

# Как запустить «САМО-Тур» для Windows под Linux

«...без собственного войска (программистов. – Прим. автора)  
государство непрочное – более того, оно всецело зависит  
от прихотей фортуны...»

Николо Макиавелли, Государь, 1532



**Сергей Барановский**

**Туристический бизнес отличается от любого другого, офисного. Сегодняшняя его проблема – это отсутствие программ, автоматизирующих офисную деятельность в полной мере. Речь не идёт об альтернативных Open Source-продуктах, т.к. их просто нет. Проблема в том, что даже коммерческие и специализированные продукты, как «САМО-Тур» и «Мастер-Тур», не дотягивают по своим возможностям до уровня потребностей туроператоров. Статья рассказывает о том, как можно сэкономить, запустив один из коммерческих продуктов на платформе Linux, что изначально не поддерживалось разработчиком. Тот, кто не кормит свою армию программистов-разработчиков, кормит чужую!**

**П**оддержка туристических услуг техническими решениями часто отстаёт от существующих потребностей туроператоров, что приводит к повышению издержек и сказывается на конечном потребителе. Угодить всем операторам и создать универсальный продукт невозможно, как нельзя создать один универсальный автомобиль, подходящий всем.

Не секрет, что большинство из нас хотели бы отдыхать с комфортом, по разумным ценам, но кто из нас задумывается, какая часть денег, оплаченных за путёвку, идёт непосредственно на отдых, а какая покрывает издержки производства?

Кажется, что работа менеджеров проста: расскажи красиво туристу о продукте, оформи документы, полу-

чи деньги – типичное представление работы турфирм обывателем. Но это как надводная часть айсберга, за которой кроется очень много внутренней работы, скрытой от потребителей.

Распечатать путёвку – означает иметь какой-либо текстовый редактор. Связаться с гостиницей – значит использовать электронную почту или факс. Забронировать билеты – исполь-

зовать браузер и on-line-сервис. Хранить данные о клиенте – использовать специфическую базу данных.

Уважающая себя фирма будет работать «в белую», а это означает, что все технические решения будут выполнены без нарушений авторских прав и будут поддержаны либо купленными коммерческими лицензиями сторонних фирм, либо GPL-подобными, либо у фирмы будет собственный штат программистов. Что из этого дешевле сейчас и выгоднее в перспективе? Можно установить OpenOffice и другие продукты на базе Linux, они удовлетворяют больше половины потребностей всех турфирм и сэкономят значительно средств, но как быть крупным фирмам, где без автоматизации производства нельзя, где в день могут проходить сотни и тысячи заявок. Простым хранением текстовых файлов или таблиц с ценами не обойдёшься. Нужны специальные программные продукты, которых просто нет! Как и нет стандартов в этой сфере.

Разные фирмы находят разные решения этой проблемы и в силу конкуренции не спешат делиться своими секретами друг с другом.

На сегодня можно выделить два больших программных продукта, старающихся занять рынок обслуживания туроператоров, – это «Мастер-Тур» [1] и «САМО-Тур» [3]. Обе программы присутствуют на рынке несколько лет и далеко от совершенства.

Первая пишется на C, вторая – на Delphi, и ни одна не предназначена для использования под Linux, по крайней мере такое мнение можно услышать на официальных форумах [2, 4]. На мой взгляд, обе фирмы, выпускающие эти продукты, «вырыли себе большую яму» и оказались заложниками своих решений. Рынок требует продуктов под Linux. Переписать большое количество кода можно, но для этого необходимы дополнительные программисты, которых и так в этих фирмах не хватает. А привлечение сторонней помощи, как в GPL-продуктах, невозможно, т.к. код закрытый. Включение поддержки одного лишь экспорта данных в продукты OpenOffice из «САМО-Тур» заняло почти год, и, предполагаю, было реализовано только после того, как было «престимулировано» потребностями одного крупного клиента, т.к. до этого они отказывались делать поддержку OpenOffice в своих продуктах.

Других продуктов подобного уровня или их аналогов сегодня нет.

Турфирмам выкручивают руки, им надо работать сейчас, а не держать штат программистов или ждать несколько лет завершённой реализации под Linux. Вот и получается, выгоднее купить сотню-другую лицензий для импортной коммерческой закрытой операционной системы и отработать сезон-другой, окупив затраты за счёт туристов, т.е. нас с вами, нежели ждать появления на рынке подходящего продукта, позволяющего снизить цены. Грустно, но, в то же время наш народ очень изобретателен в плане «как не платить», поэтому снизить цены на конечные продукты законным путём можно и оставить деньги в нашей стране тоже!

## Требования

Для перевода клиентов «САМО-Тур» под Linux потребуются лицензионная и работающая версия «САМО-Тура»

под Windows. Работающий САМО-сервер, дистрибутив Linux Fedora release 8 (Werewolf) для i386-платформы. Подключение к Интернету или все упоминаемые и устанавливаемые дополнительно пакеты. Средства разработки приложений, компилятор gcc и сопутствующие пакеты не нужны.

