

Как зарабатывают на Open Source

Крис Касперски

Ожесточенная конкуренция между программистами заставляет не только совершенствовать качество кода, но также искать новые, более эффективные бизнес-схемы, использующие все доступные средства, в том числе и концепцию открытых исходных текстов, уверенно противостоящую компаниям-гигантам.

Практически на любом программистском форуме можно встретить человека (а то и целую группу людей), пропагандирующую «Свободу Кода» (в той или иной трактовке) и призывающую программистов открывать исходные тексты своих проектов, разделив знание с миром. Продавать уже проданную программу – страшный грех и единственно правильный путь, ведущий в рай, – это трудиться на благо человечества в поте лица своего, не заботясь о коммерческой выгоде.

Примечательно, что большинство вышеуказанных личностей сами ничего не программируют, но с удовольствием пользуются программами, за которые не нужно платить. Естественно, им хочется, чтобы таких программ было как можно больше. Тем временем программистам нужно на что-то

жить, и не просто жить, но еще покупать дорогостоящую вычислительную технику, летать на различные конференции, да и просто носиться на Феррари-512-TR в свое удовольствие – почему бы и нет, если их труд оплачивается?!

В 16 лет еще можно работать на благо человечества, не требуя ничего взамен, но с возрастом неизбежно встает проблема: как зарабатывать на жизнь?! Программисты относятся к тем немногим счастливчикам, совмещающим приятное с полезным, зачастую полностью отдающимся своей работе и при этом зарабатывающим неплохие деньги, правда, с каждым годом этих денег становится все меньше и меньше. Массовый рынок давно находится под пятой крупных корпораций, мелкие фирмы скупаются на корню или идут ко дну косяками. Узкоспе-

циализированные секторы рынка еще как-то держатся на плаву, но объемы продаж ПО до приобретения Феррари явно не дотягивают.

Открытые исходные тексты представляют реальную угрозу для компаний-гигантов и дают программистам возможность урвать свой кусок пирога, пополнив карманы пачками приятно хрустящих банкнот. Здесь, в этой статье, мы будем говорить исключительно о коммерческой составляющей Open Source, оставив идеологическую подоплеку в покое. Но для начала нужно разобраться с терминологической путаницей, царящей в этом мире.

Различные виды лицензий

Английский язык относится к числу наиболее примитивных среди всех высококоразвитых, чем, собственно, и объ-

ясняется его популярность, а вовсе не тем, что Америка правит миром и центр научно-технического прогресса находится именно там. В английском очень мало слов, вследствие чего каждое слово имеет множество значений, что, в конечном счете, едва не погубило движение Open Source. Термин «free» означает не только «свободный», но и «бесплатный», поэтому понятие «Free Software» можно толковать двояко, и большинство пользователей (по закону Мерфи) истолковали его совсем не так, как это задумывалось. Чуть позже «free» заменили на «open», но это лишь добавило путаницы и позволило Microsoft и другим компаниям «примазаться» к движению за свободное ПО, путем раскрытия части исходных текстов под подписку о неразглашении и без передачи прав на их последующую модификацию.

Давайте выделим следующие уровни свободы. Абсолютная свобода предоставляется BSD-подобными лицензиями, позволяющими делать с исходными кодами все, что угодно: нарезать на болванки и перепродавать, модифицировать по своему усмотрению, использовать в составе коммерческих проектов... и все это без каких-либо отчислений! В частности, на базе свободной BSD построена закрытая CISCO OS, а ряд сетевых компонентов лег в основу стека NT. Компании-гиганты обогатились, а разработчики... подарили миру свободный код, повторив историю с Прометеем. (Тут кто-то может увидеть противоречие, ведь выше утверждалось, что Open Source создает угрозу компаниям-гигантам и... в то же самое время их обогащает. На самом деле, никакого противоречия здесь нет, CISCO поставляет на рынок маршрутизаторы, снабженные всеми необходимыми сертификатами, и схожую функциональность можно получить от более дешевой связки PC + BSD, коллектив разработчиков BSD маршрутизаторами не занимается, поскольку здесь кроме энтузиазма нужны большие капиталовложения. Microsoft, довольно много позаимствовавшая из BSD, сократила расходы на разработку своей системы и смогла создать более конкурентоспособный серверный продукт, который миллионы людей покупают за деньги, в то время как BSD можно

