на это особое внимание). Rock Ridge включается параметром **-r**, а название этого расширения взято из фильма Мэла Брукса «Сверкающие седла» (там был городок Рок Ридж). Особой нужды во включении Rock Ridge я не вижу. К слову, записанные мною в **k3b** (с включенным Rock Ridge) диски с DVD Video читаются только с правами пользователя **root**.

Запись образа

Когда образ подготовлен, надо его записать на болванку. Здесь поможет программа **wodim**. В простейшем случае ее запуск выглядит так:

```
# wodim dev=/dev/hdc -eject -v mycd.iso
```

Очевидно, что вместо /dev/hdc вы можете подставить другое устройство, а mycd.iso – это название файла с образом. Параметр **-eject** выдвигает каретку привода после окончания записи. У **wodim** есть ряд других интересных параметров:

- ☑ **-dummy** – симуляция прожига с выключенным лазером. Полезно, чтобы определить, сколько времени может занять запись.
- ☑ **-multi** – многосессионный диск.
- ☑ **-fix** – фиксация диска (без записи).
- ☑ **speed=значение** – задает скорость записи.
- ☑ **blank=режим** – очистка содержимого CD-RW. Доступные режимы:
 - **all** – полная очистка, долго;
 - **fast** – быстро;
 - **session** – очистить последнюю сессию.
- ☑ **-format** – форматировать болванку CD/DVD-RW.
- ☑ **-overburn** – включить возможность записи больше, чем стандартный объем диска. В этом случае, как правило, необходимо также использовать режим записи SAO (-sao).

На этом можно было бы и закончить статью, но я не коснулся еще одной важной темы – записи обычных звуковых CD.

Запись музыкальных дисков

Перед записью аудиодиска необходимо подготовить звуковые дорожки. Это обычные WAV-файлы с определенными параметрами: два канала (стерео), знаковый **integer** – 16 бит, частота оцифровки – 44100 герц, PCM (Pulse-code modulation). Грубо говоря, это самый стандартный WAV-файл, который можно себе представить. Поскольку зачастую такие файлы и появляются

в результате их копирования с аудиодисков, сначала рассмотрим эту, «обратную» запись, операцию. Если вы хотите скопировать audioCD, получив музыкальные файлы с оригинальным качеством, вам поможет утилита `cdparanoia`. Вообще она знаменита своим умением считывать музыку даже с крайне поцарапанных дисков. Это умение занимает очень много времени, поэтому его можно отключать ключом `Z`. Тогда запуск программы выглядит примерно так:

```
$ cdparanoia -BZ
```

Второй используемый здесь параметр – `B`, указывает, что каждую дорожку надо сохранять в отдельный файл. Все эти файлы сохраняются в текущий каталог. Если вы желаете скопировать только некоторые дорожки, их диапазон можно задать следующим образом:

```
$ cdparanoia 2-6 -BZ
```

В примере копируются звуковые дорожки со второй по шестую включительно.

Теперь – вновь к записи аудиодисков. Звуковой CD можно записать в двух режимах: TAO (track at once) и DAO (disk at once). TAO добавляет между дорожками паузы длиной в 2 секунды, а кроме того, после записи каждой дорожки выключает лазер. DAO – более новый и гибкий режим. В нем между дорожками можно записывать что угодно (не только паузы, но и, например, звуковые заметки), благодаря чему становится возможной даже запись дорожки, которая будет доступна к воспроизведению, однако не пронумерована – слушатель найдет ее только перемоткой. (Прим. ред.: Это и есть так называемые hidden tracks, зачастую помещаемые музыкальными исполнителями в конец альбома.) В такие диски можно внедрять и текст (CD-Text). Кроме того, DAO-режим – единственный способ записать диск без пауз между песнями. Еще один режим, SAO (session at once), напоминает DAO, но поддерживает многосессионность. Для выбора режима записи в `wodim` существуют переключатели:

- ☒ -tao
- ☒ -dao
- ☒ -sao

Чтобы записать WAV-файлы на болванку, нужно выполнить примерно такую команду (из каталога, где они находятся):

```
# wodim dev=/dev/hdc -tao -eject speed=4 -pad -audio *.wav
```

Рассмотрим параметры. Новых для нас – всего два:

- ☒ **-pad**. Размер дорожки (а значит и исходного звукового файла) должен быть кратен 2352 байтам. Иначе `wodim` откажется его записывать. Этот параметр (`-pad`) заставляет `wodim` автоматически дополнять нулями размер/длину дорожки до кратности 2352.
- ☒ **-audio** – указывает, что диск записывается в формате CD-DA (Red Book).

Заключение (про growisofs)

Многие задачи, описанные в этой статье, решаются и другими способами. Например, для копирования музыкального диска в WAV-файлы есть утилита `icedax`, а копировать диск с данными можно командой `readom`. Совершенно за бортом осталась консольная утилита `growisofs`, служащая для записи данных на DVD, поэтому концовку я решил посвятить именно ей. `Growisofs` – это подобие надстройки над `wodim` и `genisoimage`. Программа понимает параметры от `genisoimage`. Вот как с ее помощью можно записать диск:

```
# growisofs -Z /dev/dvd -J -f -udf /foo/bar
```

«Родной» параметр тут один `-Z`, который указывает, что записывается первая сессия. Для следующей надо указывать ключ `-M`. Других собственных параметров у `growisofs` немного. Из основных назову лишь `-overburn`, `-speed`=скорость и `-dvd-compat` (закрывает диск DVD+R и DVD-R: его нельзя будет дополнить, зато он хорошо читается на «железных» плеерах).

Финализация многосессионного диска с `growisofs`:

```
# growisofs -M /dev/dvd=/dev/zero
```

Запись подготовленного ранее образа диска:

```
# growisofs -dvd-compat -Z /dev/dvd=mycd.iso
```

В ходе работы `growisofs` образ диска создается на лету и через канал (pipe) отправляется на программу записи.