## Начало установки

Предполагаем, что установка Linux Fedora 8 (i386) прошла успешно. Сверить версию ядра можно, запустив команду «uname -a», которая должна выдать следующий результат:

```
Linux new02 2.6.23.1-42.fc8 #1 SMP Tue Oct 30 13:55:12 EDT 2007
i686 i686 i386 GNU/Linux
```

## Отключение yum-updatesd

Обычно после установки демон yum-updatesd стоит в автозапуске и отвечает за проверку и установку обновлений к системе. Вещь полезная, но нам она не очень нужна, т.к., во-первых, это будет лишний трафик, представьте, если сотня-другая компьютеров начнет для обновления скачивать одни и те же пакеты (хотя тут на усмотрение администратора), а, во-вторых, обновления могут помешать запуску нужных нам команд ниже по тексту, т.к. yum и yum-updatesd не могут работать одновременно.

Если этот шаг пропустить, то вполне возможна ситуация, что при запуске команды «yum install что-то» вы получите:

```
Existing lock /var/run/yum.pid: another copy is running as pid 3592.
Another app is currently holding the yum lock; waiting for it to exit...
Another app is currently holding the yum lock; waiting for it to exit...
Another app is currently holding the yum lock; waiting for it to exit...
...
Exiting on user cancel.
```

Отключить запуск можно двумя путями:

Первый – остановить службу yum-updatesd командой:

```
# service yum-updatesd stop
```

или

```
# /etc/rc.d/init.d/yum-updatesd stop
```

После чего вы увидите:

```
Останавливается yum-updatesd: [ OK ]
```

Это поможет до следующей перезагрузки или смены уровня запуска.

Второй – убрать эту службу из автозапуска вообще, чтобы после перезагрузки ситуация не повторялась. Проще всего это сделать командой:

```
# chkconfig yum-updatesd off
```

которая «молча» выполнится, удалив соответствующие файлы запуска из директорий /etc/rc.d/rd?.d.

## Полезные приложения и настройка прокси

Для комфортной работы и успешной настройки нам понадобятся дополнительные пакеты.

Комфорт – чувство относительное, поэтому ряд команд и пакетов можно не ставить, заменив их другими. Опытные пользователи сами поймут, где это можно сделать, всем остальным будет проще следовать всем пунктам.

Если в вашей фирме Интернет работает не как «проходной двор», а ведётся учёт и контроль соединений, то, скорее всего, это делается через прокси. Для того чтобы не писать лишние ключи к программам `wget`, `yum` и другим, проще задать прокси через переменную окружения, выполнив команду:

```
export http_proxy=http://user:password@192.168.0.1:3128
```

от имени пользователя, под которым будет выполняться запуск программ.

Если ваш прокси настроен на работу без аутентификации, то можно написать:

```
export http_proxy=http://192.168.0.1:3128
```

Запись можно сократить даже до:

```
export http_proxy=192.168.0.1:3128
```

но опытным путём установлено, что это работает не со всеми программами.

По завершении работы следует не забыть «почистить» переменные окружения и `.bash_history`, т.к. в них будет содержаться пароль в открытом виде. В конце статьи мы вернёмся к этому вопросу.

Также во внутреннем пакетном фильтре, используя команду `iptables`, следует не забыть создать правила, разрешающие исходящие подключения и сопутствующие им обратные пакеты. По умолчанию, после установки, все исходящие соединения разрешены, поэтому не должно быть проблем.

Те же, кто внесет правку в правила пакетного фильтра, без труда напишут разрешающие правила в обе цепочки для исходящих TCP-соединений по портам 80, 3128, 1433 и не забудут про пакеты, идущие к DNS-серверу и обратно, в случае его использования.

Первое, что нам нужно для упрощения жизни, – это подключить репозиторий `livna` [5], чтобы после устанавливая из него необходимые пакеты одной строчкой, а не ставить компиляторы, средства разработки и пр. Для этого скачиваем пакет `livna-release-8.rpm`:

```
wget http://rpm.livna.org/livna-release-8.rpm
```

После устанавливаем его:

```
# rpm -ihv livna-release-8.rpm
```

```
предупреждение: livna-release-8.rpm:
Заголовок V3 DSA signature: NOKEY, key ID a109blec
Подготовка... ##### [100%]
1:livna-release ##### [100%]
```

Затем ставим пакеты `mc` и `unrar`, отвечая «y».

```
# yum install mc unrar
```

```
...
mc      i386      1:4.6.1a-50.20070604cvs.fc8 updates      2.1 M
unrar   i386      3.7.8-1.lvn8      livna        102 k
...
Is this ok [y/N]: y
Downloading Packages:
(1/2): unrar-3.7.8-1.lvn8 100% |=====| 102 kB  00:00
(2/2): mc-4.6.1a-50.20070 100% |=====| 2.1 MB  00:00
warning: rpmts_HdrFromFdno: Header V3 DSA signature: NOKEY, key ID 4f2a6fd2
Importing GPG key 0x4F2A6FD2 "Fedora Project <fedora@redhat.com>"
from /etc/pki/rpm-gpg/RPM-GPG-KEY-fedora
Is this ok [y/N]: y
warning: rpmts_HdrFromFdno: Header V3 DSA signature: NOKEY, key ID a109blec
Importing GPG key 0xA109BLEC "Livna.org rpms <rpm-key@livna.org>"
from /etc/pki/rpm-gpg/RPM-GPG-KEY-livna
Is this ok [y/N]: y
...
Installed: mc.i386 1:4.6.1a-50.20070604cvs.fc8 unrar.i386 0:3.7.8-1.lvn8
Complete!
```

## Установка FreeTDS и Wine

Далее ставим пакеты, непосредственно необходимые для работы. Это Wine [7] – эмулятор, позволяющий запускать программы, написанные для Windows. И FreeTDS [6] – программа, позволяющая UNIX и Linux-приложениям работать с MS SQL-сервером. Так как «CAMO-Тур» имеет клиент-серверную архитектуру, то клиентам понадобится подключаться к серверу.