приобрести по цене носителя, причем BSD не стоит на месте и активно развивается, предоставляя возможности, которые в Windows появляются с огромным запозданием или не появляются вообще! Вот только один пример: при возникновении исключения в одном из драйверов, Windows останавливает всю систему целиком, не позволяя драйверам файловой системы сбросить свои буфера. И хотя такая возможность на уровне архитектуры в NT закладывалась изначально, она до сих пор остается нереализованной, а вот крах одного из модулей в BSD оказывает минимально пагубное воздействие на остальные компоненты системы, позволяя ей корректно завершить свою работу или даже перезапустить «упавший» модуль на лету). Свобода «коммунистического типа» предлагается нам под видом GPL-подобных лицензий, требующих отчислений в случае применения их в коммерческих продуктах или открытия кода всего проекта под все той же GPL. Это справедливо, но... никакой свободой здесь и не пахнет. Во многих случаях GPL является хитрым маркетинговым трюком: «свободная» программа требует гораздо меньших вложений в рекламу для ее продвижения. Людям свойственно скачивать то, что бесплатно, но... как только дело доходит до коммерческого применения, приходится либо нарушать лицензию (ведь открывать свой код только потому, что там используется пара «свободных» библиотек, никто не будет!), либо... платить разработчикам!

Поэтому BSD – это свобода и открытость, а GPL – всего лишь открытость. Что же касается «бесплатности», то это не более чем распространенный миф. По дорогам тоже можно ездить как бы бесплатно, но если вспомнить, что их строительство обычно финансируется из местного бюджета, а местный бюджет – это налоги, то какая-то странная «бесплатность» у нас получается. Другой пример – телевидение, оплачиваемое рекламодателями, включившими в свои продукты добавочную стоимость на рекламу. То есть «бесплатные» телеканалы оплачиваются пивом, стиральным порошком, прокладками и прочими продуктами народного потребления. При всей ненависти зрителей к рекламе «бесплат-

ные» каналы весьма успешно конкурируют с коммерческими.

Давайте попробуем найти бизнес-схему, позволяющую выпускать «бесплатные» программы. Ведь никто же не будет спорить, что бесплатная программа намного более конкурентоспособна своего коммерческого аналога, даже если аналог превосходит ее по ряду параметров.

Ниже обсуждаются различные бизнес-схемы, ориентированные как на индивидуальных программистов, так и на крупные компании. Подчеркну еще раз – именно бизнес-схемы, целью которых является максимализация прибыли, а вовсе не стремление осчастливить человечество в планетарном масштабе.

Мотивы раскрытия исходного текста

Прежде чем обсуждать аргументы сторонников Open Source, обратите внимание на одну интересную деталь. Большинство пользователей предпочитают скачивать готовые бинарные сборки, игнорируя наличие исходных текстов, которые, во-первых, занимают гораздо больше места, а во-вторых, очень часто требуют наличия внешних библиотек, влекущих в свою очередь дополнительные зависимости, причем все они не самым лучшим образом стыкуются друг с другом, отказываясь компилироваться новой (или, наоборот, старой) версией компилятора. И даже когда сборка проходит без сучка и задоринки, она все равно требует процессорного времени и дискового пространства, не говоря уже о сетевом трафике.

Так зачем же скачивать исходные тексты, если мы не собираемся дорабатывать программу «напильником» или проводить аудит безопасности? Правильно, – незачем. Вот потому большинство пользователей их и не качает, а один из самых популярных дистрибутивов Linux – KNOPPIX распространяется без исходных текстов, и ведь никто же не жалуется! Другой вопрос, что собирая программу с определенными ключами, можно здорово сократить ее размер, выкинув все ненужное или оптимизировать под конкретный тип процессора, но при желании этого же эффекта можно добиться путем использования ди-

намически подключаемых библиотек, включив в дистрибутив все необходимые двоичные компоненты, размер которых будет заведомо меньше исходных текстов!