В завершение дам совет, значение которого трудно недооценить: Работая с `genisoimage` и `growisofs`, не забывайте о параметре `-f`!

Петр Семилетов
(tea@list.ru)

Использование интегрированной среды NetBeans для разработки приложений на C/C++ и Java. Часть 2

Прим. ред.: Первая часть этой статьи была опубликована в предыдущем выпуске «Open Source» – #034. В ней рассматривались основные инструменты NetBeans для разработки на Java.

Тестирование

Тестирование – это важный этап создания программного обеспечения, который не остался без внимания разработчиков среды. Для тестирования своих

программ на языке Java вы можете использовать систему модульного тестирования JUnit.

Если посмотреть в дерево проектов, можно заметить, что в каждом проекте присутствует каталог Unit Test Packages, куда помещаются все модульные тесты. Внутри каталога можно создавать пакеты и размещать в них тесты. Если не создано ни одного пакета, все тесты помещаются в безымянный пакет.

Создавать тесты в среде просто: достаточно выбрать класс в дереве проектов, вызвать контекстное меню и в подменю Tools выбрать команду Create JUnit Tests. После этого среда создаст файл с реализованными по умолчанию методами для тестирования. Остается лишь наполнить эти методы функциональностью.

Для выполнения написанных вами тестов в главном меню Run предусмотрен

пункт Test, который запускает тесты и выводит информацию о процессе тестирования в окно вывода (см. **рис. 1**), а результаты тестирования – в специальное окно JUnit Test Results (см. **рис. 2**).

Отладка

После того как код написан и протестирован, может потребоваться отладка. Среда NetBeans имеет отличную интеграцию с отладчиком jdb из стандартного пакета Java SDK.

Все консольные команды отладчика реализованы в соответствующем модуле. Прямо в редакторе кода можно не только установить точки останова отладчика,

но и задать условие срабатывания для точки. Например, в ситуации, когда определенное значение переменной приводит к краху программы, вы можете установить точку останова с условием останова, когда параметр метода равен значению, вызывающему сбой программы, а не ожидать в пошаговом режиме появления этих данных.

В нижней части отладчика отображается окно с вкладками. Они позволяют просмотреть локальные переменные, заданные переменные (может содержать выражение), стек вызовов, список потоков в приложении и другие параметры.

Отладчик предоставляет стандартный

набор команд для пошаговой отладки приложения (см. **рис. 3**).

Профайлер

Для исследования производительности приложения и выявления узких мест в среду встроен профайлер. Он запускается из меню Run. При выборе команды вам будет предложен выбор отслеживаемых характеристик процесса.

На выбор предложены три режима:

- ☒ анализ производительности и использования времени процессора,
- ☒ использование памяти,
- ☒ отслеживание данных в реальном времени.