Ранее эти пакеты приходилось ставить вручную, скачивая исходные коды и компилируя их, но сейчас ситуация изменилась, и готовые rpm-пакеты прекрасно устанавливаются. Изменился лишь путь расположения файлов по умолчанию. Интересную статью по настройке FreeTDS в UNIX вы можете найти по адресу [8]. Раньше установка происходила более сложно, но в процессе написания статьи выяснилось, что жизнь становится проще, и в нашем случае это плюс.

Установку обоих пакетов вместе с необходимыми зависимыми пакетами можно запустить командой:

```
# yum install freetds wine
```

После чего будет предложено установить пакеты и относящиеся к ним зависимости:

```
...
Installing:
freetds      i386      0.64-7.fc8      fedora      869 k
wine         i386      0.9.58-1.fc8    updates     20 k

Installing for dependencies:
jack-audio-connection-kit.i386 0:1.0.3-1.fc7 updates 139 k
libfreebob.i386 0:1.0.3-1.fc7 fedora 154 k
unixODBC.i386 0:2.2.12-5.fc8 updates 954 k
wine-capi.i386 0:0.9.58-1.fc8 updates 25 k
wine-cms.i386 0:0.9.58-1.fc8 updates 40 k
wine-core.i386 0:0.9.58-1.fc8 updates 10 M
wine-esd.i386 0:0.9.58-1.fc8 updates 35 k
wine-jack.i386 0:0.9.58-1.fc8 updates 37 k
wine-ldap.i386 0:0.9.58-1.fc8 updates 87 k
wine-nas.i386 0:0.9.58-1.fc8 updates 23 k
wine-tools.i386 0:0.9.58-1.fc8 updates 628 k
wine-twain.i386 0:0.9.58-1.fc8 updates 46 k
...
Installed: freetds.i386 0:0.64-7.fc8 wine.i386 0:0.9.58-1.fc8
Dependency Installed: jack-audio-connection-kit.i386 0:1.0.3-1.fc7
libfreebob.i386 0:1.0.3-1.fc7 unixODBC.i386 0:2.2.12-5.fc8
wine-capi.i386 0:0.9.58-1.fc8 wine-cms.i386 0:0.9.58-1.fc8
wine-core.i386 0:0.9.58-1.fc8 wine-esd.i386 0:0.9.58-1.fc8
wine-jack.i386 0:0.9.58-1.fc8 wine-ldap.i386 0:0.9.58-1.fc8
wine-nas.i386 0:0.9.58-1.fc8 wine-tools.i386 0:0.9.58-1.fc8
wine-twain.i386 0:0.9.58-1.fc8
Complete!
```

По окончании установки вы должны увидеть:

```
Installed: freetds.i386 0:0.64-7.fc8 wine.i386 0:0.9.58-1.fc8
Dependency Installed: jack-audio-connection-kit.i386 0:1.0.3-1.fc7
libfreebob.i386 0:1.0.3-1.fc7 unixODBC.i386 0:2.2.12-5.fc8
wine-capi.i386 0:0.9.58-1.fc8 wine-cms.i386 0:0.9.58-1.fc8
wine-core.i386 0:0.9.58-1.fc8 wine-esd.i386 0:0.9.58-1.fc8
wine-jack.i386 0:0.9.58-1.fc8 wine-ldap.i386 0:0.9.58-1.fc8
wine-nas.i386 0:0.9.58-1.fc8 wine-tools.i386 0:0.9.58-1.fc8
wine-twain.i386 0:0.9.58-1.fc8
Complete!
```

После установки появится файл `/etc/freetds.conf`, куда мы прописываем сервер, с которым будем работать (адрес, порт, протокол, кодировку), добавлением следующих строк:

```
[Server80]
host = 192.168.0.2
port = 1433
tds version = 8.0
client charset=WINDOWS-1251
```

где Server80 – имя сервера, по которому с ним будет общаться ODBC. Строчка «`client charset=WINDOWS-1251`» нужна для того, чтобы русские буквы правильно передавались в запросах, иначе вы увидите вместо списка пользователей на входе в «CAMO-Тур» абракадабру.

## Настройка unixODBC

Если вы были внимательны, то заметили, что у нас при установке `wine` оказался зависимым, и был установлен пакет `unixODBC-2.2.12-5.fc8`.

Далее необходимо настроить базу данных Server80 через `unixODBC`, чтобы её «видел» `wine` и тем самым «видели» и все приложения, запущенные под ним. Для этого создаём два файла, имя и размещение которых не критичны. В дальнейшем мы к ним обратимся. Для удобства имена файлов совпадают с указанными в [8].

Файл `tds.driver.template`:

```
[FreeTDS]
Description=FreeTDS for MSSQL
Driver=/usr/lib/libtdsodbc.so.0
```

Файл `tds.datasource.template`:

```
[labirint]
Driver=FreeTDS
Description=Labirint database
Trace=No
Servername=Server80
Database=labirint
```

После чего мы выполняем с правами администратора установку драйвера:

```
# odbcinst -i -d -f tds.driver.template
```

В результате получим:

```
odbcinst: Driver installed. Usage count increased to 1.
Target directory is /etc
```

Далее с правами пользователя(!) устанавливаем базу, которая будет работать через драйвер, установленный администратором (root) для всех:

```
$ odbcinst -i -s -f tds.datasource.template
```

Команда выполняется «молча».

