Subject: Compiling under Linux
Posted by cbpporter on Tue, 27 Nov 2012 10:09:32 GMT
View Forum Message <> Reply to Message

It has been years since I had to compile U++ for Linux, and even then I used the NOGTK flag. Today I tried to compile from source on a machine I have no root access, and I got this:

Quote:
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections -I./ -I/usr/include/gtk-2.0
-I/usr/lib64/gtk-2.0/include -I/usr/include/atk-1.0 -I/usr/include/cairo -I/usr/include/pango-1.0
-I/usr/include/glib-2.0 -I/usr/lib64/glib-2.0/include -I/usr/include/pixman-1 -I/usr/include/freetype2
-I/usr/include/libpng12   -I/usr/include/freetype2 -I/usr/include/gtk-2.0 -I/usr/include/glib-2.0
-I/usr/lib/glib-2.0/include -I/usr/lib/gtk-2.0/include -I/usr/include/cairo -I/usr/include/pango-1.0
-I/usr/include/atk-1.0 -I/usr/X11R6/include -I/usr/X11R6/include/freetype2
-I/usr/X11R6/include/gtk-2.0 -I/usr/X11R6/include/glib-2.0 -I/usr/X11R6/lib/glib-2.0/include
-I/usr/X11R6/lib/gtk-2.0/include -I/usr/X11R6/include/cairo -I/usr/X11R6/include/pango-1.0
-I/usr/X11R6/include/atk-1.0 -I/usr/include/gdk-pixbuf-2.0 -DflagGUI -DflagGCC -DflagSHARED
-DflagLINUX -DflagPOSIX CtrlLib/TrayIconX11.cpp -o
_out/CtrlLib//home/upp/Scripts/GCCMK.bm-Gcc-Gui-Linux-Posix- Shared/TrayIconX11.o
CtrlLib/TrayIconX11.cpp:9:34: error: libnotify/notify.h: No such file or directory

scope

It seems it can't find libnotify. Is this lib required? Can I remove the dependency with the NOGTK flag? And how do you specify that flag. I can no longer remember. I am digging thought the makefiles to see.

EDIT: Changed to quote, to allow line wrap. Honza

Subject: Re: Compiling under Linux
Posted by dolik.rce on Tue, 27 Nov 2012 12:54:01 GMT
View Forum Message <> Reply to Message

Hi cbporter,

Libnotify is used to show the notifications near panel. If I recall correctly, it is not used when compiling with NOGTK flag.

Since you write about Makefiles, I'll assume that you are using the Makefiles generated by theide. In that case, you can just add -DflagNOGTK to Macro variable (should be on line 3 or 4 in the Makefile) and it should work.

Best regards,
Honza

Subject: Re: Compiling under Linux
Posted by cbpporter on Tue, 27 Nov 2012 13:00:59 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Tue, 27 November 2012 14:54
Since you write about Makefiles, I'll assume that you are using the Makefiles generated by theide. In that case, you can just add -DflagNOGTK to Macro variable (should be on line 3 or 4 in the Makefile) and it should work.

No, I was talking about the makefile you get when downloading only U++ sources and doing a manual first build.

Subject: Re: Compiling under Linux
Posted by cbpporter on Tue, 27 Nov 2012 13:03:17 GMT
View Forum Message <> Reply to Message

But I did find something there. Let me see if it compiles .

Subject: Re: Compiling under Linux
Posted by dolik.rce on Tue, 27 Nov 2012 14:02:16 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Tue, 27 November 2012 14:00dolik.rce wrote on Tue, 27 November 2012 14:54
Since you write about Makefiles, I'll assume that you are using the Makefiles generated by theide. In that case, you can just add -DflagNOGTK to Macro variable (should be on line 3 or 4 in the Makefile) and it should work.

No, I was talking about the makefile you get when downloading only U++ sources and doing a manual first build.

Well, that one (in uppsrc/Makefile in the downloaded tarball I believe) is generated by theide too

Honza

Thanks! That kind of worked. Is there an U++ specific 64 bit flag or is a "-m64" option sufficient? I am getting:

/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libpthread.so when searching for -lpthread
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libpthread.a when searching for -lpthread
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libdl.so when searching for -ldl
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libdl.a when searching for -ldl
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/librt.so when searching for -lrt
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/librt.a when searching for -lrt
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libz.so when searching for -lz
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libz.a when searching for -lz
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libdl.so when searching for -ldl
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libdl.a when searching for -ldl
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libXext.so when searching for -lXext
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libX11.so when searching for -lX11
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libX11.so when searching for -lX11
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libdl.so when searching for -ldl
/usr/lib64/gcc/x86_64-suse-linux/4.3/../../../../x86_64-suse-linux/bin/ld: skipping incompatible /usr/lib/libdl.a when searching for -ldl

and I am guessing that it compiled using 32 bits on a 64 system.

Using -m64 I managed to get TheIDE to work, but when compiling the Bombs example and adding -m64 "Setup/build methods, I am still some errors.

Subject: Re: Compiling under Linux
Posted by dolik.rce on Tue, 27 Nov 2012 14:47:19 GMT
View Forum Message <> Reply to Message

Well, on my 64bit machine theide doesn't use -m64 and everything works well. It must be because you have both 32b and 64b libs available. Chceck if you build method is configured with the paths to the 64bit versions of the libraries (probably /usr/lib64). The default file that comes with the tarball is probably only good for pure 32b or pure 64bit systems...

Honza

Subject: Re: Compiling under Linux
Posted by cbpporter on Mon, 03 Dec 2012 07:13:48 GMT
View Forum Message <> Reply to Message

Thank you for the help!

Now it works perfectly!

It wasn't a very smooth process, but still U++ has come a long way when installing under Linux.