Рынок уже давно все расставил по местам, доказав, что у массового пользователя нет никакой потребности в исходных текстах, по крайней мере, в мире Windows. Пестрота UNIX-подобных систем требует создания множества бинарных сборок, а разработчикам заниматься сборками лень. Гораздо проще выложить исходные тексты, и пусть каждый пользователь самостоятельно компилирует их под свою платформу, ругаясь при этом всеми словами и пытаясь найти готовые сборки на неофициальных сайтах.

Таким образом, на потребительские свойства массового продукта раскрытие исходных текстов никак не влияет. Тем не менее, сторонники движения Open Source утверждают, что: **имея исходные тексты на руках, пользователь может убедиться, что в программе нет закладок, дыр и прочих ошибок. Другими словами, пользователь может провести аудит безопасности или заказать его сторонней фирме.**

Но это даже не смешно! Да, действительно, вокруг крупных проектов таких, например, как OpenBSD, образовалось целое сообщество поклонников, среди которых есть как эксперты, так и новички, занимающиеся поиском ошибок из чисто спортивного интереса или в качестве демонстрации (читай – рекламы) силы своего интеллекта. Однако открытость проекта никак не связана с количеством имеющихся в нем дыр. Взять хотя бы открытый Firefox и сравнить его с закрытым браузером Opera. Лавинообразный рост критических ошибок, обнаруженных в Firefox за последнее время, доказывает, что качество кода определяется исключительно профессионализмом его создателей и доступность исходных текстов для всеобщего аудита не в состоянии исправить многочисленные дефекты изначально небрежно сконструированного продукта.

Кстати говоря, «аудит безопасности» в подавляющем большинстве случаев – формальная процедура, проводимая по заказу тех, кто свято верит в идею, что при целенаправлен-

ном поиске можно обнаружить что-нибудь «интересное». Ни одна фирма со всей «свитой» своих программистов не смогла выпустить продукт, полностью свободный от ошибок, допускающих удаленные атаки, а ведь они знают свои исходные тексты намного лучше, чем кто-то чужой. И это ошибки, допущенные по небрежности! Тщательно же продуманная закладка может (и должна!) срабатывать лишь при сочетании определенной комбинации внешних воздействий. Она не сосредоточена в каком-то одном конкретном месте, а как бы «размазана» по всем исходным текстам. Чтобы ее обнаружить, необходимо не только держать весь проект в голове, но и проанализировать все возможные связи между его компонентами, что невозможно, не говоря уже о том, что закладка может использовать некоторые особенности компилятора/линкера, поэтому они должны быть проанализированы тоже. А железо?! Можем ли мы ему на 100% доверять?! Достаточно вспомнить «странности» реализации команды AAA, чтобы доказать, что, даже имея полный исходный код программы и справочник по командам x86, мы не можем знать, как он поведет себя, встретившись с «необычными» аргументами. Аудит, заказанный конкретной фирме, на 99% бесполезен, а исходные тексты, выложенные на всеобщее обозрение, создают иллюзию безопасности, успокаивающую общественность, поскольку существует вероятность, что кто-то обнаружит спрятанную закладку случайно (и ведь обнаруживают!), однако при этом мы неявно постулируем, что все люди, живущие на земле, это – хорошие люди и они не будут использовать свои знания в корыстных или тем более преступных целях. К тому же шансы обнаружить ошибку не так уж и велики. В частности, в конце августа 2006 года Daniel Bleichenbacher выступил с докладом на конференции криптоаналитиков, где показал, что при стечении определенных обстоятельств цифровая подпись RSA может быть подделана в прямом смысле слова – одной лишь бумагой и карандашом (даже без использования кластера суперкомпьютеров). Виной тому не сам алгоритм RSA, а ошибки его реализации. Одной из таких ре-

