

Alternative way to distribute Panda3d applications

Panda3d SDK has its own tools for distributing Panda3d applications. Packpanda is an easy way to make an NSIS installer for your game. The Packp3d and Pdeploy are relatively new tools for making an installer or standalone file. I'm not a fan of the Packp3d/Pdeploy and the P3D system at all, however, these are still relatively new tools and have issues/bugs like any other. This is all just my opinion and opinions vary. The purpose of this tutorial is not to convince you which is the best way for you to distribute your app, it just shows an alternative way of doing it. Panda3d applications are C++ and Python applications and can be distributed like any other.

In this tutorial I'm going to show you how to make an archive containing the main exe and data files of your game. You can later use NSIS or any other similar tool to generate an installer, however I will not teach you how to do that. If you want to make an installer, I suggest you to just use Packpanda. I personally use the below method because in my opinion it's faster (with Packpanda, you can only generate an installer which you then need to manually make an archive from) and I've used it for my other non-Panda3d projects.

This tutorial is for Windows users, but I think all the tools I use here are available for Mac/Linux. Another reason you might find this method more convenient is that you don't need to use the console at all.

Tools you'll need:

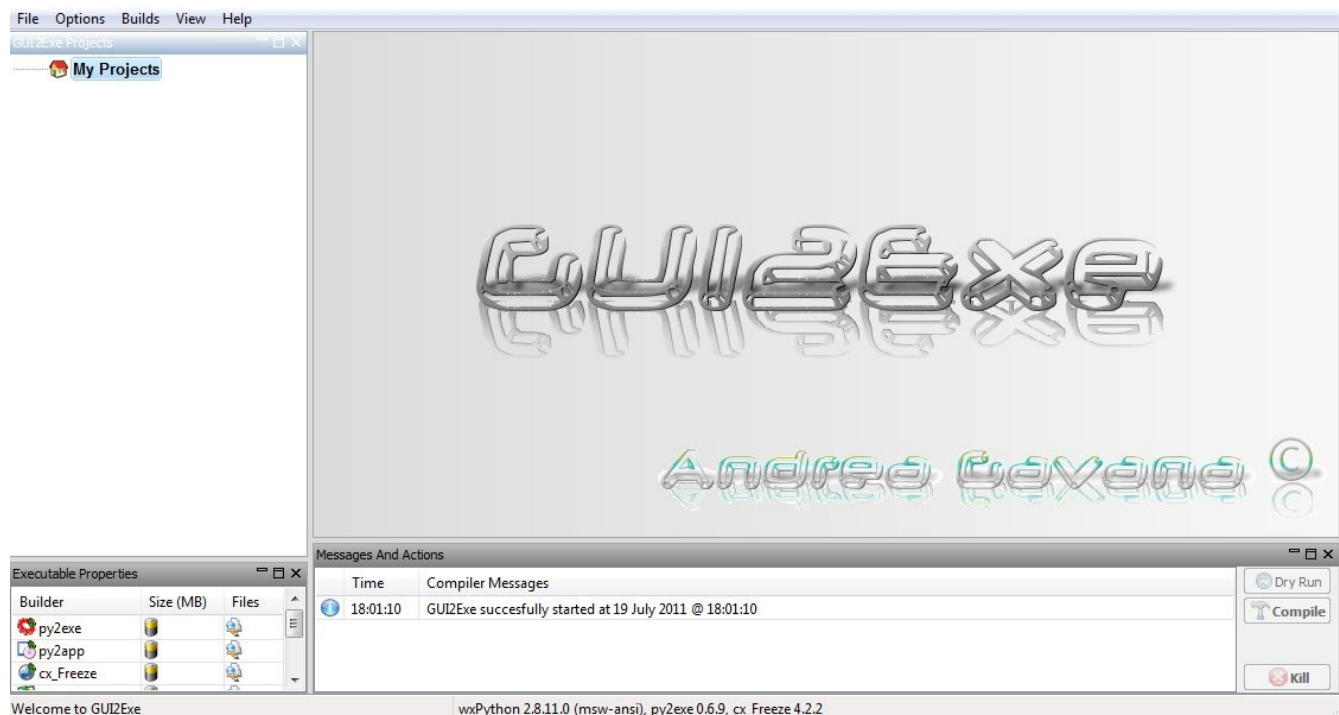
GUI2EXE: <http://code.google.com/p/gui2exe/>

cxFreeze: <http://cx-freeze.sourceforge.net/>

wxPython <http://www.wxpython.org/> (used by GUI2EXE, it's possible you already have it)

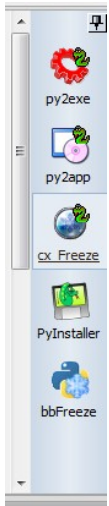
GUI2EXE is a Python script, there is no exe to run it, you need to run GUI2Exe.py to start it. I suggest to make a batch file and run that each time instead.

