

## Tutorial: Turning your python Ogre game scripts into a Windows .exe

Troy Sedgwick  
CS381: Game pipeline  
Department of Computer Science and Engineering  
University of Nevada, Reno  
[tn.sedgwick@gmail.com](mailto:tn.sedgwick@gmail.com)

This is a tutorial to show you how to turn your python Ogre game scripts into a Windows executable file.

Requirements:

- Python 2.6 and python Ogre 1.6.4 (different versions might work too, but could require extra steps that this tutorial does not provide)

-py2exe (<http://www.py2exe.org/>)

- Basic Knowledge of python and python Ogre setup/programming

Follow the steps below:

1. If you haven't already, download and install py2exe from SourceForge:  
(<http://sourceforge.net/projects/py2exe/files/>)
2. Make a setup.py file (either read the online tutorial from py2exe's main site or use the setup file I already created).

Here is the code for the setup.py I made:

```
import sys
from distutils.core import setup
import py2exe
from glob import glob

#Make sure setup knows where all of your
#program modules are, modify dirs below as needed
sys.path.append('.')
sys.path.append('.')
sys.path.append('.')
sys.path.append('.')
sys.path.append('.')
sys.path.append('.')
sys.path.append('.')
sys.path.append('.')

#Link Microsoft visual C runtimes...
#again, modify dir as needed
data_files = [("MicrosoftVisualCRunTimes", glob(r'.\redis*.'))]

setup(data_files=data_files,console=['main.py'])
```

3. Make sure the setup file is in the same dir as your game's main script.
4. Now run the setup.py in the command prompt as: "python setup.py py2exe"
5. Okay, unless you're really lucky, you should be getting a whole bunch of ".dll" missing errors (which will stop the py2exe build). In order to fix this, all you need to do is find the .dll's that are reported missing and paste them into your " python-root-dir/DLL's" dir (or in the same dir as your setup.py). All of the .dll's you need (for python Ogre at least) are found in python Ogre's root dir. Just do a search of the dir and they should come up.
6. In order to build and run an .exe with py2exe your machine needs access to the Microsoft Visual C runtime .dll's. If you installed python or Visual Studio and either of these two programs work, the .dll's are present on your machine. The problem is that if you wish to distribute your .exe to users, these users might not have the .dll's on their machines. In order to fix this you need to link the .dll's to your .exe (lines 4 and 18 in my setup.py) . The trick is though, is that you need to find the folder in which contains these .dll's and point the setup.py to them. What I did is made a "redis" folder in the same dir as my setup.py and used that. You can find the .dll's usually under:  
"Program Files (x86)\Microsoft Visual Studio 9.0\VC\redist\Debug\_NonRedist\x86\Microsoft.VC90.DebugCRT"  
Make sure you copy everything from this folder into your local "redis" folder. Also, make sure your .dll's versions are " 9.0.21022.8" because that is what py2exe uses. If your .dll's are the wrong version or are not in the path I just specified, download Microsoft's Visual C Runtime distributor (vcredist\_x86.exe):  
<http://www.microsoft.com/downloads/details.aspx?FamilyID=9b2da534-3e03-4391-8a4d-074b9f2bc1bf&displaylang=en>  
And install it. This should give you the right .dll's and place them in the path I specified.
7. Okay, now with all of that .dll stuff out of the way, it is downhill from here. Doing "python setup.py py2exe" now should run smoothly. You might get a couple of errors saying it can't find some modules, but ignore them if they have anything to do with python Ogre.
8. Now go to the dir with your setup.py in it and you should see a "dist" folder. Inside this, is your .exe and all of the stuff it needs to run (mainly .dll's). Before your run your .exe, make sure your Ogre configuration and resource files are in "dist" since python Ogre needs these in order to operate.
9. Double click on the .exe to see your game run. If you happen to get any .dll missing errors, just find them and paste them into your "dist" folder. (For example, when I first ran my .exe I get a cg.dll missing, and all I did to fix it was paste cg into my "dist" folder)
10. We can now enjoy our game without dragging along all of the source code..... Yay!!