С этого момента можно считать, что установка эмулятора закончена и все последующие настройки следует выполнять из-под него.

**Замечание:** часть команд, как и установка эмулятора, проходит с правами администратора (root), а все последующие запуски приложений и пр. с целью повышения безопасности выполняются с правами обычных пользователей.

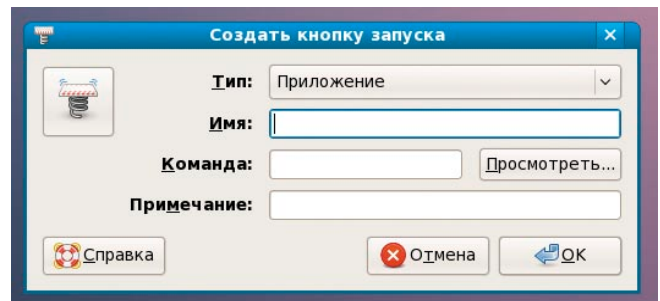


Рисунок 1. Окно создания кнопки запуска (ярлыка на «Рабочем столе»)

То есть для работы с системой вы создаете учетную запись обычного пользователя и под ним запускаете X, браузеры, «CAMO-Тур» под эмулятором и другие программы.

Чтобы отделить возможные ошибки на этом этапе от тех, что могут быть совершены в будущем, и не ломать голову, почему нет связи с сервером, не обязательно, но можно проверить работу с базой данных от имени обычного пользователя командой:

```
isql -v labirint user pass
```

где labirint – имя базы из файла `tds.datasource.template`, а user и pass – логин и пароль, заведённые на сервере. В случае успеха вы должны увидеть приглашение базы данных:

```
+-----+
| Connected! |
| |
| sql-statement |
| help [tablename] |
| quit |
| |
+-----+
SQL>
```

Если же соединение по какой-то причине не было установлено, то вы увидите ошибку вроде этой:

```
[S1000][unixODBC][FreeTDS][SQL Server]Unable to connect to data source
[37000][unixODBC][FreeTDS][SQL Server>Login failed for user 'wintour'.
[28000][unixODBC][FreeTDS][SQL Server>Login incorrect.
[ISQL]ERROR: Could not SQLConnect
```

**Замечание:** обратите внимание, если вы запускали команду выше, то переданные ей через командную строку логин и пароль сохранились в открытом виде в файле `.bash_history`.

## Работа под wine

После установки эмулятора необходимо его запустить, чтобы в домашней директории запускающего его пользователя была создана директория `.wine`, в которой будет находиться эмулируемый диск C: и др. файлы. Можно создать эти директории вручную, но наиболее простой способ запустить `wine`, передав ему в качестве параметра имя любого несуществующего файла, например 123.

```
wine 123
```

после этого вы получите сообщение об ошибке:

```
wine: could not load L"C:\\windows\\system32\\123.exe":
Module not found
```

Но мы и так знаем, что такого файла нет, зато после мы сможем взять программу «САМО-Тур» с работающей Windows-машины и скопировать в директорию: «/home/домашняя\_директория\_пользователя/.wine/drive\_c/Program Files/WinTour».

Практика показала, что под эмулятором это наиболее простой и быстрый путь, чем установка и настройка из дистрибутива.

После копирования программы, с помощью правой кнопки мыши, создаём на рабочем столе ярлык (кнопку запуска) (см. рис. 1).

В поле «Команда» с помощью кнопки «Посмотреть...» находим файл.

Учтите, что директории и файлы, имена которых начинаются с точки, как «.wine», отображаться не будут. Для того чтобы их увидеть, надо нажать правую кнопку мыши и включить показ скрытых файлов (см. рис. 2).

Или руками прописываем:

```
"/home/домашняя_директория_пользователя/.wine/drive_c/Program Files/WinTour/Wintour.exe"
```

Далее дописываем слева от этой команды wine и создание переменной WINEPREFIX:

```
env WINEPREFIX="/home/labirint/.wine" wine "/home/домашняя_директория_пользователя/.wine/drive_c/Program Files/WinTour/Wintour.exe"
```

Часто работает и без указания переменной префикса пути:

```
wine "/home/домашняя_директория_пользователя/.wine/drive_c/Program Files/WinTour/Wintour.exe"
```

Если необходимо, то меняем значок на иконке, записываем имя и комментарий, после чего программа «САМО-Тур» готова к запуску.

## Запуск «САМО-Тур»

Запускаем программу щелчком по иконке и видим ошибку (см. рис. 3, 4).

Как видим, программа запускается, но с базой работать не хочет. Так как это приложение Delphi, то для нормальной работы ей требуется наличие BDE (Borland Database Engine), что недвусмысленно указано во второй ошибке, к установке которого мы и переходим.

## Установка BDE

Забегая немного вперёд, скажу, что для установки нам нужен файл Bdeinst.dll, так, что если он у вас есть, то можете пропустить несколько абзацев. Если же у вас нет этого файла, то вы найдёте его внутри cab-архива bdeinst.cab.