лизаций оказался знаменитый проект OpenSSL, которой вплоть до версии 0.9.8c использовал RSA-ключи с экспонентой 3, удаляющие padding-поля PKCS #1 до генерации хэш-суммы, что позволяет удаленному атакующему подделать PKCS #1 сигнатуру, подписанную RSA-ключом, и препятствует корректной проверке различных цифровых сертификатов, использующих PKCS. А ведь OpenSSL подвергался аудиту множество раз. Его исследовали и профессионалы, и любители, и просто «интересующиеся», но... результат все сказал сам за себя: **наличие исходных текстов позволяет каждому пользователю исправлять ошибки, допущенные разработчиками, дорабатывать программу под свои нужды, развивать и совершенствовать ее сообща.**

Прежде чем исправлять ошибку, как минимум, требуется ее воспроизвести, а каждый программист, хоть какое-то время проработавший в службе технической поддержки, знает, что практически никто из пользователей не способен грамотно описать условия, при которых возникает ошибка. А ведь чтобы ее устранить, одной лишь квалификации недостаточно. Первым делом следует разобраться со структурой исходных текстов, что требует времени, причем весьма значительно. Не лучше ли поручить исправление ошибок самим разработчикам? Что же касается новых возможностей... ну хорошо, допустим, разобравшись, какая строка за что отвечает, мы добавили необходимую нам функциональную возможность в программу, изменив десяток файлов в сотне мест. Мы посылаем исправленные исходные тексты координатору проекта, но нас посылают обратно, поскольку уже давно вышла новая версия и никому не интересно «перетаскивать» все изменения из старой, к тому же такая функциональная возможность, по его мнению, никому не нужна, да и реализована она слишком криво, нестабильно и вообще идеологически неправильно. Если наши изменения не будут приняты и включены в основной проект, это означает, что мы намертво привязываем себя к своей «собственной» правленной версии, поскольку переход на новые потребует больших трудозатрат.

На самом деле, для расширения функционала программы достаточно реализовать механизм плагинов или встроенный скриптовый язык. Отладчик OllyDbg распространяется без исходных текстов, но для него написано огромное количество плагинов, которые не привязаны к какой-то одной конкретной версии и могут распространяться как по официальным, так и по независимым каналам: **исходные тексты служат гарантом, что автор программы неожиданно не уедет в Израиль и проект, лишенный поддержки, не умрет, ударив по всем его пользователям.**

Какая-то доля истины в этом утверждении есть, но... раскрыть исходные тексты это одно, а подобрать достойного преемника – совсем другое. Не нужно приводить в пример FireFox, возникший на обломках Netscape, или FAR, создатель которого сосредоточился на RAR, поняв, что не сможет тянуть два проекта одновременно. Возьмите DOS Navigator, пользующиеся во времена DOS огромной популярностью, особенно в среде программистов. Ну и где он сейчас? А ведь исходные тексты открыты... Но сплоченной команды разработчиков, подхвативших знамя и продолживших «дело Ленина», как-то не сложилось и продукт распался на несколько независимых проектов, ни один из которых не получил дальнейшего развития.

Подводя итог, можно сказать, что потребителя в первую очередь интересуют не исходные тексты, а бесплатность продукта, поэтому дальнейший разговор мы будем вести именно в этом ключе.

Мотивы удержания исходных текстов под колпаком

Что мешает программистам распространять исходные тексты вместе с продуктом? В первую очередь – опасение, что кто-нибудь возьмет готовую программу, слегка доработает ее и будет продавать без каких бы то ни было отчислений. Но эта угроза совершенно безосновательна. Если действие происходит в стране, где законы об охране авторского права действуют не только на бумаге, распространителей запросить можно привлечь к ответственности, если же законы «отдыхают», то дораба-

тывать программу совершенно необязательно. Гораздо проще скопировать ее один к одному и продавать, как это делают пираты.

