Tutorial 4
Copyright (c), Freeman-Teresa Software, 1995
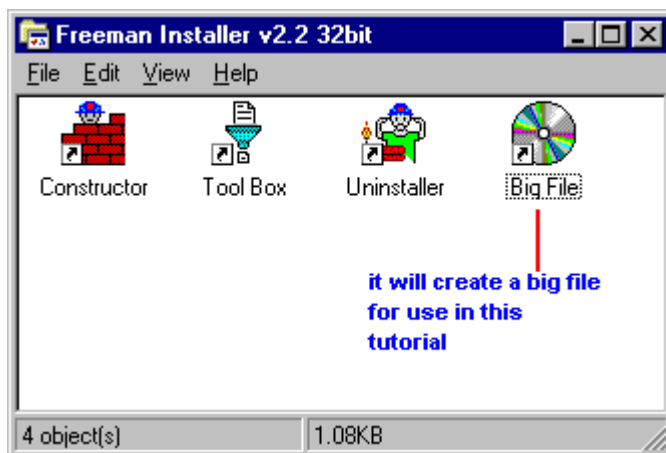
What topics are covered?
- Installing an extra-ordinarily large file, spanning across disks.
- Installing true type fonts.
- Inserting your own C++ code for simple customizations.

Step 1
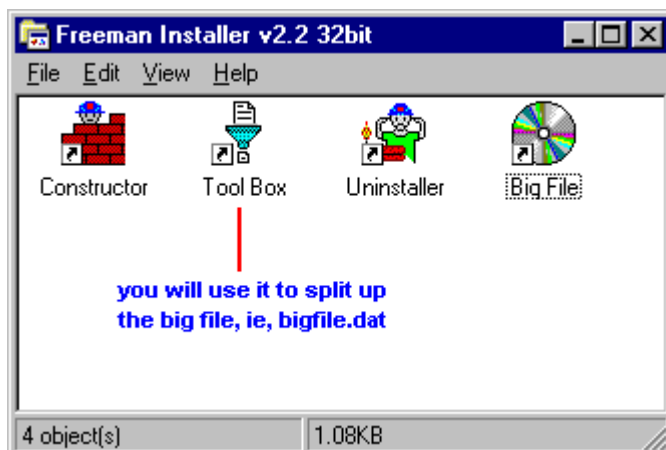Open c:\prj\install.inf. You will base this tutorial on the last one.

