Subject: Linux U++ libraries
Posted by koldo on Thu, 26 Mar 2020 14:53:27 GMT

View Forum Message <> Reply to Message

Sorry for the question.

My Linux crashed and I have had to reinstall it.
Lots of problems installing U++ in Ubuntu because of folder and file permissions and untrusted application...
Finally all is running, but I have had to create the GCC.bm by hand, and I need these folders:

It is clear I am not an expert . How to get these folders?

File Attachments
1) Sin título.png , downloaded 923 times

```
Linking has failed
/usr/bin/ld: cannot find -lgtk-3
/usr/bin/ld: cannot find -lgdk-3
/usr/bin/ld: cannot find -latk-1.0
/usr/bin/ld: cannot find -lgdk_pixbuf-2.0
/usr/bin/ld: cannot find -lgdk_pixbuf-2.0
collect2: error: ld returned 1 exit status
```

Subject: Re: Linux U++ libraries
Posted by Oblivion on Thu, 26 Mar 2020 15:40:37 GMT

View Forum Message <> Reply to Message

Hello Koldo,

Mine, for GCC:

/usr/X11R6/lib
/usr/lib
/usr/lib64
/usr/local/lib
/opt/whatever/optional/library/you_use

For CLANG

((LIBRARIES))
/usr/X11R6/lib
/usr/lib

/usr/local/lib
/opt/whatever/optional/library/you_use

Almost all distros use these settings by default.

Best regards,
Oblivion

Subject: Re: Linux U++ libraries
Posted by koldo on Thu, 26 Mar 2020 16:02:52 GMT
View Forum Message <> Reply to Message

Thank you Oblivion, but in Ubuntu 18.04, those folders do not work

---

Subject: Re: Linux U++ libraries
Posted by Oblivion on Thu, 26 Mar 2020 17:37:43 GMT
View Forum Message <> Reply to Message

Hmm, that's bad.
I'm not on Ubuntu, but you can try a trick I use sometimes:
Try ldd to get the paths for TheIDE (ldd can show the required libraries linked against theIDE's executable):

#ldd /usr/bin/theide # if it is the right path, I mean.

You can check the paths.

The output should be something like this:

linux-vdso.so.1 (0x00007ffd5b64b000)
libc.so.6 => /usr/lib/libc.so.6 (0x00007f9c2c110000)
libglib-2.0.so.0 => /usr/lib/libglib-2.0.so.0 (0x00007f9c2bfe8000)
libgio-2.0.so.0 => /usr/lib/libgio-2.0.so.0 (0x00007f9c2be30000)
libgobject-2.0.so.0 => /usr/lib/libgobject-2.0.so.0 (0x00007f9c2bdd0000)
libgtk-3.so.0 => /usr/lib/libgtk-3.so.0 (0x00007f9c2b6f0000)
libgdk-3.so.0 => /usr/lib/libgdk-3.so.0 (0x00007f9c2b5f0000)
//....

Hopefully you can start from there.

Best regards,
Oblivion

Subject: Re: Linux U++ libraries
Posted by koldo on Thu, 26 Mar 2020 19:46:44 GMT
View Forum Message <> Reply to Message

Dear Oblivion

I have done that obtaining these:
/usr/lib/x86_64-linux-gnu
/lib/x86_64-linux-gnu
/lib64

But the link error remains...

Subject: Re: Linux U++ libraries
Posted by Oblivion on Thu, 26 Mar 2020 20:08:34 GMT
View Forum Message <> Reply to Message

hello Koldo,

Did you download the devel or dev packages?
AFAIK, they have to be installed separately.

If that does not do the trick, can you check the current library cache? Because it has to be
updated. (devel packages usually do that automatically for you.

First,

ldconfig -p

It should list the current library paths.

Still, if you don't see the paths you need, add the paths /usr/lib, /lib/, /usr/local/lib to /etc/ld.so.conf,
and then run ldconfig to update cache.

