
Subject: Re: Support for plug-in architecture
Posted by [slashupp](#) on Sat, 21 Mar 2020 12:15:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi mirek

I attach my attempt at creating .so for plug-ins.

There are two: `plug_one` that uses 'standard' c++ class, and the second tries to create a `Upp-ctrl` as plug-in.

The first works fine after a bit of a fiddle, but the second gives invalid access/segfault for reasons I am yet to find.

I will appreciate if you or anybody checks over my code and hopefully spot the error.

To create the three packages I used two package dirs: 'temp' to hold the test-app and 'libs' to hold the plug-ins.

I then changed for the plugins the following:

In [Build/Output mode..] I set the 'Target file override' to "`~/plug_in_libs/libplug_one.so`" and "`~/plug_in_libs/libmyctrl.so`" after checking 'Release' for each plug-in.

The plugins must be compiled in Release mode.

Also for the plugins I changed each separately in [Setup/Build methods..] the 'Release options' by pre-pending `-fPIC` to it's content (`-fPIC -O3 -ffunction-sections -fdata-sections`), and 'Release link options' to (`-shared -Wl,--gc-sections,-soname,libmyctrl.so`) and (`-shared -Wl,--gc-sections,-soname,libplug_one.so`) for the two plug-ins

((POSSIBLE BUG: the [Setup/Build methods..] are not unique per open theide, a change in one gets reflected in others))

Compile the three packages now as you would do normally and the two .so's will be created in the `~/plug_in_libs` directory.

Change the constants in '`plug_one.cpp`' to reflect your path to the .so's.

File Attachments

1) [testplugins.zip](#), downloaded 242 times