Научное ноу-хау, заложенные в программу, лучше сразу же патентовать, поскольку в противном случае их все равно выдерут дизассемблером, хотя... 99% программ ничего принципиально нового (на уровне кода) в себе не содержат и основная доля находок приходится на дизайнерские решения, то есть на видимый невооруженным глазом интерфейс, а вот код... порою на него страшно смотреть. А кому хочется демонстрировать свой

непрофессионализм? Особенно, если программа содержит ворованные компоненты. Анализируя откомпилированный код, невозможно доказать, соответствует ли он данному исходному тексту или нет, поэтому воровство легко сходит с рук.

Плюс еще стадный признак и инерционность мышления. В мире UNIX принято распространять программы вместе с исходными текстами, и большинство программистов так поступают, даже когда их исходные тексты интересны только им самим. В мире Windows, напротив, большинство программ распространяется в двоичном



2-я МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ И ВЫСТАВКА
БЕСПРОВОДНЫЕ И МОБИЛЬНЫЕ ТЕХНОЛОГИИ В ПРОМЫШЛЕННОСТИ
21 - 23 ноября 2006
 РОССИЯ, МОСКВА, СК ОЛИМПИЙСКИЙ

WIRELESS
COMMUNICATION

WIRELESS
SECURITY

WIRELESS
CONTROL & AUTOMATION

RF ID

БЕСПРОВОДНАЯ СВЯЗЬ

БЕСПРОВОДНЫЕ СИСТЕМЫ БЕЗОПАСНОСТИ

ДИСТАНЦИОННЫЙ КОНТРОЛЬ И УПРАВЛЕНИЕ

СИСТЕМЫ РАДИОЧАСТОТНОЙ ИДЕНТИФИКАЦИИ

Организатор:

INCONEX
International Conferences & Exhibitions

ИНКОНЭКС
 Тел.: (495) 739 55 09
 Факс: (495) 641 22 38
 e-mail: electronica@list.ru

Время работы выставки:
 21 ноября: 10:00 - 18:00
 22 ноября: 10:00 - 18:00
 23 ноября: 10:00 - 16:00

Информационная поддержка:









Проезд:
 Москва, Олимпийский пр., 16
 СК Олимпийский, Северный вход
 Метро: Проспект Мира



ПРИГЛАСИТЕЛЬНЫЙ БИЛЕТ www.inconex.ru

виде, и программисты «зажимают» исходные тексты только потому, что так поступают все остальные.

Бесплатный сыр вне мышеловки

Разобравшись в проблеме, перечислим мотивы раскрытия исходных текстов и распространения программы на бесплатной основе, основываясь на чисто коммерческих соображениях. А мотивов таких достаточно много.

Проба пера

Ух ты, сколько у нас лазерных дисков скопилось! Без хорошего каталогиза-

тора не обойтись. Но все существующие либо неудобны, либо не поддерживают необходимых нам возможностей, либо хотят немереных денег. Возникает мысль – а что если попробовать написать каталогизатор самостоятельно?! Заодно и в программировании попрактикуемся!

И вот, каталогизатор написан, мы с удовольствием пользуемся им и... в какой-то момент до нас доходит, что его можно продать. Мысль интересная, но с точки зрения бизнеса неправильная. Чтобы продавать продукт, его нужно очень сильно довести до ума – это раз. Нужно придумать

хоть какую-то защиту, чтобы не взломали (хотя, все равно ведь взломают) – это два... Также придется приготовить себя к необходимости отвечать на вопросы покупателей, оказывая им поддержку, съедаящую все время и отрывающую от действительно важных проектов – это три. Причем заработать много денег все равно не удастся. Во-первых, на рынке уже существует куча подобных каталогизаторов, во-вторых, хакеры не дремлют, а платить свои кровные не любят даже миллионеры. Наконец, в-третьих, единой платежной системы не существует, и даже те пользователи, которые хотят купить вашу программу, далеко не всегда могут это сделать, поскольку перевод денег требует слишком больших телодвижений.

