Компилируем исполняемый файл на Харборе **[1]**

Admin

В процессе работы с БЭСТ-4+ иногда возникают задачи, решать которые плагинами или пунктами меню либо неудобно, либо небезопасно. Как правило, это могут быть задачи административного характера или утилитарного назначения. В этом случае на помощь может прийти полноценный запускаемый файл, скомпилированный самостоятельно.

Как это сделать и что для этого надо?

Постараюсь ответить на эти вопросы. Сразу оговорюсь - я постараюсь избежать теоретических выкладок и подробного рассмотрения каждой опции. Новичку эти подробности навряд ли помогут, а корифеи лишь лукаво улыбнуться ;) .

Для формирования exe файла нам понадобится компилятор **Borland C++ 5.5.1 for Win32**. Скачать его можно на сайте разработчика [2].
Распаковав архив в любое удобное место (обычно каталог Install\c++) запускаем инсталлятор. По умолчанию, будет создан каталог C:\Borland\Bcc55.
Далее следуем инструкциям, приведенным в readme.txt.

1. В переменную окружения PATH добавим - ;C:\Borland\Bcc55\Bin

2. Создадим в каталоге C:\Borland\Bcc55\Bin файл bcc32.cfg следующего содержания:
   (Здесь следует заметить, что лучше всего использовать Far Manager, т.к. операции по созданию файлов, смене кодировки файла, быстрее и проще делать в нем.)
   -I"c:\Borland\Bcc55\include"
   -L"c:\Borland\Bcc55\lib"

3. Создадим в каталоге bin файл ilink32.cfg следующего содержания:
   -L"c:\Borland\Bcc55\lib;c:\Borland\Bcc55\lib\PSDK"

На этом настроечный этап можно считать завершеным.
Для проверки того, насколько правильно мы установили C++ - можно скомпиллировать набор примеров, идущих в поставке. Для этого переходим в каталог c:\Borland\Bcc55\Examples\StdLib и собираем набор примеров командой make.
Можно поступить еще проще: создать текстовый файл test1.cpp (не в WORD ! и в DOS-кодировке) содержания:

```
#include <stdio.h>
int main ()
{
printf("Мой первый опыт на C++");
return 0;
}
```

Затем скомпиллировать коммандой

```
bcc32 test1.cpp
```

В итоге получаем test1.exe файл, проверяем на запуск и убеждаемся в работоспособности установленного нами C++.
Поздравляю! Первый этап завершен.

Приступаем к установке xHarbour. Следует отметить, что стандартный xHarbour, скомпилированный из официальных исходников проекта xHarbour нас не устроит. Специальную версию, "понимающую" базы БЭСТ-4+, можно скачать здесь [3]
Нам остается только раскрыть архив в любое удобное место, к примеру C:\xHarbour.
Теперь приступаем к тестированию.
В каталоге для своих проектов, скажем C:\Hrb_Src открываем каталог для первого примера C:\Hrb_Src\Lesson1.
Формируем test1.prg (DOS-кодировка):

```
Function Main ()
? "Мой первый опыт на xHarbour и C++"
Return NIL
```

Создаем bat файл для сборки нашего "проекта" b.bat:

```
@echo off
set HB_INSTALL=c:\xharbour
%HB_INSTALL%\bin\harbour  test1.prg -p -b -n -i%HB_INSTALL%\include
bcc32 -c -O2 -tW -M -DHB_MULTI_GT -I%HB_INSTALL%\include  test1.c
IF EXIST  test1.rc brc32 -r  test1
echo c0w32.obj + > b32.bc
echo  test1.obj, + >> b32.bc
echo  test1.exe, + >> b32.bc
echo  test1.map, + >> b32.bc
echo %HB_INSTALL%\lib\rtl.lib + >> b32.bc
echo %HB_INSTALL%\lib\vm.lib + >> b32.bc
echo %HB_INSTALL%\lib\gtwvt.lib + >> b32.bc
echo %HB_INSTALL%\lib\gtwin.lib + >> b32.bc
echo %HB_INSTALL%\lib\gtnul.lib + >> b32.bc
echo %HB_INSTALL%\lib\optgui.lib + >> b32.bc
echo %HB_INSTALL%\lib\lang.lib + >> b32.bc
echo %HB_INSTALL%\lib\codepage.lib + >> b32.bc
echo %HB_INSTALL%\lib\macro.lib + >> b32.bc
echo %HB_INSTALL%\lib\rdd.lib + >> b32.bc
echo %HB_INSTALL%\lib\dbfntx.lib + >> b32.bc
echo %HB_INSTALL%\lib\dbfcdx.lib + >> b32.bc
echo %HB_INSTALL%\lib\dbffpt.lib + >> b32.bc
echo %HB_INSTALL%\lib\dbfdbt.lib + >> b32.bc
echo %HB_INSTALL%\lib\bcc640mt.lib + >> b32.bc
echo %HB_INSTALL%\lib\common.lib + >> b32.bc
echo %HB_INSTALL%\lib\debug.lib + >> b32.bc
echo %HB_INSTALL%\lib\hbct.lib + >> b32.bc
echo cw32.lib + >> b32.bc
echo import32.lib, >> b32.bc
if exist  test1.res echo  test1.res >> b32.bc
ilink32 -Gn -Tpe -ap @b32.bc
```