Скачиваем файл следующей командой:

```
wget http://www.jrsoftware.org/download.php/bdeinst.cab
```

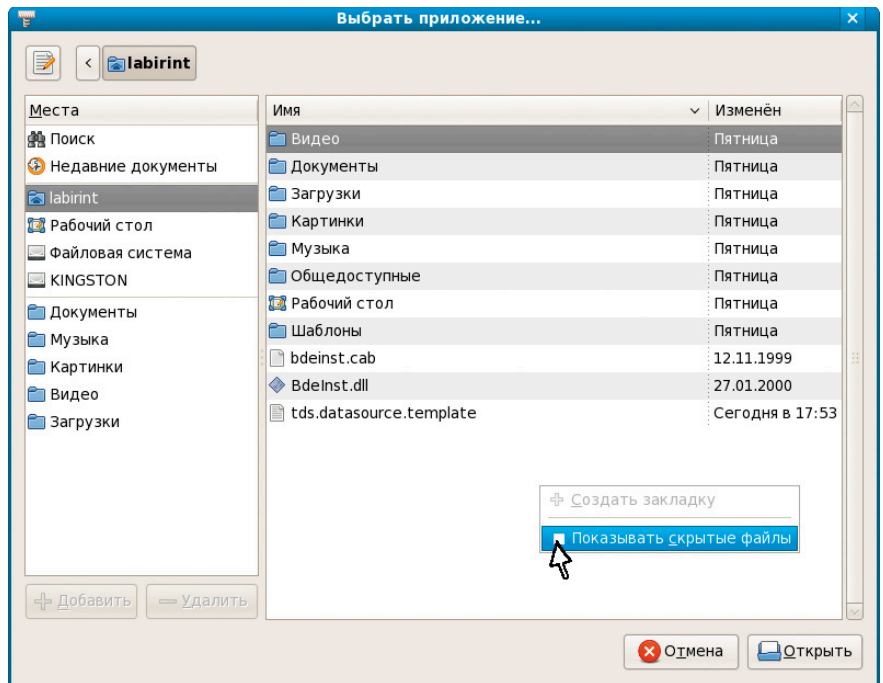


Рисунок 2. Включение опции «Показывать скрытые файлы» (имена которых начинаются с точки)

В результате выполнения команды узнаём, что файл был скачан с другого адреса: <http://files.jrsoftware.org/misc/bdeinst.cab> и его размер 3 663 697 (3.5M) байт.

Следующий этап – это распаковка архива. Для этой цели нам понадобится команда cabextract. В установке по умолчанию её нет, поэтому найти её можно тут: <http://www.cabextract.org.uk>, а скачать, распаковать, скомпилировать и поставить командами:

```
wget http://www.cabextract.org.uk/cabextract-1.2.tar.gz
tar -zxvf cabextract-1.2.tar.gz -C /progi
cd /progi/cabextract-1.2
./configure
make
#make install
```

но, т.к. это требует средств разработки и компилятора, которые у нас не установлены, и вообще, это команда нам нужна всего лишь один раз, чтобы достать из архива файл Bdeinst.dll, то, как в анекдоте про муравья, который, глядя на железнодорожный рельс, произнёс фразу: «Умный в го-

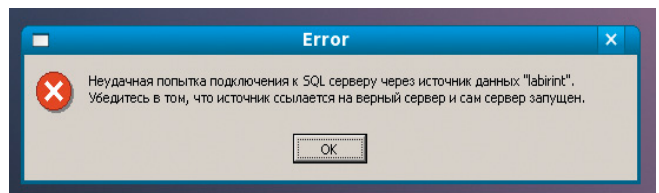


Рисунок 3. Ошибка подключения к SQL-серверу

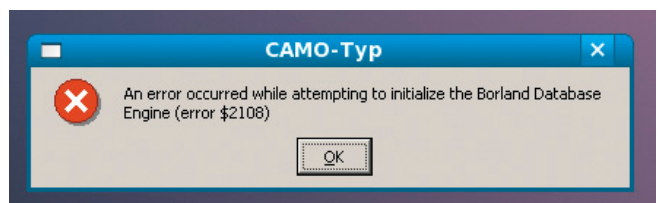


Рисунок 4. Ошибка инициализации BDE

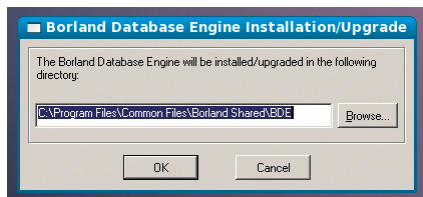


Рисунок 5. Окно с выбором пути установки BDE

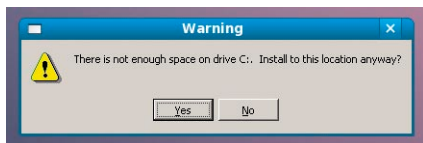


Рисунок 6. Предупреждение о нехватке свободного места

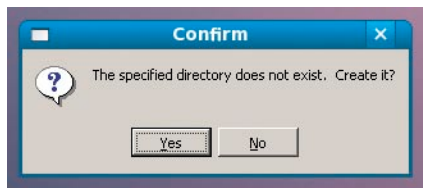


Рисунок 7. Подтверждение создания директории

пу не пойдёт, умный гору обойдёт», мы воспользуемся установкой через yum, запустив команду:

```
# yum install cabextract
```

```
...
Installed: cabextract.i386 0:1.1-7.fc8
Complete
```

После этого извлекаем из архива нужный файл:

```
$ cabextract bdeinst.cab
```

```
bdeinst.cab: WARNING: possible 4536 extra
bytes at end of file.
Extracting cabinet: bdeinst.cab
  extracting BdeInst.dll
All done, no errors.
```

В результате, у нас должен получиться файл BdeInst.dll размера 3 853 824 байт от 27.01.2000.