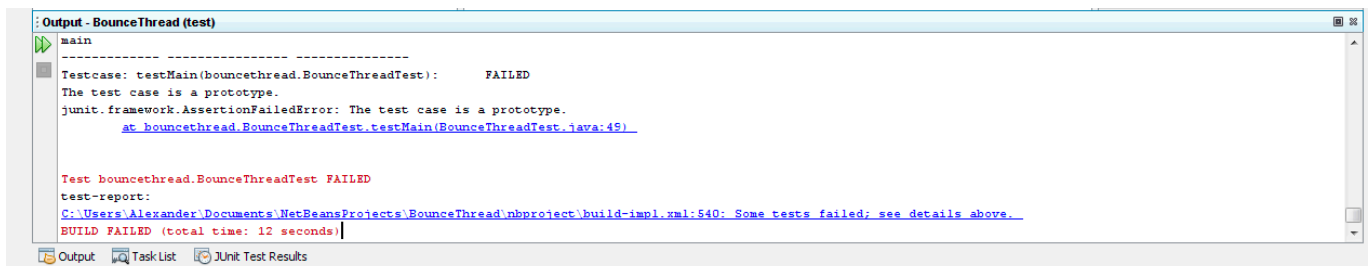


Рисунок 1. Окно вывода сообщений

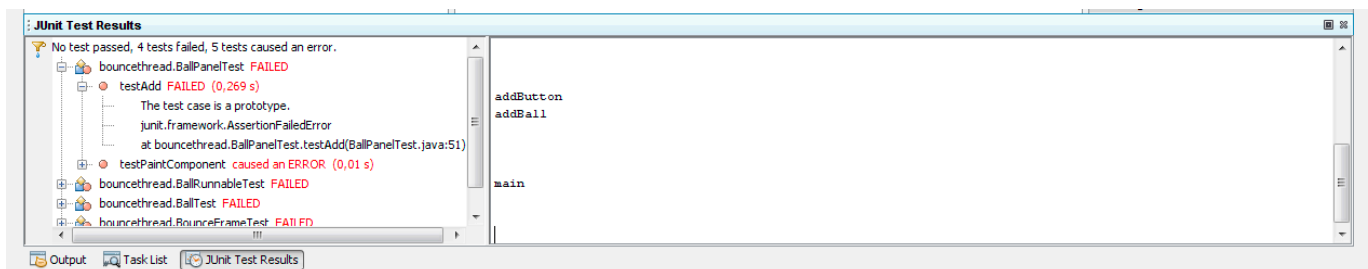


Рисунок 2. Окно результатов тестирования

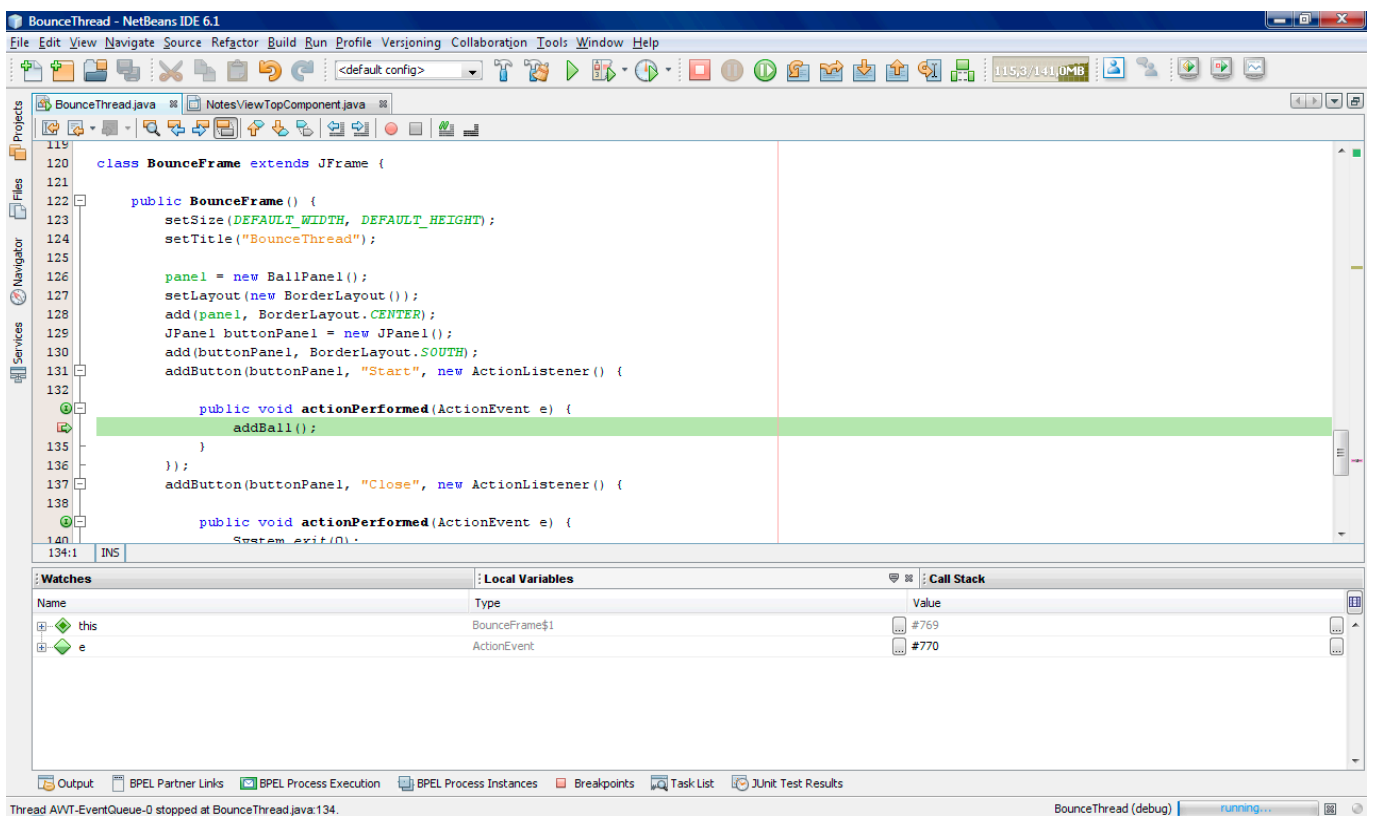


Рисунок 3. Режим отладки программы

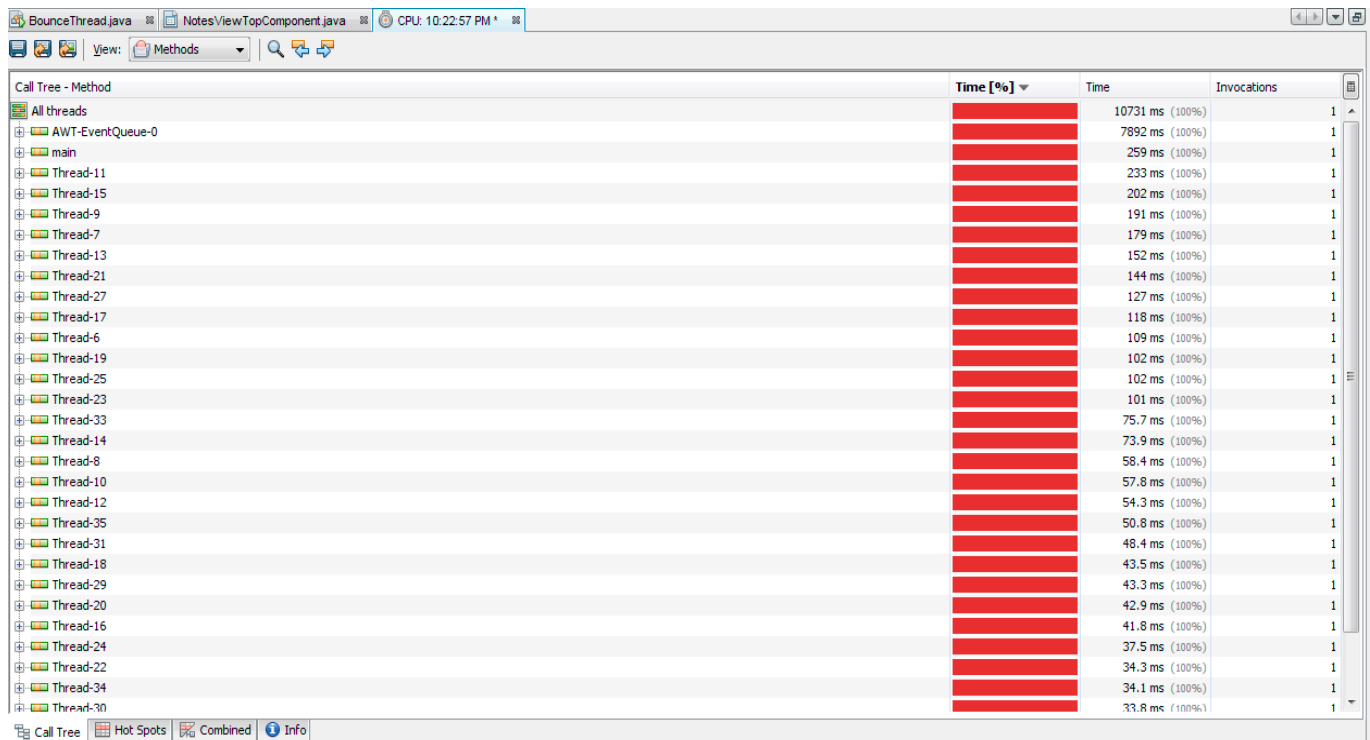


Рисунок 4. Окно результатов профайлера

После выбора режима среда запустит приложение «под наблюдением» профайлера. Далее следует выполнить различные действия с программой, запустить загрузку данных, обработку и анализ данных. Как только работа приложения будет завершена, среда спросит, хотите ли вы увидеть собранные результаты. В случае согласия появится окно результатов профайлера (см. **рис. 4**). В нем можно увидеть, какие потоки приложения запускались, какие методы были вызваны в потоке, сколько времени было затрачено на исполне-

ние метода. Аналогичная картина доступна и для использования памяти.

Еще один метод профилирования, предоставляемый средой, – это мониторинг процесса с помощью JConsole, которая позволяет отслеживать состояние процесса в режиме реального времени (см. **рис. 5**).

Для этого необходимо из среды запустить JConsole и в появившемся окне диалога выбрать приложение. Тогда консоль подключится к процессу приложения и начнет мониторинг.