Step 2
Run "Big File" to create a big file c:\finstall\bigfile.dat (note that by c:\finstall we mean the directory of Freeman Installer itself which is c:\fi22w4 for this particular version). You must be patient since it takes quite a while to create bigfile.dat.
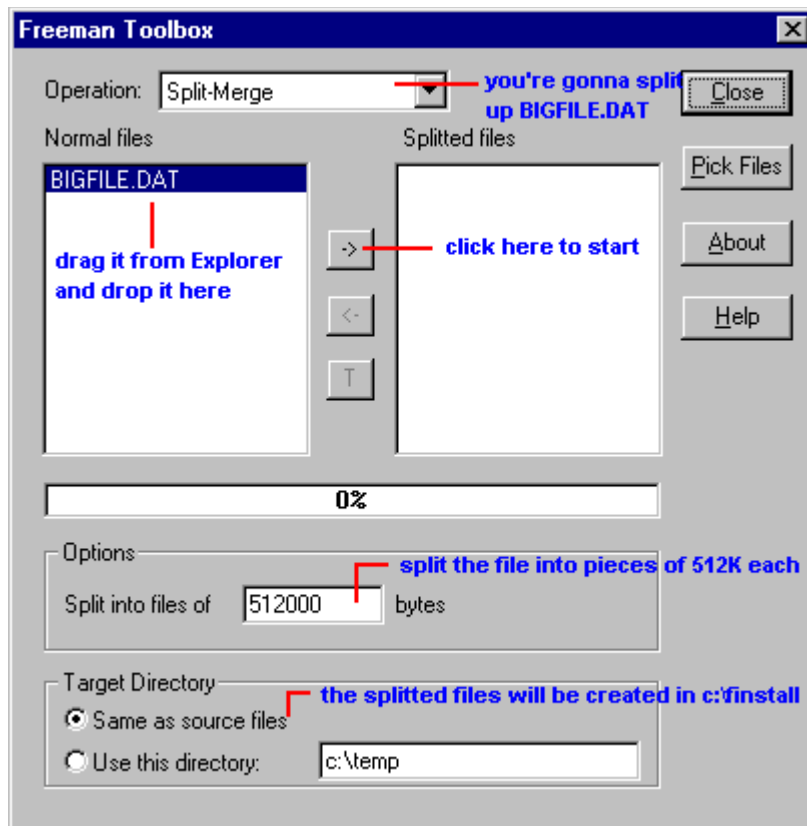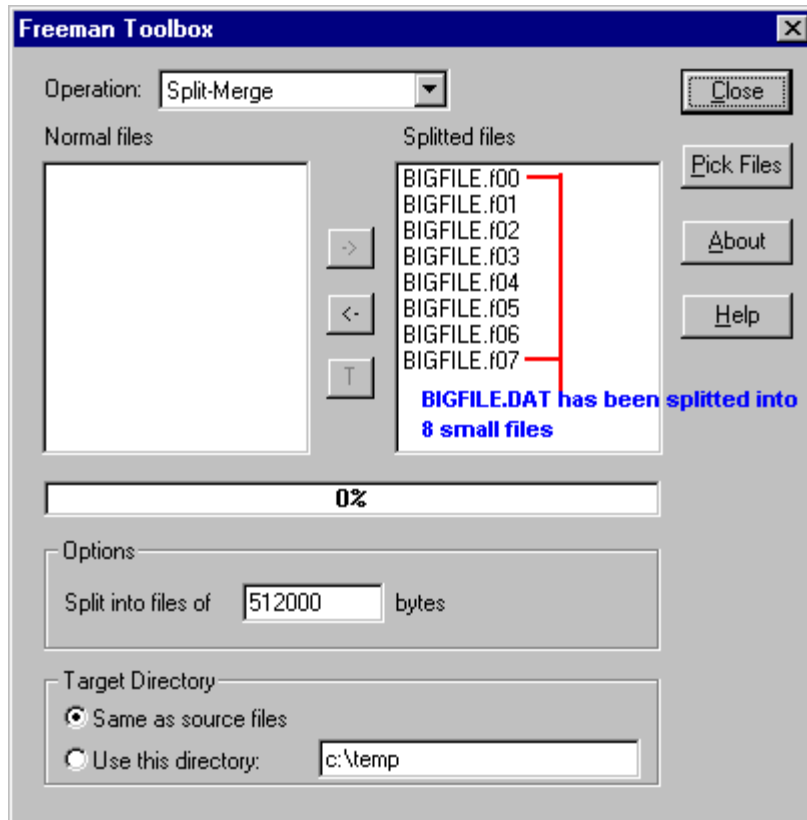

Step 3
Run Freeman Toolbox:

Click on "Pick Files" to go to the Explorer (or File Manager) to select c:\finstall\bigfile.dat and drop it into the "Normal files" list box. Then set the parameters as shown below:

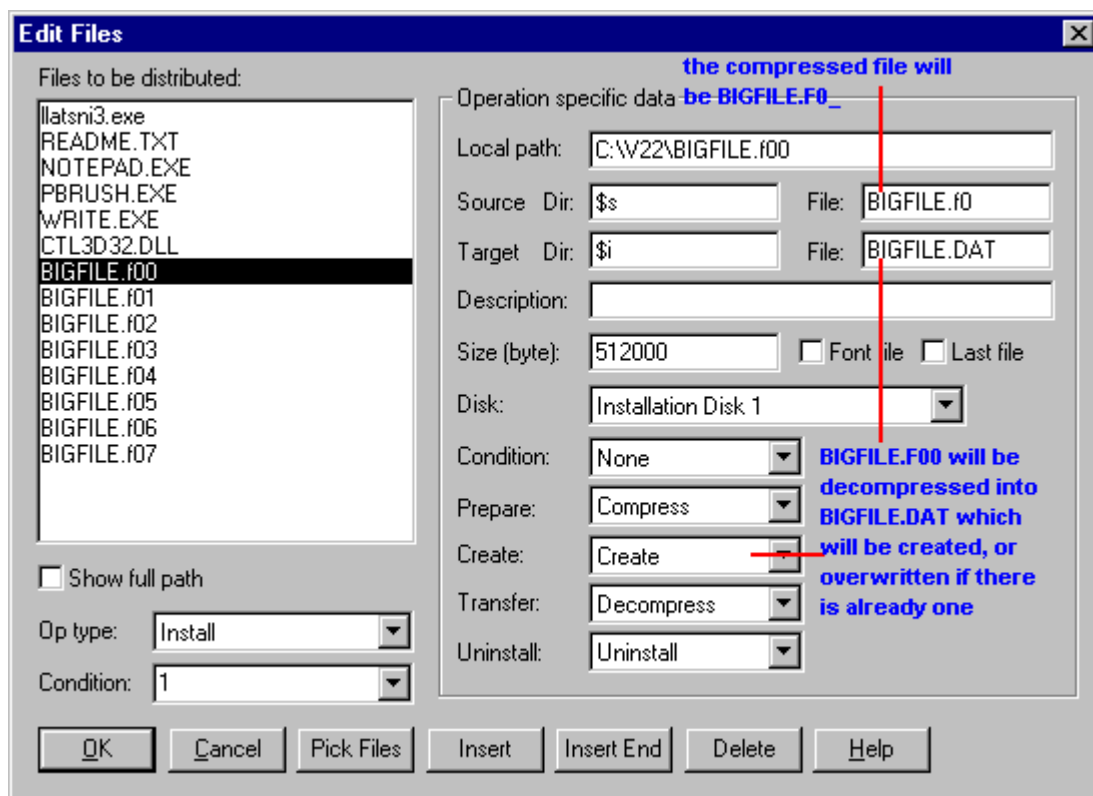Click on the "->" to start. The result will look like:

Click on "Close" button to exit Freeman Toolbox.

Then you can delete c:\finstall\bigfile.dat to save disk space.

Step 4

Go to the file list and add the splitted files (bigfile.f00, bigfile.f01, ..., bigfile.f07) to it. Make sure they are in exactly this order. If you don't interfere with the constructor, the compressed files for these splitted files will all be the same, i.e., bigfile.f0_. To resolve this name conflict, you can change the source file name for each of them as shown below:
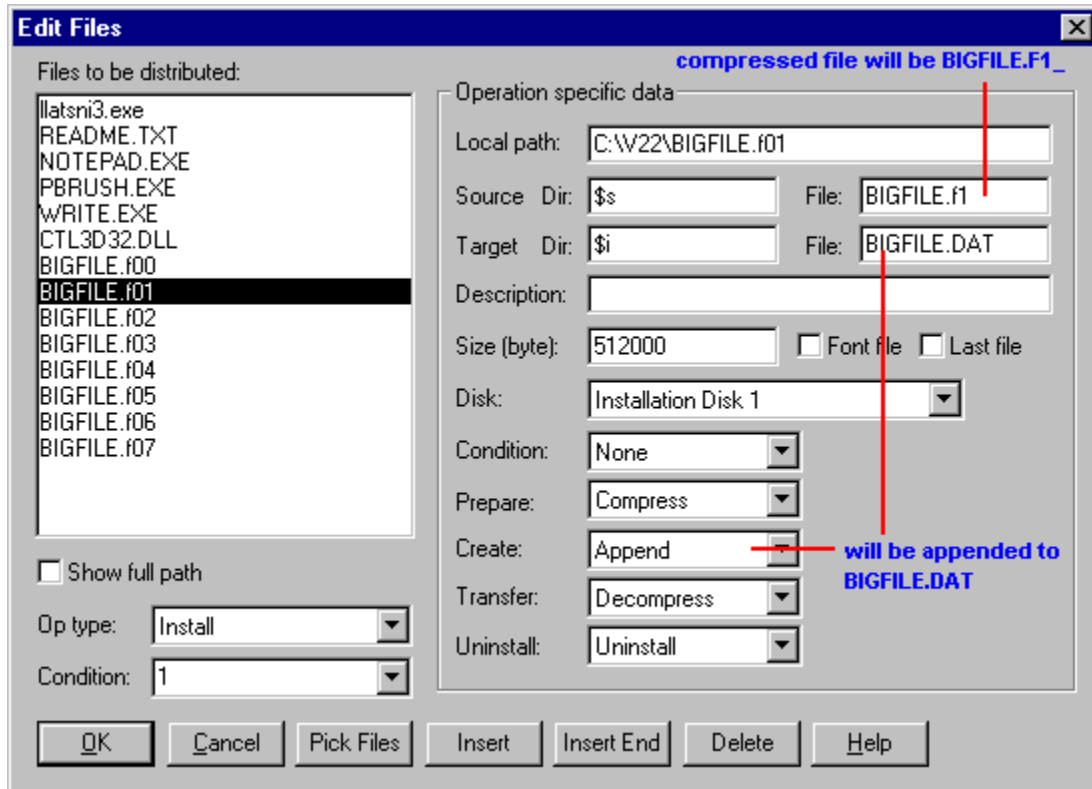