Цветом выделены строки, обязательные к редактированию для своего проекта.
Запустив b.bat мы должны получить полноценный программный файл test1.exe.
О том, как эту возможность использовать - можно обсудить на форуме [4]

Продолжаем знакомство с xHarbour

Прежде всего, для работы с БД БЭСТ-4+ в наш тестовый файл, НЕОБХОДИМО добавить следующие строки:

```
REQUEST HB_CODEPAGE_RU866,HB_LANG_RU866
Request dbSetFilter
```

```
REQUEST DBFCDX,DBFFPT
```

в тело функции function main()

```
   RDDSetDefault('DBFCDX')
   SET( _SET_AUTOPEN, .F. )
   SET( _SET_DELETED, .T. )
   hb_SetCodepage( "RU866" )
   Set Date FRENCH
   Set EXACT ON
```

В качестве примера, напишем простенький просмотрщик баз данных. Не будем рисковать сразу на рабочих базах, достаточно будет скопировать таблицу в каталог с программой, скажем partner.dbf & partner.cdx.
Из действий определяем - открытие и просмотр с параметрами по умолчанию.

```
use partner.dbf
browse()
use
```

Итоговый код:

```
REQUEST HB_CODEPAGE_RU866,HB_LANG_RU866
Request dbSetFilter
REQUEST DBFCDX,DBFFPT

Function Main ()

RDDSetDefault('DBFCDX')
SET( _SET_AUTOPEN, .F. )
SET( _SET_DELETED, .T. )
hb_SetCodepage( "RU866" )
Set Date FRENCH
Set EXACT ON

use partner.dbf
browse()
use

Return NIL
```

Компилируем файл через bat файл, из предыдущей статьи. В глаза бросается большое количество файлов, формируемых при компиляции. Избавимся от них добавив в bat Файл:

```
del *.tds
del test1.c
del test1.map
del test1.obj
del b32.bc
```

Теперь запускаем bat файл, и проверяем работоспособность нашей программки.
Что мы теперь имеем?
В наших руках появился инструмент для работы с базами данных БЭСТ-4+ (правда, осталось проверить работу с memo-полями)).

Первое применение, которое я нашел для себя - это написал установочную утилиту для hrb плагинов. Порой, плагин может применяться боле чем в одном модуле и проще написать программу, чем довериться файлу readme.txt и пользователю.
Существенным недостатком является отсутствие стандартных функций БЭСТ-4+. То есть "быстренько" сформировать модуль в стиле БЭСТ-4+ будет проблематично, зато реализовать алгоритм массовой обработки данных, или выполнить другую задачу стандартными средствами - будет легко.

Успехов.

Тип материала: Практикум [5]

Источник (modified on 31/10/2014 - 09:57): https://spb4plus.ru/praktikum/kompiliruem-ispolnyaemyy-fayl-na-harbore

Ссылки
[1] https://spb4plus.ru/praktikum/kompiliruem-ispolnyaemyy-fayl-na-harbore
[2] http://www.borland.com/downloads/download_cbuilder.html
[3] http://www.itman.tulpar.net/modules/mydownloads/visit.php?lid=4
[4] http://www.bestnet.ru/club/phpBB/viewtopic.php?t=9383
[5] https://spb4plus.ru/kategoriya/praktika