А вот если выложить каталогизатор в свободный доступ, мы хоть ничего и не приобретем, но зато ничего и не потеряем. Тысячи людей скажут «спасибо», протестируют программу на всем спектре оборудования, подскажут несколько интересных идей по ее улучшению и это еще не все! Кое-кто может подкинуть интересный коммерческий заказ, а кое-кто даже предложить работу.

Случай из жизни. Прохожу собеседование при приеме на работу и на вопрос: что, мол, я за «северный олень» такой, говорю – а вот у вас на рабочем столе я вижу ярлык своей программы. Отношение сразу меняется, и нижняя планка моей квалификации определяется сама собой. Только не надо вставлять свое резюме прямо в программу. Во-первых, это не солидно (специалисты, как правило, не ищут работы, это работа ищет их), во-вторых, жестко вшитое резюме доставляет множество хлопот и проблем после того, как вы уже нашли работу. Лучше просто дать ссылку на сайт с динамически обновляемой информацией.

Оружие против монстров

Конкурировать с компаниями-гигантами на самом деле не так сложно, как это кажется, особенно, если распространять свой продукт на бесплатной основе. Несмотря ни на какие усилия Microsoft, весовая доля Firefox и Opera все растет и растет. И хотя стабильности IE ничего не угрожает, Microsoft с большим удовольствием



COMPLIMENTARY TICKET www.inconex.ru

2-nd INTERNATIONAL CONFERENCE & EXHIBITION

MOBILE & WIRELESS TECHNOLOGIES

21 - 23 November 2006

SC OLIMPIYSKIY, MOSCOW, RUSSIA

WIRELESS COMMUNICATION WIRELESS DATA COMMUNICATION EQUIPMENT

WIRELESS SECURITY WIRELESS SECURITY SYSTEMS

WIRELESS CONTROL & AUTOMATION REMOTE CONTROL & TEST SYSTEMS

RF ID RADIO FREQUENCY COMPONENTS & SYSTEMS

Organizer:



INCONEX
International Conferences & Exhibitions

Tel: +7 (495) 739 55 09
Fax: +7 (495) 641 22 38
e-mail: electronica@list.ru

Opening hours:
21 November: 10:00 - 18:00
22 November: 10:00 - 18:00
23 November: 10:00 - 16:00

Venue:
SC Olimpiyskiy, North exit
16 Olimpiyskiy pr., Moscow
Metro: "Prospekt Mira"

Information support:







БЕСПРОВОДНЫЕ ТЕХНОЛОГИИ



скупил бы конкурентов, если бы они только продавались. Многие компании изначально строят свой бизнес на том, что через некоторое время их купит тот, кому они мешают, или тот, кому нужна клиентская база. Так было в случае и с WinAmp, и с ICQ... да много можно привести примеров! Вон, легендарный исследователь недр Windows Макр Руссинович, и тот продал свой бизнес Microsoft, чем остался страшно доволен: <http://www.sysinternals.com/blog/2006/07/on-my-way-to-microsoft.html>.

Конечно, подобные шаги требуют внушительных капиталовложений, и существует риск, что продукт не «пойдет», конкуренты не обратят на него внимания или рыночная ситуация неожиданно изменится, сделав продукт ненужным (в частности, распространение высокоскоростных каналов связи снизило популярность менеджеров закачек в десятки раз, и выжили те, кто обеспечивал лучшую каталогизацию содержимого, лимитацию трафика, продвинутую планировку и т. д.). Однако затраты на разработку программы можно существенно сократить, если не раздувать штаты сверх необходимого и активно вербовать энтузиастов со всех концов земли, а такие всегда найдутся, если вести проект под эгидой Open Source.