Note that because bigfile.f00 is the first part of bigfile.dat, you should use bigfile.f00 to CREATE bigfile.dat:
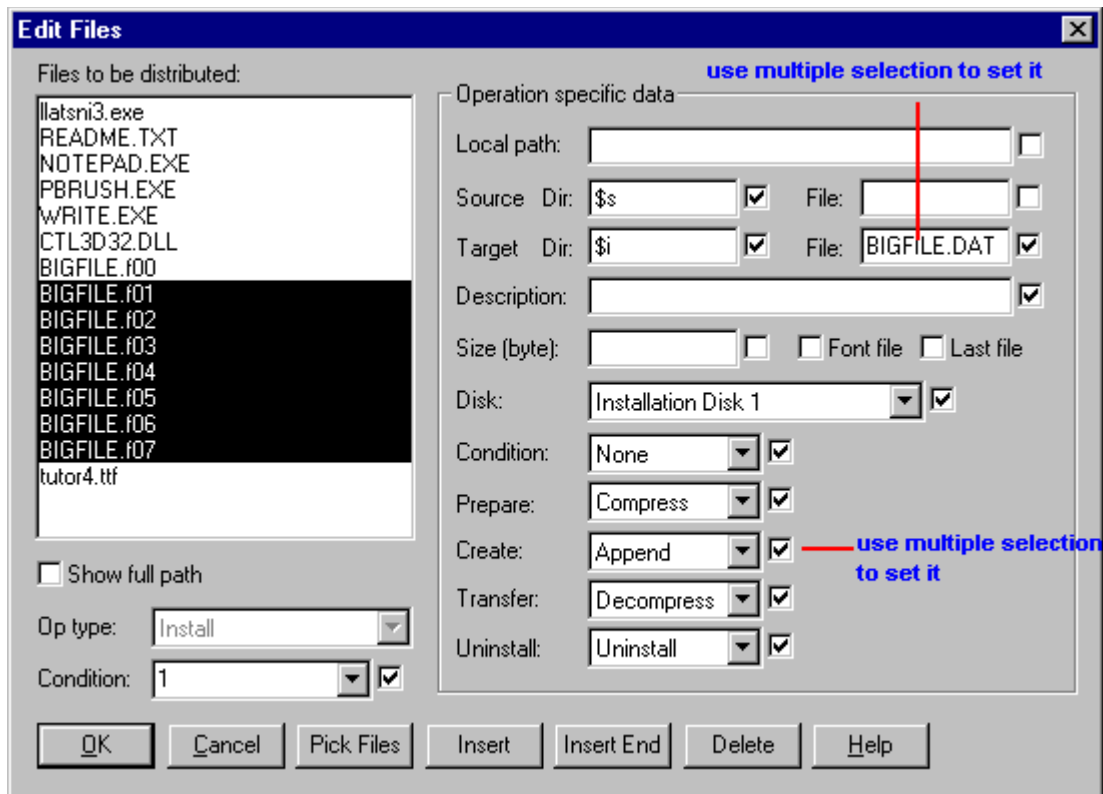
Prepare:	Compress
Create:	Create
Transfer:	Decompress