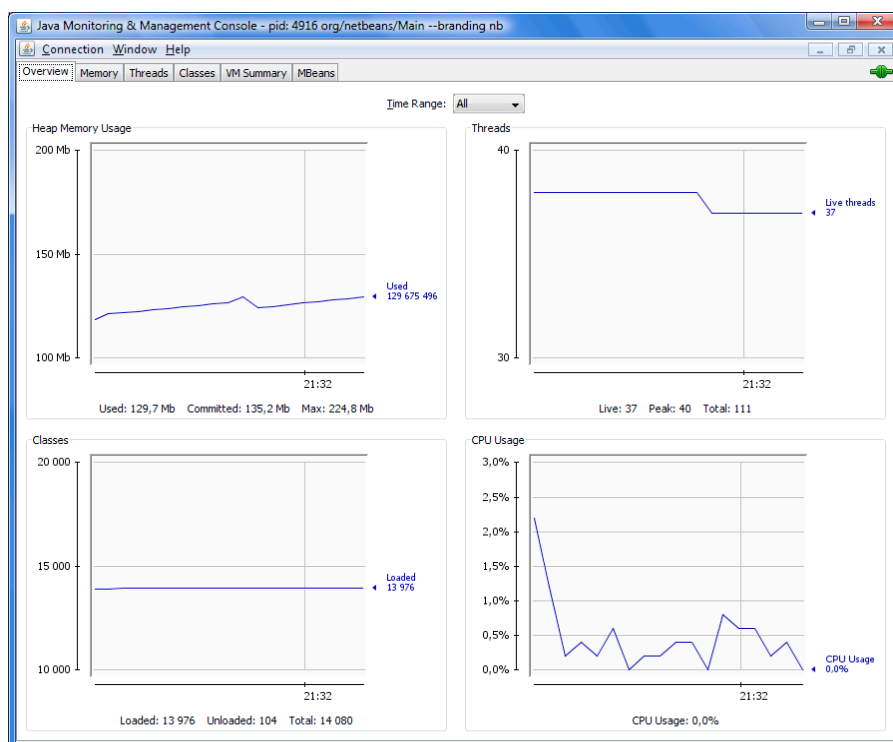


Рисунок 5. Мониторинг с помощью JConsole

С и C++

Поддержка языков С и С++ в NetBeans появилась не очень давно – с 6-й версии. Она позволяет разрабатывать и отлаживать программы. В среде нет встроенного компилятора и отладчика – NetBeans всецело полагается на те, что установлены в системе. Если же их несколько, среда позволит зарегистрировать все и выбирать, какой из них будет использоваться для того или иного проекта. При этом все взаимодействие реализовано очень хорошо, и нарекания может пока вызывать только поддержка кросс-компиляторов. Впрочем, судя по «дорожной карте» (Roadmap) проекта, скоро этот недостаток будет исправлен. Вторым и последним минусом в поддержке С/С++ можно считать использование нестандартной для UNIX и Linux системы проектных файлов: для управления проектом среда использует свою собственную структуру файлов сборки. В остальном NetBeans оставляет приятные впечатления по поддержке языков С и С++.

Так как среда состоит из набора модулей, то все возможности, доступные для проектов на Java, есть и для С/С++: это и системы управления версиями, и окно отладчика, и многое другое. В дополнение к навигатору появилась вкладка классов, которая не работает с Java-классами.

