

Для компиляции ресурсов, используемых в приложении, таких как курсоры, изображения, иконки в динамически связанную библиотеку **DLL**, применяется следующая технология:

1. составляется список ресурсов в текстовом файле с расширением ***.RC**,
2. этот файл, с помощью утилиты `brcc32.exe`, компилируется в файл ресурса, с расширением ***.RES**,
3. затем этот файл в среде Delphi компилируется в библиотеку **DLL**

Разберем детально этот механизм.

Сначала в обычном текстовом редакторе, например в Блокноте, создадим текстовый файл с расширением ***.RC**, представляющего собой файл описания ресурсов. Файл содержит следующие строки вида:

например,

```
image1 RCDATA "picture.jpg"
```

где строка описания ресурса содержит: имя ресурса - *image1*, тип ресурса - **RCDATA**, с расположением ресурса `picture.jpg` в этом же каталоге, т.к. путь к ресурсу не указывается.

Тип ресурса **RCDATA** используется для пользовательских ресурсов, для курсоров указывается тип **CURSOR**, для иконок - **ICON**, для изображений формата **BMP** - **BITMAP**

Например:

```
new BITMAP "new.bmp" finger CURSOR "cursor1.cur" tray ICON "icon15.ico"
```

В Delphi поддерживаются следующие типы ресурсов

RT_ACCELERATOR - таблица ускорителей

RT_ANICURSOR - анимированный курсор

RT_ANIICON - анимированная икона

RT_DIALOG - диалоговое окно

RT_FONT - шрифт

RT_FONTDIR - набор шрифтов

RT_GROUP_CURSOR - коллекция курсоров

RT_GROUP_ICON - коллекция иконок

RT_HTML - HTML документ

RT_MENU - меню

RT_MESSAGETABLE - таблица сообщений

RT_STRING - таблица строк

RT_VERSION - ресурс версия

Т.е. если вы хотите включить в библиотеку шрифт, то в файле **RC** указывается тип **FON**
T .

Ресурсы, включенные в файл описания ресурсов *. **RC** , затем нужно скомпилировать в
файл ресурсов с расширением **RE**
S

при помощи родного компилятора ресурсов Delphi - brcc32.exe (Borland Resource
CommandLine Compiler 32 bit) , который находится в каталоге BorlandDelphiXBin , где X -
версия Delphi . Утилита работает в режиме командной строки. Компиляция файла
example.RC будет выглядеть следующим образом:

```
c:\Проекты\brcc32.exe example.RC
```

После компиляции в каталоге c:\Проекты появится файл example.RES.

Затем в среде Delphi создаем библиотеку **DLL** , куда включаем наш файл ресурсов
example.RES.

Для создания библиотеки **DLL** в среде Delphi меню *File* выбираем пункт *New* и пункт
подменю *her..* От

В появившемся окне выбираем тип проекта **DLL Wizard** , что представляет собой
нужную нам библиотеку **DLL**
создав проект, указываем компилятору включать наш ресурс в директиве {\$R}. В
листинге №1 приводится код проекта
Project1

```
library Project1; uses SysUtils, Classes; {$R example.RES} begin end.
```

Затем проект компилируем в файл **DLL** , выбрав в меню *Project* пункт *Build*.

Получаем файл **DLL** с одноименным названием проекта Project1. Загрузка нашей картинки **JPG** из библиотеки **DLL** осуществляется следующим образом. Сначала загружаем **DLL** в память (1), создаем поток для конкретного ресурса (2), извлекаемого из библиотеки, создаем объект изображения (3), производим загрузку ресурса в поток (4), освобождаем дескрипторы потока и библиотеки (5).

```
var h: THandle; imageStream: TResourceStream; Image:TImage; begin h:=
LoadLibrary('Project1.Dll'); (1)
imageStream:=TResourceStream.Create(h,'image1',rt_RCData);(2)
Image.Picture.Graphic:=TJPEGImage.Create; (3)
Image.Picture.Graphic.LoadFromStream(ResStream); (4) ResStream.Free; FreeLibrary(h);
(5) end;
```

Учитывая особенности извлекаемых из **DLL** ресурсов, создается тот или иной объект, т.е. для растрового изображения типа BMP создается объект

TBitmap

и т.д.