For all other parts, you will APPEND them to the bigfile.dat created by bigfile.f00:

Prepare:	Compress
Create:	Append
Transfer:	Decompress

You need to do something similar to bigfile.f02, bigfile.f03, ..., and bigfile.f07. To modify the source file name to avoid compressed name conflict you have to edit the entries one by one. To set "Create" to "Append" and set the target file name to "bigfile.dat", you can use multiple selection:

Click on "OK" button.

Step 5
Suppose that bigfile.dat is required by paint brush and as such should be included in all components that include paint brush. Go to the component list to do it now.
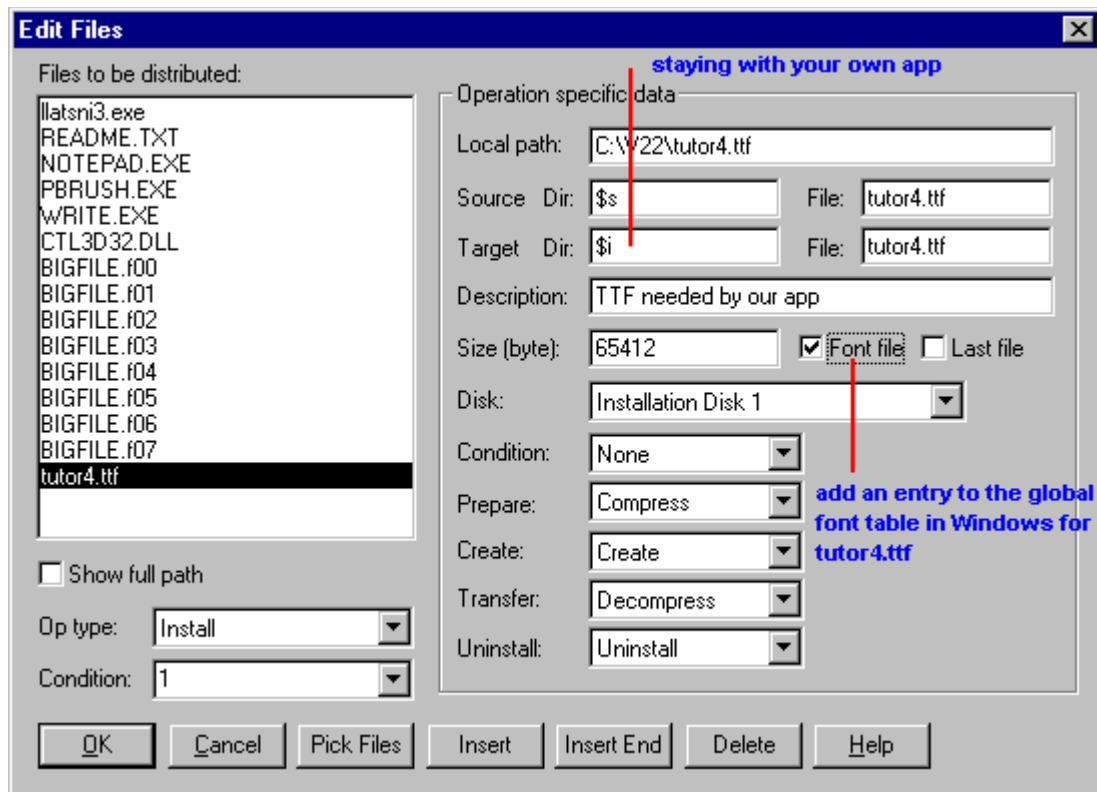
Step 6
Now you are going to install a true type font. First you need to make a "new" true type font called "tutorial 4". If you're running Win95:

```
c:\>copy c:\windows\fonts\arial.ttf c:\finstall\tutor4.ttf
```
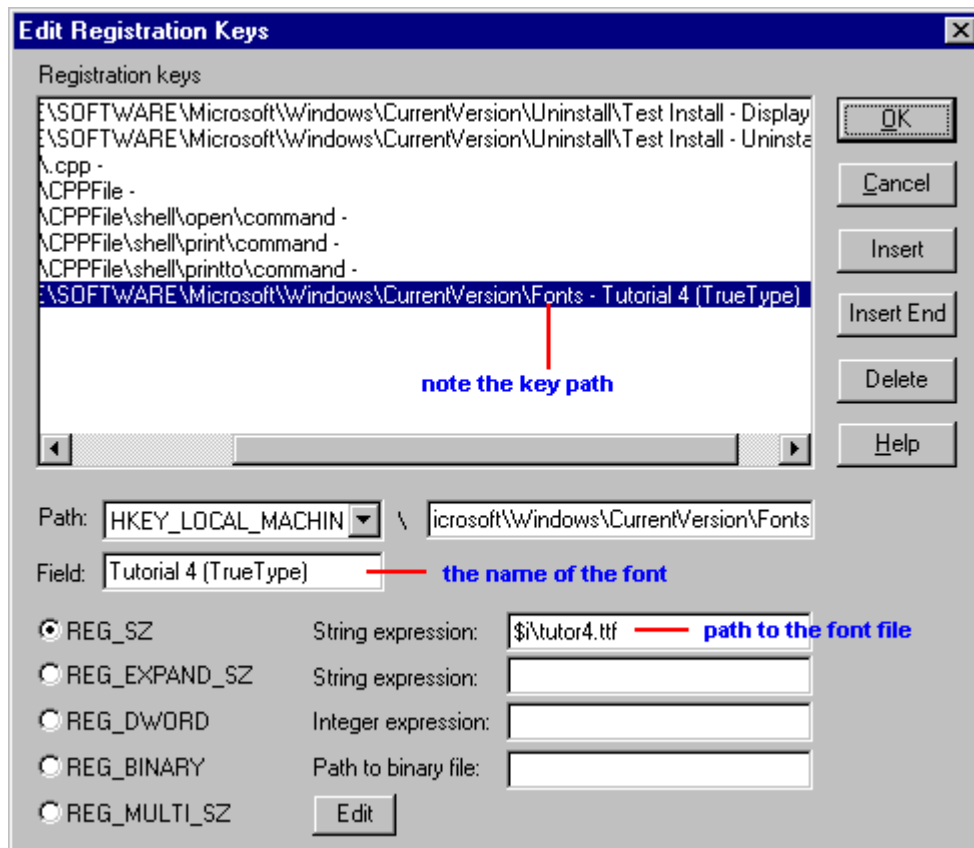
If you're running NT:

```
c:\>copy c:\windows\system\arial.ttf c:\finstall\tutor4.ttf
```