Редактор

При сравнении с редактором для Java может показаться, что набор функций редактора для С и С++ беднее, но, как показывает практика, ряд функций просто не-

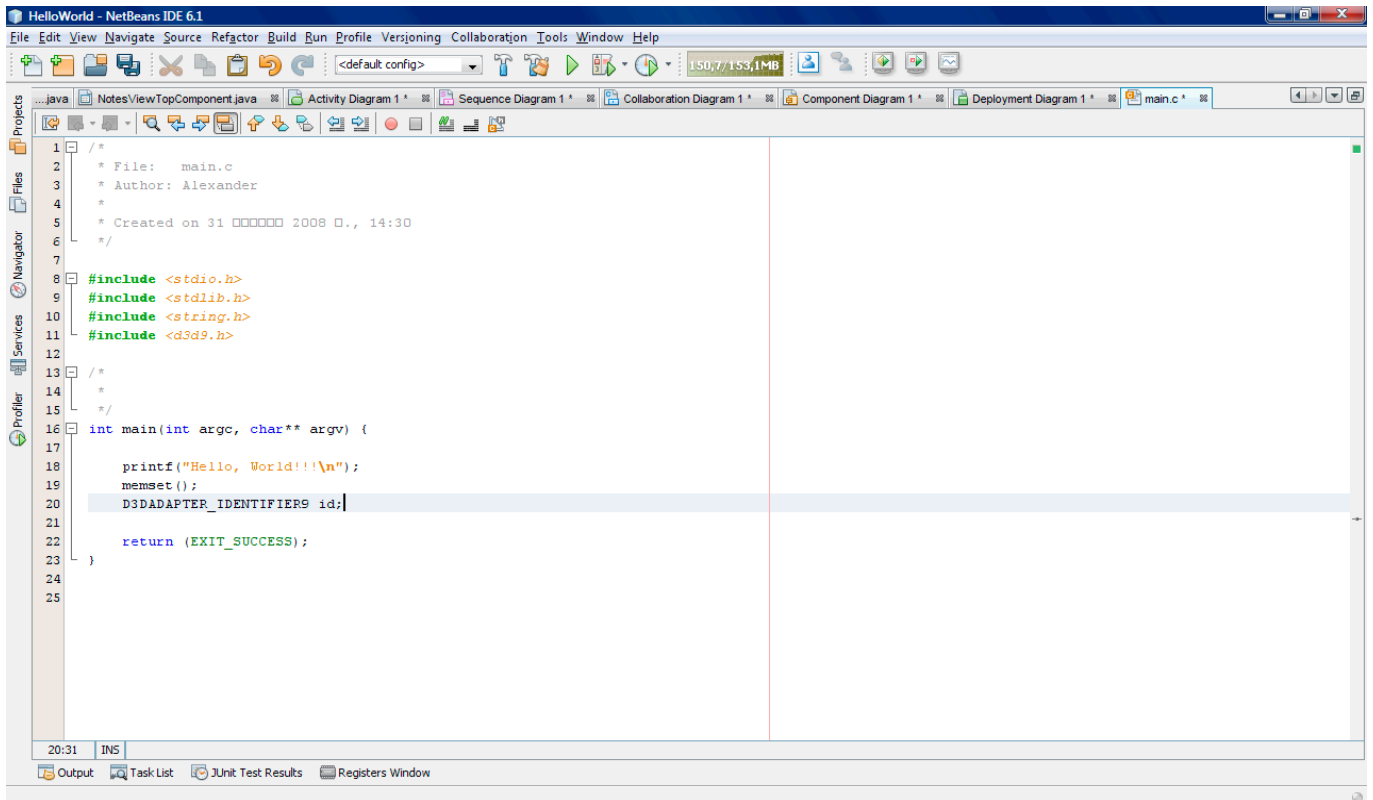


Рисунок 6. Редактор кода на C/C++

применим и/или не нужен для этих языков. А все основные функции сохранились: автоматическое дополнение, динамическая справка по функциям, удобная работа с комментариями, свертка блоков кода, форматирование, контекстный поиск и даже часть функций рефакторинга (см. **рис. 6**).

Панель классов

В этой панели доступны все классы, реализованные как в открытом файле, так и во включенных в него заголовочных файлах. С ее помощью можно легко перейти к любому нужному классу или методу (см. **рис. 7**).

UML-редактор

Для поддержки полного цикла разработки программного обеспечения в среду встроен модуль для создания диаграмм на языке UML. Как и вся среда, модуль «заточен» под язык Java и позволяет создавать шаблоны кода (заготовки классов и интерфейсов) на языке Java.

При создании диаграмм модуль предлагает возможность выбора из библиотеки шаблонов (паттернов) проектирования из набора общих и специфичных для Java шаблонов. Модуль позволяет создавать стандартные для языка UML диаграммы. Все они рассмотрены ниже.

Диаграмма классов

Диаграмма классов¹ позволяет создать классы, описывающие основные дейст-

вующие классы будущей программы. Здесь можно определить все интерфейсы, абстрактные классы, пакеты и логические связи между ними.

Интерфейс редактора очень удобен: на палитре расположены основные обозначения языка UML, и выбрав один из элементов, можно с легкостью создать множество элементов диаграммы, соответствующих этому элементу. Например, выбрав класс в палитре, вы с легкостью создадите несколько безымянных классов в рабочем поле. Остается лишь наполнить их смыслом и задать имена, методы, поля и т.д.

Здесь же можно выбрать шаблон проектирования из библиотеки и применить его к рабочему полю, тем самым создав соответствующие прототипы классов (см. **рис. 8**).

Диаграмма использования

Диаграмма использования позволяет изобразить варианты использования программы, классов или модели предметной области. Вы определяете основные действующие лица и их отношения. Для диаграммы использования присутствует другой набор элементов в палитре.

Диаграмма состояний

Диаграммы состояний больше характерны для автоматного подхода в программировании. Они показывают конечный автомат – динамическое состояние системы. С их помощью можно отобразить последо-

вательности состояний программы в процессе ее выполнения. Например, вы можете нарисовать последовательность состояний использования банкомата клиентом (см. **рис. 9**).