See: http://man7.org/linux/man-pages/man8/ldconfig.8.html

Hope this helps.

Best regards,
Oblivion

Subject: Re: Linux U++ libraries
Posted by koldo on Fri, 27 Mar 2020 11:32:00 GMT

Thank you Oblivion

I have tried with it, but there are no additional folders.
In the output of "make install" there is something that may be relevant:
Copying *.bm
cp: cannot stat '*.bm': No such file or directory

It seems that for any reason, "make" does not generate the CLANG.bm and GCC.bm files.

This is all the "make install" output:
./doinstall --verbose
DOINSTALL INFO: >> Starting installation in HOME directory (U++ style)
DOINSTALL INFO: Variable values:
DOINSTALL INFO: DESTDIR   =
DOINSTALL INFO: prefix   = /home/aupa
DOINSTALL INFO: Creating upp, upp.out, MyApps and .upp directories in /home/aupa/
DOINSTALL INFO: Copying theide and umk
DOINSTALL INFO: Creating theide.desktop, copying default icon and creating user menu entry
/usr/bin/xdg-desktop-menu
xdg-desktop-menu install --mode user --novendor theide.desktop
DOINSTALL INFO: Copying U++ source code
DOINSTALL INFO: Cleaning source code
DOINSTALL INFO: Copying *.bm
cp: cannot stat '*.bm': No such file or directory
DOINSTALL INFO: Copying cs-cz.udc de-de.udc en-gb.udc en-us.udc es.udc fi.udc fr-fr.udc
hu.udc it.udc nb.udc nl.udc ro.udc sk-sk.udc
DOINSTALL INFO: Generating default theide var files in /home/aupa/.upp/theide/
DOINSTALL INFO: << End of installation
(base) aupa@aupa:~/Desktop/upp-x11-src-14188$ sudo make install
[sudo] password for aupa:
./doinstall --verbose
DOINSTALL INFO: >> Starting installation in HOME directory (U++ style)
DOINSTALL INFO: Variable values:
DOINSTALL INFO: DESTDIR   =
DOINSTALL INFO: prefix   = /home/aupa
DOINSTALL INFO: Creating upp, upp.out, MyApps and .upp directories in /home/aupa/
DOINSTALL INFO: Copying theide and umk
DOINSTALL INFO: Creating theide.desktop, copying default icon and creating user menu entry

/usr/bin/xdg-desktop-menu
xdg-desktop-menu install --mode user --novendor theide.desktop
DOINSTALL INFO: Copying U++ source code
DOINSTALL INFO: Cleaning source code
DOINSTALL INFO: Copying *.bm
cp: cannot stat '*.bm': No such file or directory
DOINSTALL INFO: Copying cs-cz.udc de-de.udc en-gb.udc en-us.udc es.udc fi.udc fr-fr.udc
hu.udc it.udc nb.udc nl.udc ro.udc sk-sk.udc
DOINSTALL INFO: Generating default theide var files in /home/aupa/.upp/theide/
DOINSTALL INFO: << End of installation

Subject: Re: Linux U++ libraries
Posted by koldo on Fri, 27 Mar 2020 11:36:59 GMT
View Forum Message <> Reply to Message

This is the beginning of the "make" output.
Apparently the creation of CLANG.bm.in GCC.bm is OK...
./domake
DOMAKE INFO: Build ide = true
DOMAKE INFO: Build umk = true
/bin/uname
DOMAKE INFO: Detected system: Linux
DOMAKE INFO: Searching for g++ compiler
/usr/bin/g++
DOMAKE INFO: Found /usr/bin/g++
DOMAKE INFO: g++ version: 7
DOMAKE INFO: Searching for supported C++ mode
DOMAKE INFO: The -std=c++14 is supported by 'g++'
DOMAKE INFO: Searching for extra make parameters
DOMAKE INFO: Make parameters:
DOMAKE INFO: --> CXXFLAGS=-O3 -ffunction-sections -fdata-sections -std=c++14
DOMAKE INFO: Extra parameters will supersede default parameters
DOMAKE INFO: Configuring uppsrc/Makefile.in uppsrc/uMakefile.in CLANG.bm.in GCC.bm.in
with pkg-config
/usr/bin/pkg-config
DOMAKE INFO: pkg-config: Found library gtk+-3.0
DOMAKE INFO: pkg-config: Found library x11
DOMAKE INFO: pkg-config: Found library libnotify
DOMAKE INFO: pkg-config: Found library freetype2
DOMAKE INFO: Searching for gmake or make
/usr/bin/make
DOMAKE INFO: Found make. Compiling...