Весь фокус в том, что несмотря на то, что исходный код никому из участников проекта единолично не принадлежит, основную ценность представляет не текст, а... координация его создания. Уберите лидера, и проект развалится, в лучшем случае распавшись на несколько независимых «островков», в худшем – просто задохнется. К тому же, не надо забывать, что движение Open Source имеет разветвленную сеть каналов распространения программного обеспечения и постоянно используется средствами массовой информации в качестве фетиша по поводу и без повода. То есть разработка продукта под знаменем Open Source обеспечивает хорошую рекламу, способствует его продвижению на рынок, что оказывает поразительное психологическое давление на конкурентов, у которых желание прибрать наш бизнес к рукам возникает раньше, чем попытка реально оценить: какую угрозу для их существования он представляет.

Да, по лицензии GPL продукт не может быть куплен без согласия всех разработчиков, на что поклонники Open Source всегда нападают, но зато он может быть похоронен, к чему конкуренты, собственно, и стремятся. Конечно, с психологической точки зрения, не слишком-то приятно работать с перспективой быть закопанным в могилу, но... эта бизнес-схема приносит неплохие деньги, а программное обеспечение в конечном счете все равно устареет, отправляясь в небытие.

Программа как услуга

Рассматривая программу как товар, мы, во-первых, ввязываемся в войну с хакерами и пиратами, норовящими взломать программу, выложить ее в файло-обменную сеть или распространять по цене носителя. А конкуренция?! Десятки компаний предлагают схожие по своим функциональным возможностям программы, и на рынке появляются все новые и новые игры. Как говорится, тут не до наживы! Выжить бы... А что!!! И выживем, если начнем рассматривать программу как услугу.

Продажи программы – не единственно возможный источник дохода. Пусть программа будет бесплатной, пусть ею пользуются все желающие! А мы им поможем! Техническая поддержка, всевозможные семинары и тренинги, литература и бумажная документация... Перечень платных услуг можно продолжать бесконечно. В technically сложных программах (таких, например, как IDA Pro, PC-3000) огромную роль играет наличие центра обучения специалистов, поскольку самостоятельно разобраться с продуктами подобного уровня среднестатистический пользователь не в состоянии. А программные комплексы еще требуют и грамотного внедрения. Здесь мало вставить лицензионный диск и щелкнуть по пиктограмме с надписью «Install»!

Но оставим корпоративный сектор тем, кто на нем пасется. Возьмем вещи простые и куда более приземленные, например, уже упомянутый каталогизатор дисков, бесперспективность продаж которого мы обсуждали выше. А теперь прикрутим к нему он-лайновую службу, позволяющую пользователям обмениваться своими базами дан-

ных (если один человек «вбил» описание дисков в базу, зачем тысячам других делать то же самое?!), после чего начнем привлекать лейблы (лейблами (от англ. label, одним из значений которого является «бренд») в музыкальной индустрии называют компании, покупающие права на произведения и занимающиеся их рекламой, распространением и т. д. (см. «record label» на wikipedia – http://en.wikipedia.org/wiki/Record_label)) (или просто магазины, торгующие дисками), предоставив им возможность информировать пользователей о новинках и вести мониторинг реальной популярности своей продукции. Ведь если у человека есть два альбома такой-то группы, то сообщение о появлении третьего, навряд ли оставит его равнодушным. То есть мы продаем контекстную рекламу, и основным источником прибыли становятся именно лейблы. Прибыль будет тем выше, чем больше у вас пользователей, количество которых обратно пропорционально стоимости программы. То есть для достижения максимальной прибыли программа должна распространяться бесплатно! Наша основная задача – собрать как можно больше пользователей, предоставив им те сервисы, в которых они острее всего нуждаются (заказы новых дисков одним кликом мыши, поиск по названию файла/композиции в базе, пиринговая сеть, наконец!).

Естественно, сразу же возникает вопрос о приватности и безопасности. Передавая он-лайновой службе описание своих дисков, пользователь рискует нарваться на очень крупные неприятности, если эти диски – пиратские (а в большинстве случаев это так и есть). Следовательно, необходимо обеспечить такой механизм передачи данных от пользователей к базе и от пользователя к пользователю, который бы гарантировал со 100% надежностью, что ничей IP-адрес не будет «засвечен». Технически добиться этого очень легко – достаточно использовать прокси-серверы, в роли которых могут выступать и сами пользователи. Да-да, если информация передается не напрямую, а через цепочку наугад выбранных пользователей, каждый из которых не запоминает IP-адрес отправителя, то определить IP-адрес обладателя данного

диска – невозможно. Правда, при одном лишь маленьком условии. Что наша программа работает честно и что в ней нет никаких закладок, оставленных по просьбе (или под нажимом) спецслужб разных стран. Существует только один путь убедить пользователей, что все в порядке, это – открыть исходные тексты.