Диаграмма деятельности

Показывает переходы от одного вида деятельности к другому. Используется

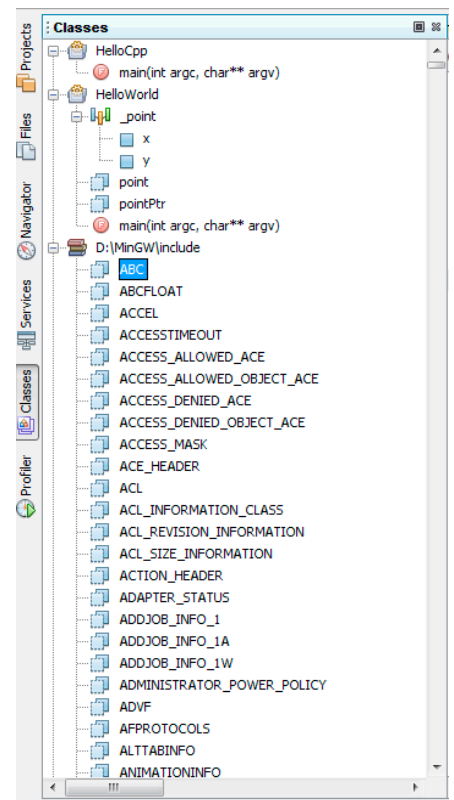


Рисунок 7. Панель классов

Электронное приложение «Open Source»

для динамического представления системы.

Диаграмма последовательности

Показывает временную последовательность обмена сообщениями, т.е. как в процессе работы объекты описанных клас-

сов взаимодействуют между собой посредством вызова соответствующих методов (см. рис. 10).

Диаграмма кооперации

Отражает структурную организацию объекта, отправляющего или принимающего

сообщения, демонстрирует организацию взаимодействия между экземплярами и их взаимосвязь.

