
Subject: Re: multiple classes include-problem
Posted by [bytefield](#) on Tue, 26 Feb 2008 15:54:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

mrjt wrote on Tue, 26 February 2008 17:39 Maybe I wasn't clear enough, I believe the correct solution is:

In .h file:

```
#define IMAGECLASS SomImg  
#define IMAGEFILE <SomImg/SomImg.iml>  
#include <Draw/iml_header.h>
```

In ONE .cpp file:

```
#define IMAGECLASS SomImg  
#define IMAGEFILE <SomImg/SomImg.iml>  
#include <Draw/iml_source.h>
```

The way the image stuff works is that the macros create a class with functions that return the image, so naturally it is best to have the header and source for this class in .h/.cpp files.

Yeah, know that but if we have more sources and still want to use same *.iml file, what we do? I cannot put all my code in 2 files (header and source).

An example:

main.h - contain the generally includes, some global functions declarations, etc.

MyApp.h - MyApp class which implement main windows (it use some images from *.iml file)

MyApp.cpp - MyApp implementation and GUI_APP_MAIN

MyDialog.h - a dialog (complex) or something

MyDialog.cpp - dialog implementation

ConfigDlg.h - ...

ConfigDlg.cpp - ...

So i need images in every class (in every source). Is there another solution or mine is not so dirty? Or perhaps others use just 2 files (*.h/*.cpp) to write an application?

I don't want 10000 lines of code or more to share the same file. Should i implement the code which use images in a single source then, and other keep separated?