**Отступление.** Кстати, а вы знаете, чем умный человек отличается от мудрого? Умный – знает, как выйти из любой ситуации, например, распаковать файл, установив компиляторы, архиваторы, библиотеки и пр., потратив полдня. А мудрый – знает, как не попасть в эту ситуацию, он, скорее всего, распакует файл на другой Windows-машине.

Имея нужную библиотеку, регистрируем её в wine-эмуляторе, выполнив следующую команду от имени обычного пользователя:

```
wine c:/windows/system32/ \
  regsvr32.exe BdeInst.dll
```

после чего на экране «бежит» индикатор прогресса и выскакивает окошко с вопросом, куда ставить Borland Database Engine (см. рис. 5).

Оставляем по умолчанию: «C:\Program Files\Common Files\Borland Shared\BDE», а в Linux это будет: «/home/домашняя\_директория\_пользователя/.wine/drive\_c/Program Files/Common Files/Borland Shared/BDE».

Нажимаем «OK» и видим ошибку о нехватке места (см. рис. 6). Программа же не знает, что мы её безбожно дурачим, поэтому нажимаем «yes».

Если же вы сомневаетесь, то проверить свободное место в Linux можно командой df, запомнить легко, сокращение от слов disk free.

Далее, выскакивает вопрос о создании директории, т.к. её нет, мы соглашаемся (см. рис. 7).

После быстро пробегают индикатор прогресса, и все окошки исчезают. Лишь в консоли, откуда осуществлялся запуск установки, можно увидеть сообщение о том, что библиотека BdeInst.dll была успешно установлена и зарегистрирована.

Successfully registered DLL BdeInst.dll

Следующий этап не обязательный, но мы можем проверить, видит ли BDE нашу базу, для чего запускаем от обычного пользователя:

```
wine "c:/Program Files/ \
  Common Files/Borland Shared/ \
  BDE/BDEADMIN.EXE"
```

В результате, мы увидим окно BDE Administrator (см. рис. 8).

Если мы заходим посмотреть параметры базы, то мы можем нажать на «+» слева от её имени, после чего выскочит окно ввода логина и пароля (см. рис. 9).

Теперь, когда мы убедились, что BDE Administrator видит базу, можно ещё раз переходить к запуску «CAMO-Тура». Щёлкаем по ранее созданной иконке и видим окошко (см. рис. 11).

Ждём пару секунд, и появляется привычное окно входа в систему (см. рис. 10).

Набираем имя для входа и пароль, и мы в программе. Выбираем в меню справочники, заявки и можем полноценно работать с программой (см. рис. 12).

## Дополнительно

Так как ядро – важная часть любой системы, то для более стабильной и безо-

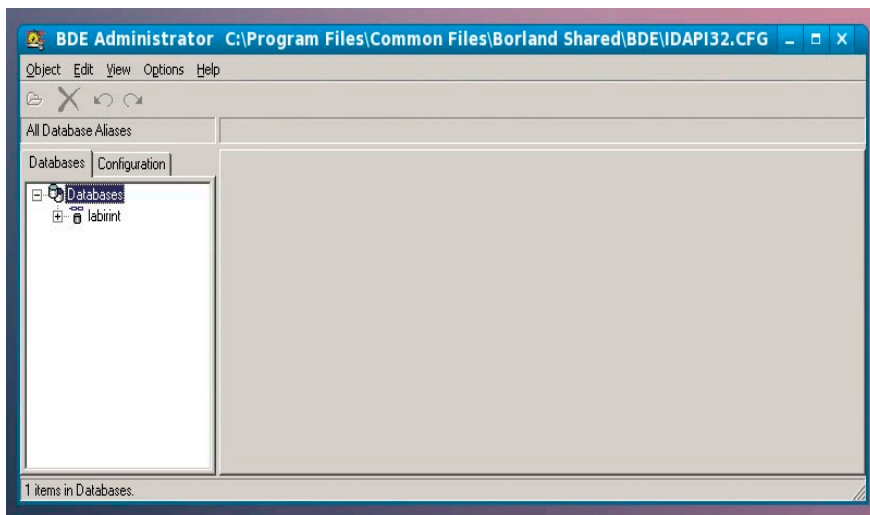


Рисунок 8. Окно BDE Administrator

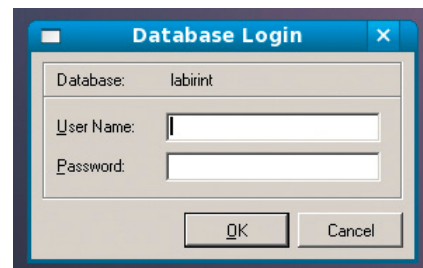


Рисунок 9. Окно ввода логина и пароля для подключения к базе данных

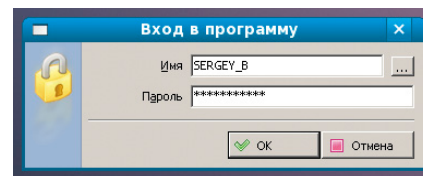


Рисунок 10. Окно выхода в программу



Рисунок 11. Окно, появляющееся после запуска программы «CAMO-Typ»

пасной работы лучше обновить его до последней доступной или, просто, более свежей версии.