Диаграмма компонентов

Позволяет определить компоненты и взаимодействие между ними. Компоненты

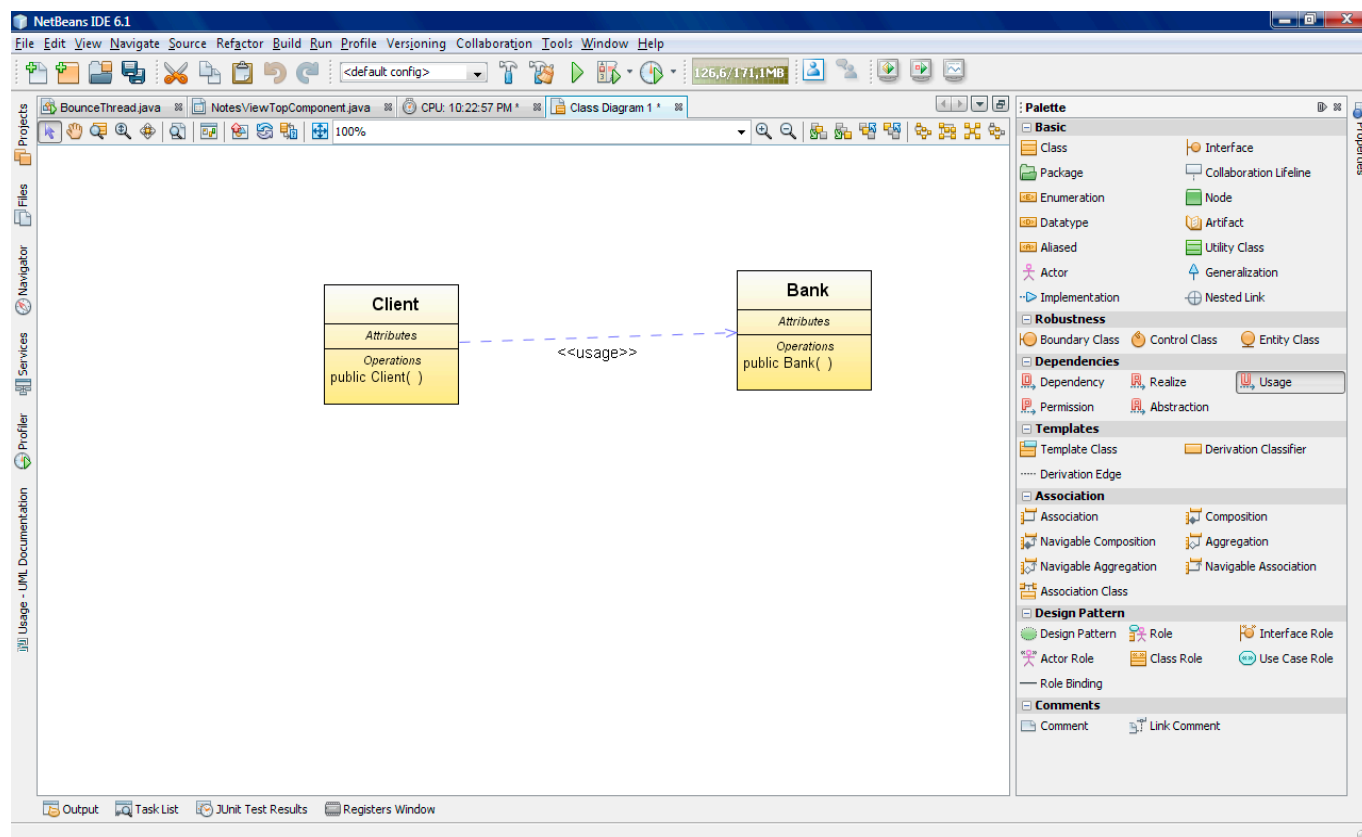


Рисунок 8. Диаграмма классов

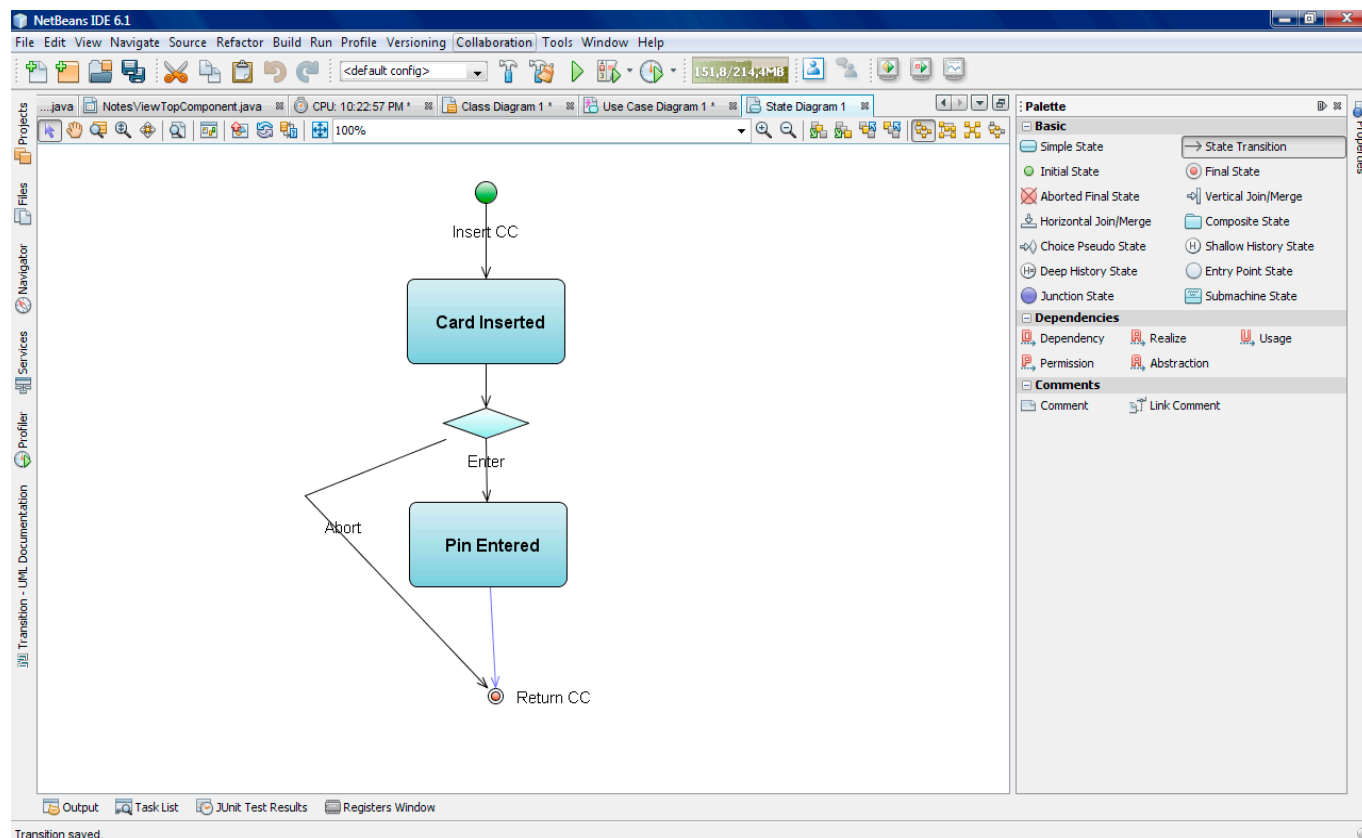


Рисунок 9. Диаграмма состояний

могут состоять из набора классов или артефактов (см. **рис. 11**).

Диаграмма развертывания

Показывает набор узлов и их отношения.

Немного сахара к кофе...

Подводя итог, хочется добавить немного «сахара к кофе». NetBeans – очень гибкая среда, которая быстро развивается благодаря отличной команде разработчиков. Но если у вас есть большой опыт программирования на языке Java и нет желания

ждать, пока разработчики добавят новую функцию в среду, можете реализовать ее сами, тем более что среда в любой поставке позволяет разрабатывать модули. Так что приятного использования!

Александр Шайхразеев
(alexander.shaykhrzeev@gmail.com)