Go to the file list and add c:\finstall\tutor4.ttf to it. Suppose that the font will be so installed that it will be available to your own application only. Therefore the target directory should be set to $i. Then check the "font file" checkbox as shown below:

Step 7
Go to Edit | Registration Keys and add a new entry as shown below:

If your target platform is NT, the key path should be:

```
HKEY_LOCAL_MACHINE\Microsoft\Windows NT\CurrentVersion\Fonts
```

instead of:

```
HKEY_LOCAL_MACHINE\Microsoft\Windows\CurrentVersion\Fonts
```
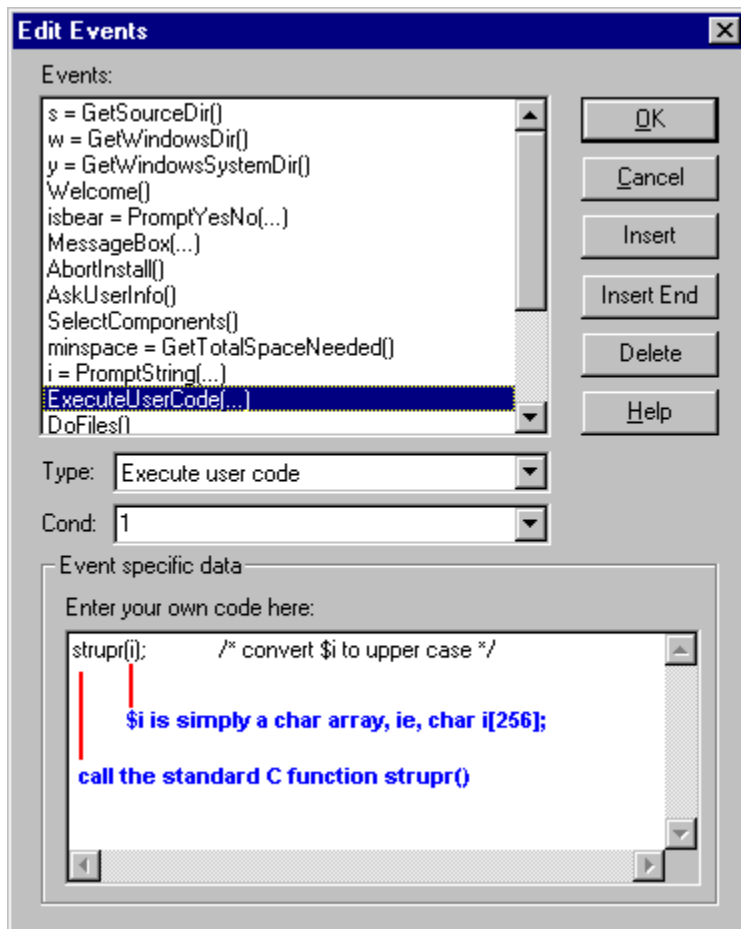
Step 8
Include arial.ttf and that registration key entry in the full install component.

Step 9
Now, you are going to convert $i to upper case. While Freeman Installer has no built-in support for this, it does provide you with easy access to the C/C++ language facility. If you don't have a C++ compiler, please go to step 10 now.

Go to the event list and add a "user code" event as shown below:

In a user code event you can write any code as long as it is a segment of correct C/C++ code. To access the string variables such as $i, $w, $y, $s and etc, you can simply treat them as variables of the type "char [256]". To access integer variables you can simply treat them as variables of the type "int". If you need to access standard C RTL functions that haven't been declared, you should #include the header file in c:\prj\fiuser.h. If you need to enter pages of code in a user code event, you should consider putting the code in a function in your own .cpp file and make a call to it in the user code event. For more information on how to do this, refer to the Q&A.
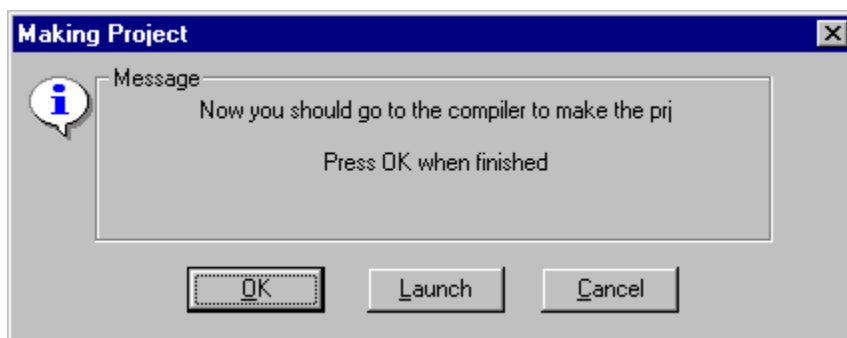
Step 10
You are ready for a test run.

Usually you can choose between the "Interpret" commands and the "Compile" commands according to your preference. "Interpret" commands are more convenient to use, give faster feed back. "Compile" commands give you a smaller installer. However, when you insert your own C/C++ code, you have no choice, you MUST use the "Compile" commands.

OK, hit Compile | Run.