Subject: Re: Linux U++ libraries
Posted by Oblivion on Fri, 27 Mar 2020 13:52:29 GMT

Hello Koldo,

Quote:
Copying *.bm
cp: cannot stat '*.bm': No such file or directory

Yes, I can confirm this, now.

I downloaded the U++ nightly, and I've got the same error with GCC.

The reason is GNU cp seems to differentiate between "*.bm" and "*.bm.in".

I don't know why the build method files have the ".bm.in" extension though. (clash ?)

Renaming the files with ".bm.in" to ".bm" (or vice versa) resolves THAT error for me.

Best regards,
Oblivion

Subject: Re: Linux U++ libraries
Posted by koldo on Fri, 27 Mar 2020 14:12:40 GMT

Hi Olivion

Thank you for your effort.
Unfortunately .bm.in files seem to be just templates, where include and lib files are empty:
INCLUDE = "";
LIB = "";
In my previous output you can see this:DOMAKE INFO: Configuring uppsrc/Makefile.in
uppsrc/uMakefile.in CLANG.bm.in GCC.bm.in with pkg-config
/usr/bin/pkg-config
DOMAKE INFO: pkg-config: Found library gtk+-3.0
DOMAKE INFO: pkg-config: Found library x11
DOMAKE INFO: pkg-config: Found library libnotify
DOMAKE INFO: pkg-config: Found library freetype2Somebody seems to search for folders, but
.bm files are not created.

Edit: I have even changed permissions to HOME and sources folder, giving full permissions to
anybody, but nothing changes.

View Forum Message <> Reply to Message

Hi,

I did fresh install on lubuntu 16.04 and upp nightly 2 days ago

I followed these steps

apt install libgtk-3-dev
sudo bash buildrequires.debian
./domake
./doinstall

I the copied GCC.bom.in to ~/.upp/theide folder. renamed to GCC.bom

Every thing seems to work ok.

My BOM file. Default file without modification.

BUILDER = "GCC";
COMMON_CPP_OPTIONS = "-std=c++14";
DEBUG_INFO = "2";
DEBUG_BLITZ = "1";
DEBUG_LINKMODE = "1";
DEBUG_OPTIONS = "-O0";
DEBUG_FLAGS = "";
RELEASE_BLITZ = "0";
RELEASE_LINKMODE = "1";
RELEASE_OPTIONS = "-O3 -ffunction-sections -fdata-sections";
RELEASE_SIZE_OPTIONS = "-Os -finline-limit=20 -ffunction-sections -fdata-sections";
RELEASE_FLAGS = "";
RELEASE_LINK = "-WI,--gc-sections";
DEBUGGER = "gdb";
PATH = "";
INCLUDE = ""
LIB = "";
REMOTE_HOST = "";
REMOTE_OS = "";
REMOTE_TRANSFER = "";
REMOTE_MAP = "";
LINKMODE_LOCK = "0";

Subject: Re: Linux U++ libraries
Posted by koldo on Sat, 28 Mar 2020 10:39:07 GMT