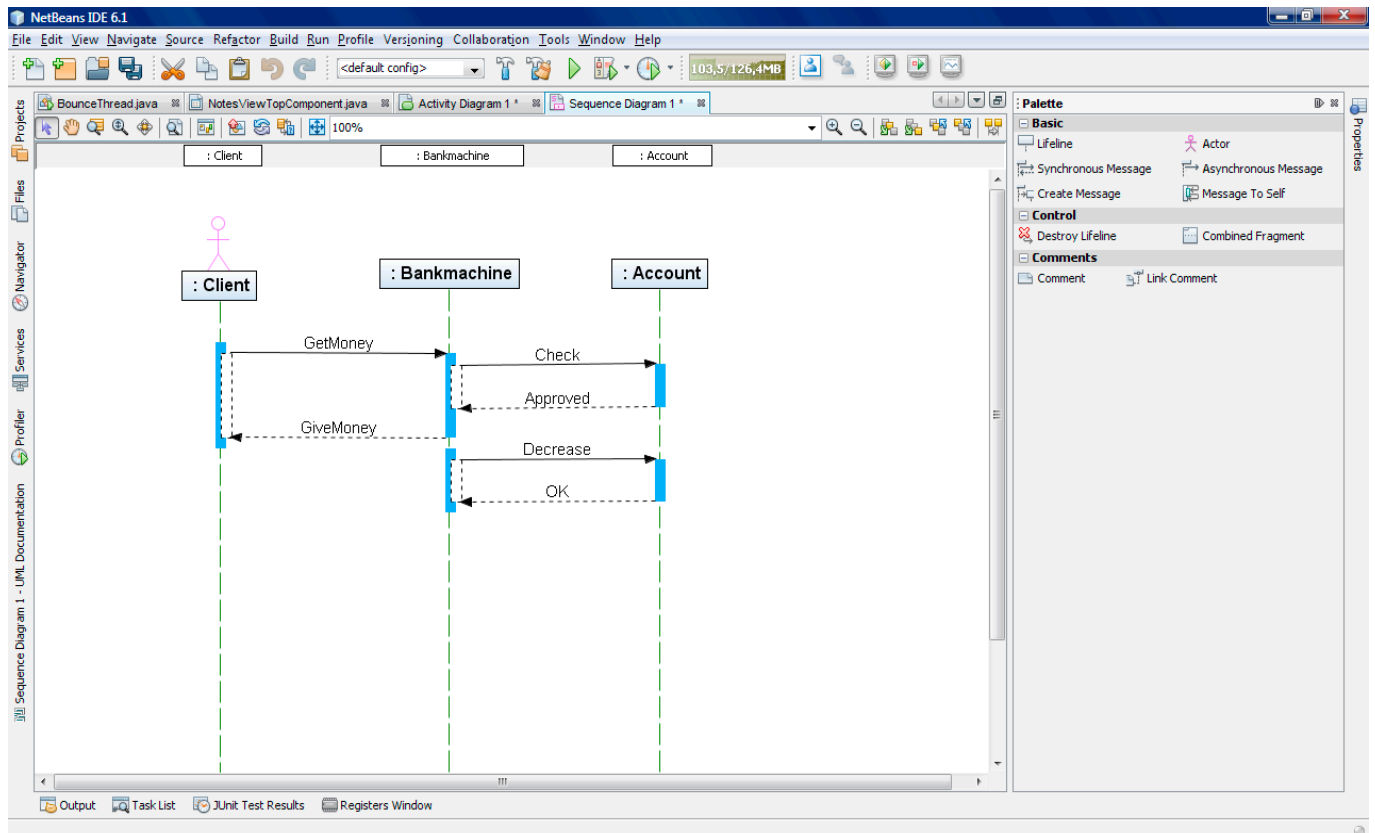


Рисунок 10. Диаграмма последовательности

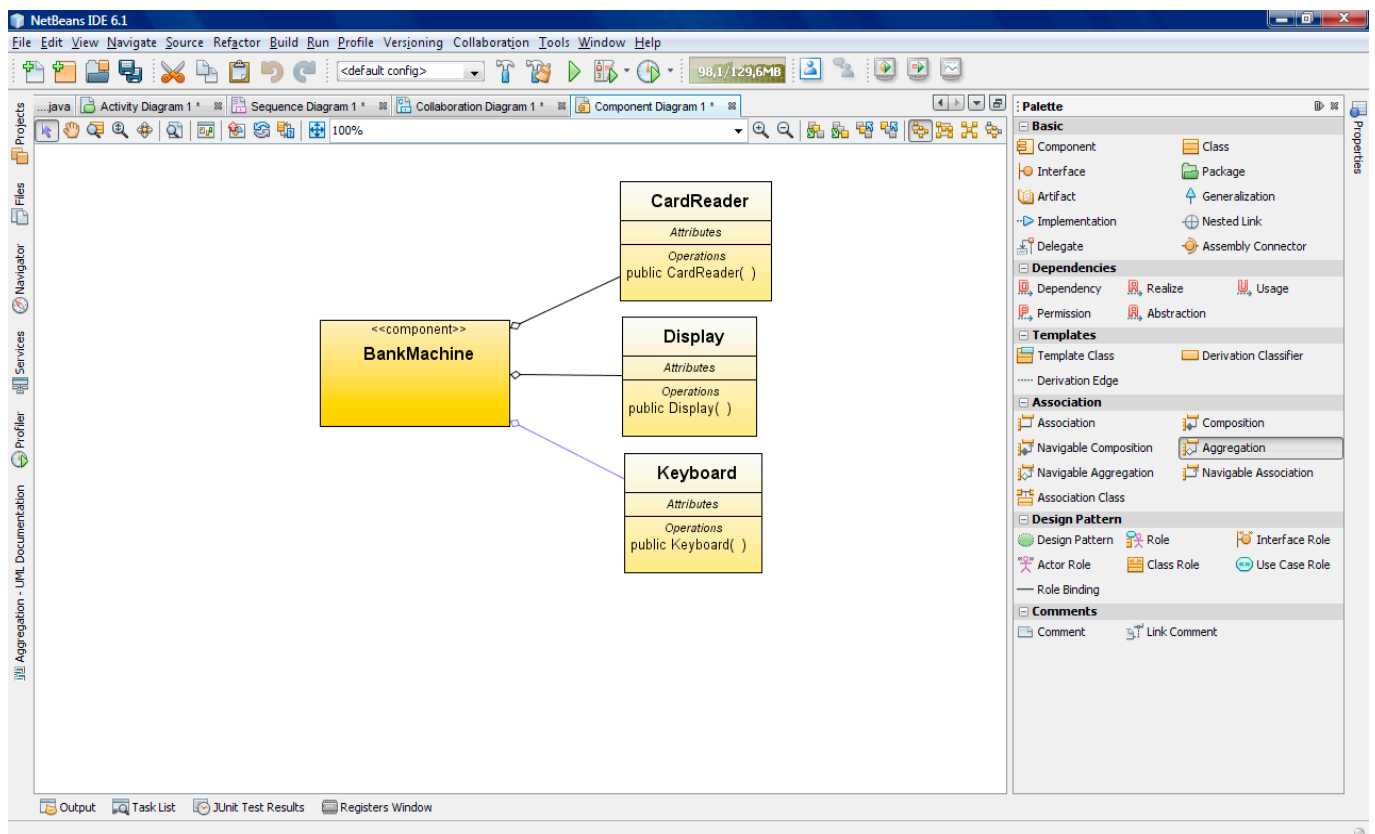


Рисунок 11. Диаграмма компонентов

«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 сегодняшнего 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 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!

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