When you see the following message box, click on the "Launch" button to try to launch the compiler and load the project. Some compilers may refuse to run if an instance of them is already running, in that case you will have to go to your compiler's IDE and load the project/make file yourself (c:\prj\usr\fimainc.ide or fimainc.mak, depending on which version of which compiler you are using).
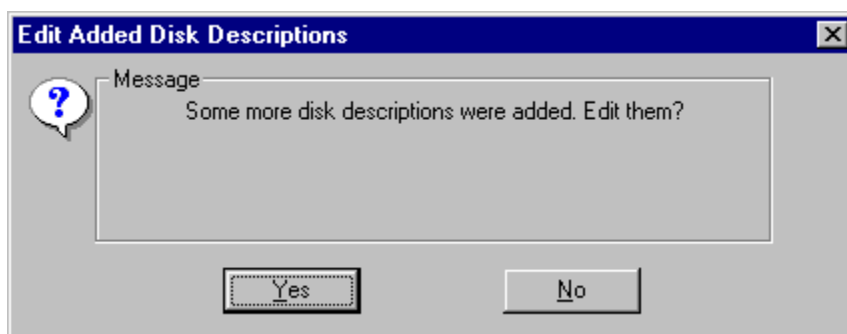

When you are in the compiler IDE, "make" the project. You may have to set the include directory and library directory in the IDE, since we don't know in advance their locations on your hard disk. When it is finished, go back to Freeman Constructor and click on "OK" to continue the test run.

After the installation the "new" font will NOT be available in your applications. This is because Windows is smart enough to know it is the same as Arial. For your own fonts it should work just fine.
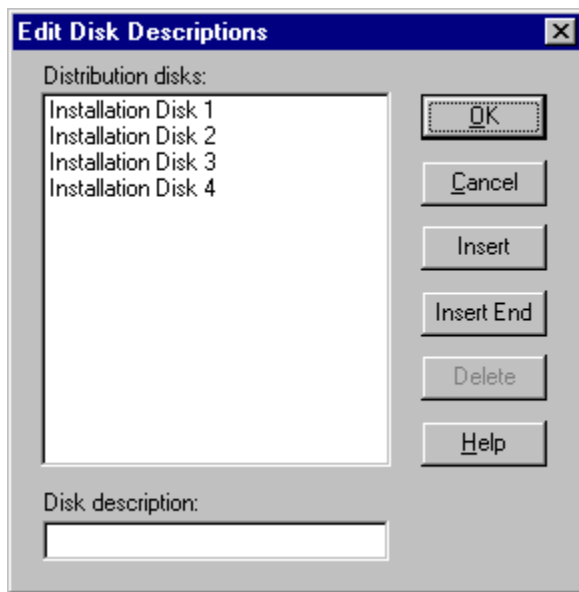
Step 11
Now, build disk set, purify and zip the install.

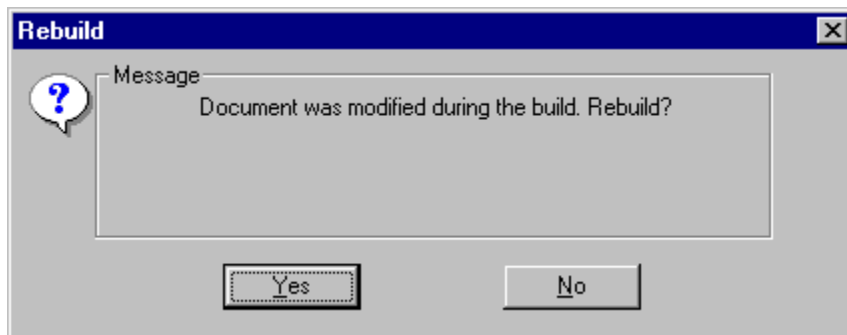When the disk set building is finished, you will see a message box shown below:

This is because the package spans across multiple disks but you only had one disk description. In this case, the constructor has added three more disk descriptions for you and

now it asks you to edit them. Click on "Yes".

If you are OK with these descriptions, click on "OK". Otherwise edit them at wish.

Due to the change of the disk description for some files, you need to rebuild the package. Click on "Yes" to start the rebuild.

Congratulations! You have finished tutorial 4.