This is what you'll see when you start it:

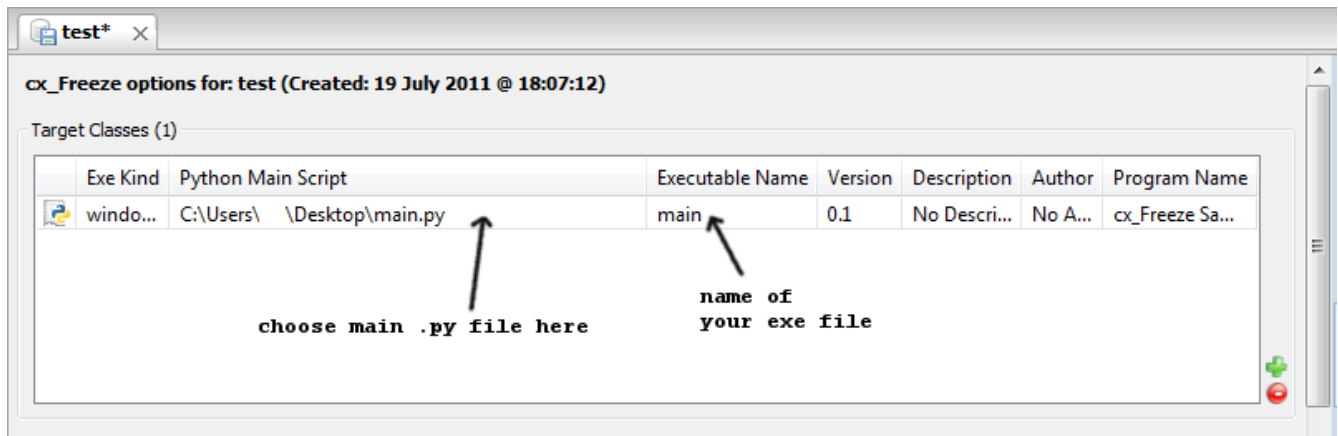


GUI2EXE is basically a GUI frontend for popular exe generation tools for Python, like py2exe and cx_Freeze. I've chosen to use it with cx_Freeze because it is the only one which seems to support all the platforms I need.

Click “File” → “New project”, then choose a name for your project. I'm going to name mine “test”. Now, on the right side on the window, click on cx_Freeze:

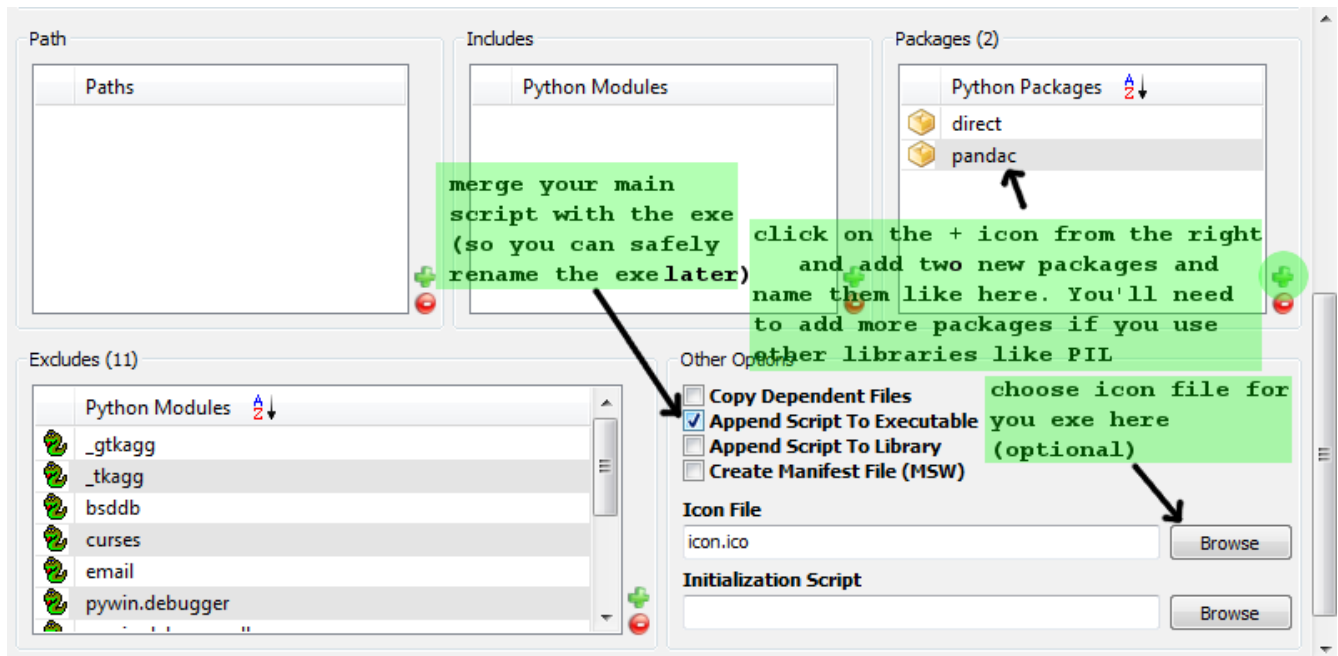


Click under “Python Main Script” and a file browser will pop up. This is where you'll set your main python file. Next to it is “Executable name” which is the name of the generated *.exe file.



Now scroll down:

Add new packages like in the images, click on “Append Script To Executable” and select your icon file you want to use, if you have one.



Finally, click on the Compile button in the bottom-right of the window:



The process will take a new seconds. A pop-up will appear, click "No".

A "dist" folder is created in the folder of your main python file. We need to do some manual work. As you have noticed no Panda dll was automatically copied. Go to your Panda SDK folder, then the "bin" folder, copy all the files except the .exe files to the dist folder. You'll also need to copy the "etc" folder from the SDK to your "dist" folder.

That's it. Rename the "dist" folder to whatever you want and put it in an archive.

Having all the dll files in the same folder as your exe isn't very nice, I know. Let me know of a way to have all the dlls in a subdirectory and I'll add the info here.