Вот простой пример бизнес-схемы, основанной на бесплатно распространяемом программном обеспечении. Самое забавное, что написание подобного каталогизатора не требует практически никаких вложений и хороший программист напишет его буквально за неделю. Однако сам по себе он бесполезен, и львиная доля труда уйдет на создание обширной базы данных (ну кого из нас не раздражает, когда Microsoft CD-Player Deluxe не опознает очередной диск?). Переложив эту задачу на плечи пользователей, мы получим базу в обмен на... в том-то и дело, что пользователи не получают ничего.

Допустим, с лейблами договориться не удалось, но, имея базу, созданную первой волной пользователей, мы можем продавать не саму программу, а доступ к базе, который, кстати, легко контролировать, пресекая хакерскую деятельность. Невозможно сгенерировать ключ, если его проверка осуществляется на сервере, хранящем список легально проданных ключей. Постойте, а как же с базой?! Ведь вторая волна пользователей, которым она доступна за деньги, за просто так ее пополнять не будет!!! Следующая хитрость:

мы говорим, что каждый, кто заливает столько-то описаний дисков, получит бесплатный доступ на какое-то время (а то и на всю жизнь).

В общем, главное фантазию иметь и не заикливаться на схеме «программа – деньги».

Другие мотивы

Разработки, оплачиваемые различными грантами или созданные в ходе различных исследовательских программ, дотируемых из госбюджета, как правило, не принадлежат их создателю и распространяются в открытых текстах на бесплатной основе. Но в основном это относится к научным программам, ареал обитания которых никак не пересекается с офисными продуктами и другими приложениями массового назначения.

Еще одним мотивом к раскрытию кода становится сворачивание работы над проектом, как произошло, например, в случае с Netscape. К сожалению, остальные компании следовать этому примеру не хотят. Например, Compuware, объявившая о прекращении поддержки soft-ice, наотрез отказалась отдавать исходные коды последней версии за разумную сумму, хотя она могла бы их просто подарить миру, ничего от этого не потеряв. Но... в мире бизнеса законы логики работают по-другому. Это намного хуже, чем собака на сене (по крайней мере, собака на сене лежит, и если она позволит корове съесть его, лежать ей придется на сырой земле, а для хвоста это нехорошо). Compuware в лю-

бом случае не получит никакой прибыли от заброшенного soft-ice и не потерпит убытка, передав soft-ice конкурентам, ведь других отладчиков у нее все равно нету, но... существуют такие понятия, как жадность и глупость. Глупость – хоронить легендарный продукт вместо того, чтобы искать новые бизнес-схемы для его распространения.

К сожалению, подобные ситуации не редкость, а скорее общепринятое правило поведения.

Заключение

Все упирается в деньги. Мы живем не при коммунизме, и окружающие нас товары и услуги стоят денег, следовательно, программировать на чистом энтузиазме можно либо за счет родителей, либо добывая средства к существованию каким-нибудь иным способом (торговлей бананами, например), что превращает программирование из ремесла в хобби. Ведь помимо бизнес-схем в мире есть и простое увлечение, страсть, стремление, азарт, любовь к компьютерам, иногда переходящая в манию, граничащую с помешательством. Есть люди, ощущающие буквально физическую потребность писать программы, разрабатывать новые алгоритмы или просто проводить за монитором все свободное и несвободное время. Но далеко не все из них примыкают к движению Open Source, предпочитая держаться в стороне от шумной и пестрой толпы, построенной на бизнесе и пропаганде. ●