Без фанатизма и компилирования это можно сделать командой:

```
# yum update kernel
```

Видим, что ядро установилось:

```
...
Installed: kernel.i686 0:2.6.24.5-85.fc8
Complete!
```

Выполняем перезагрузку, проверяем версию:

```
uname -a
```

```
Linux Newl 2.6.24.5-85.fc8 #1 SMP Sat Apr 19 12:39:34 EDT 2008
i686 i686 i386 GNU/Linux
```

Проверяем работу «CAMO-Typ», изменений в работе не должно быть.

Переходим к следующему пункту.

## Вопросы безопасности

Если помните, то в самом начале мы прописывали прокси сервер в переменных окружения, а также могли запустить программу isql, передав ей логин и пароль в командной строке.

Чтобы эта информация никому не досталась – удаляем файлы /root/.bash\_history и .bash\_history в домашней директории пользователя, от которого мы работали. Удалять файлы целиком совсем не обязательно, достаточно найти в них нужные строчки и затереть их, т.к. история команд

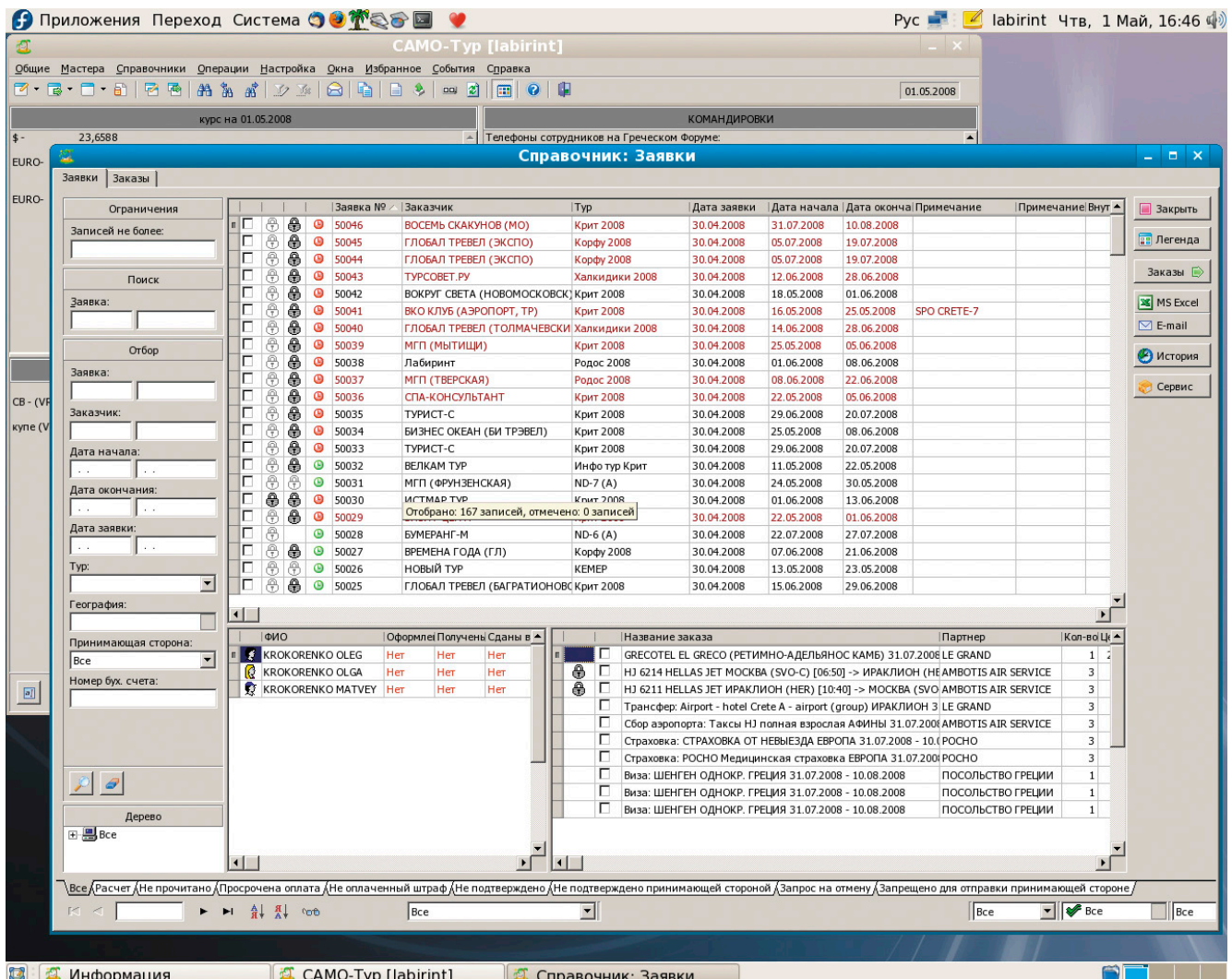


Рисунок 12. Снимок экрана, сделанный во время работы программы «CAMO-Typ» под Linux

может быть очень полезной вещью для забывчивых пользователей и администратора.

Так как мы перезагрузились, то переменная окружения `http_proxu` у нас не сохранилась, если же вы этого не делали, то посмотреть установленные переменные можно командой `set`, а удалить переменную с помощью `unset`.

## О «глюках» в «САМО-Туре», или Ложка дёгтя

Несмотря на то что программа запускается и стабильно работает, типичную русскую ситуацию «обработать напильником до принятия нужной формы» никто не отменял. Есть несколько проблем.

Первая проблема состоит в том, что программа изначально была «заточена» под использование Microsoft Office, а её разработчики так увлеклись «интегрированием» своего продукта в чужеродный, что не заметили изменившихся потребностей рынка и в спешке были вынуждены адаптировать «САМО-Тур» к OpenOffice. После чего в версии под Windows у каждого пользователя в настройках появилась долгожданная возможность выбора (см. рис. 13).

Для того чтобы «САМО-Тур» под wine увидел OpenOffice в Linux, пришлось смириться, что wine-0.9.58-1.fc8 ещё не готов предоставлять такую интеграцию и поставить ещё одну windows-версию OpenOffice под эмулятором. Надо сказать, что для OpenOffice это вообще не проблема, установка проходит несколькими щелчками мыши, запуском одной команды:

```
wine Oo_2.4.0_Win32Intel_install_ru_infra.exe
```

К сожалению, код экспорта из программы «САМО-Тур» в самой программе встречается несколько раз, и все эти места корректно переписывать у разработчиков «САМО-Тура» нет желания, поэтому зачастую можно увидеть ошибку «Variant does not reference an automation object» (см. рис. 14).

Что в этом случае делать, не знаю, т.к. разработчики «САМО-Тура» помогают отказываются. Предполагаю, что побороть это можно и другим путём, с помощью wine, если в последнем появится поддержка «родного» Open Office под эмулятором.

## Заключение

Запуск «САМО-Тура» под Linux – это большое достижение на рынке туператоров. Это последний камень преткновения при переходе на ОС Linux, т.к. для всех других пользовательских приложений давно существуют не менее функциональные аналоги.

Также это большое достижение для моей фирмы в вопросе экономии и перевода всего программного обеспечения на Open Source.

Перед Новым годом нам пришлось закупить с десяток новых ноутбуков, которые мы вынуждены были приобрести с установленной ОС Windows Vista, т.е. заплатить за неё, а после стереть и не использовать. Очень жаль, что производители не предоставляют широкий выбор ноутбуков без ОС.

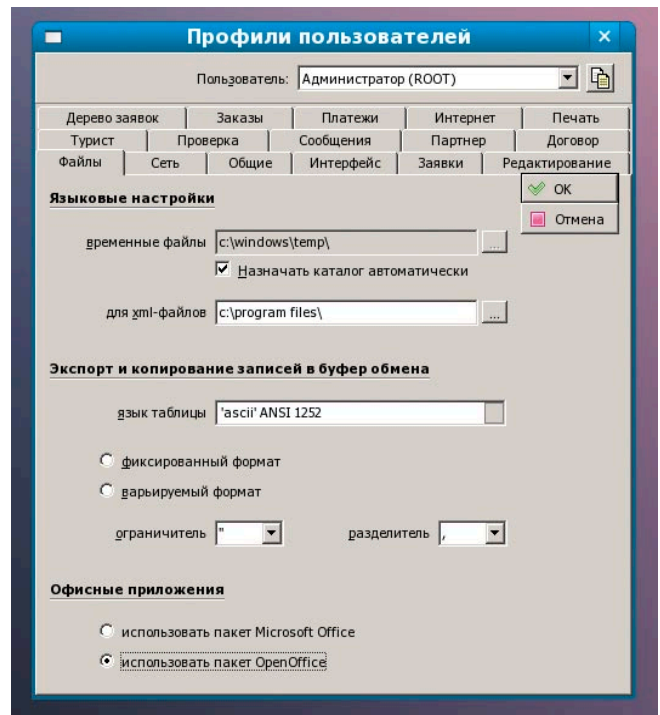


Рисунок 13. Окно настройки профиля пользователя с возможностью выбора параметров экспорта в OpenOffice

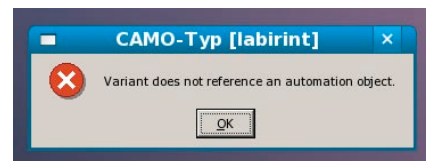


Рисунок 14. Окно ошибки «Variant does not reference an automation object», возникающей при попытке экспорта данных в Open OfficeCalc под эмулятором wine

Также следует отметить, что миграция пользователей в нашей фирме проходит довольно легко, т.к. набор используемых менеджерами программ практически кроссплатформенный: Mozilla Firefox, Thunderbird, Sunbird, а теперь и «САМО-Тур».

Обсудить возникающие вопросы всегда можно на форуме <http://www.samag.ru/forum>.

1. Страница продукта «Мастер-Тур» – <http://www.megatec.ru/?m=60>.
2. Тема форума о портировании «Мастер-Тура» под Linux – <http://crmportal.megatec.ru/Default.aspx?tabid=53&forumid=6&threadid=254&scope=posts>.
3. Страница продукта «САМО-Тур» – <http://samo.ru/index.php?module=products&showitem=2>.
4. Тема форума о портировании «САМО-Тура» под Linux – <http://forum.samo.ru/index.php?s=e9056965fd0ea9773dfc20aa2bd1ac84&showtopic=4051>.
5. Репозиторий rpm.livna.org, содержащий дополнительные пакеты для Fedora Linux – <http://rpm.livna.org/rllwiki>.
6. Проект FreeTDS, способ обращения из под UNIX и Linux к серверам MS SQL – <http://freetds.org>.
7. Эмулятор Wine HQ – <http://www.winehq.org>.
8. Руководство по ODBC и MyODBC, а также пример использования FreeTDS – <http://www.citforum.ru/database/mysql/odbc/unixodbc.shtml>.