Thank you Deepak

Do you mean that it is not necessary to fill the include/lib paths in .BM file?
Anyway, if the .BM files have to be set by hand, it seems there is something broken in Linux installer.
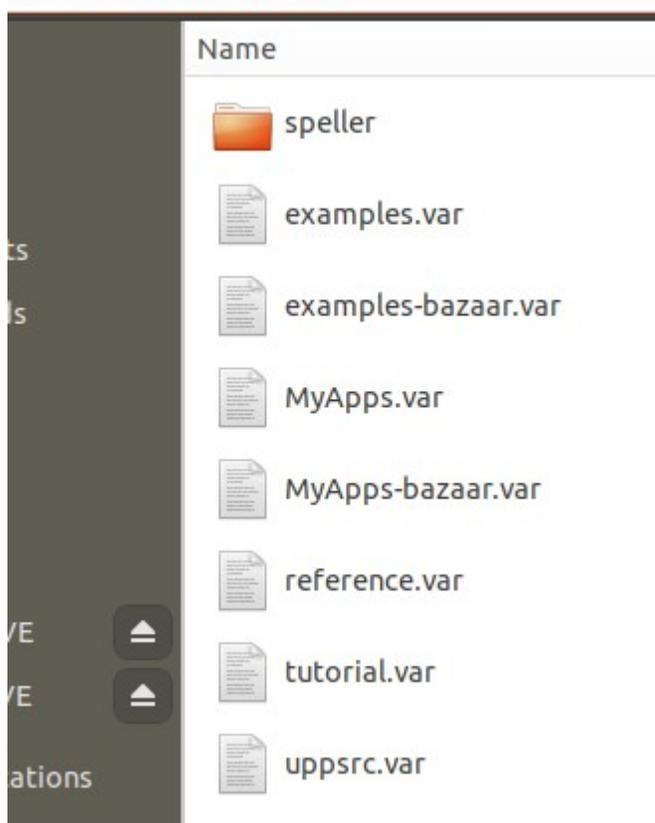
Edit: This is the .upp/theide folder just after install. No .BM file

File Attachments
1) Anotación 2020-03-28 114736.jpg , downloaded 855 times

/home/aupa/.upp/theide

Name

speller

examples.var

examples-bazaar.var

MyApps.var

MyApps-bazaar.var

reference.var

tutorial.var

uppsrc.var

Subject: Re: Linux U++ libraries
Posted by koldo on Sat, 28 Mar 2020 11:12:45 GMT

Well, GCC finally works. The trick has been:
- Delete all U++ install
- sudo make + sudo make install
- Copy and rename .bm.in files in sources folder to .bm in /home/.../.upp/theide

- Set full permissions to all upp folders under HOME

Using CLANG, the message "There were errors" appears at the beginning, but even with "Verbose" on, no additional message is shown.

Please, it would be great if the person who maintains the Linux installer could take a look at this.

Subject: Re: Linux U++ libraries
Posted by amrein on Sat, 28 Mar 2020 12:14:39 GMT
View Forum Message <> Reply to Message

Hi

Someone disabled *.bm files generation. I will fix this.

edit: done !

Subject: Re: Linux U++ libraries
Posted by mirek on Sat, 28 Mar 2020 12:26:28 GMT
View Forum Message <> Reply to Message

koldo wrote on Sat, 28 March 2020 12:12Well, GCC finally works. The trick has been:
- Delete all U++ install
- sudo make + sudo make install
- Copy and rename .bm.in files in sources folder to .bm in /home/.../.upp/theide
- Set full permissions to all upp folders under HOME

Using CLANG, the message "There were errors" appears at the beginning, but even with "Verbose" on, no additional message is shown.

Please, it would be great if the person who maintains the Linux installer could take a look at this.

I think the best is to avoid .bm completely and just setup these in theide when there is no .bm available.

Subject: Re: Linux U++ libraries
Posted by koldo on Sat, 28 Mar 2020 14:05:36 GMT
View Forum Message <> Reply to Message

Quote:I think the best is to avoid .bm completely and just setup these in theide when there is no .bm available.In this case, as in Windows, it could be an "Automatic build methods setup.." option, although I think TheIDE should have to be fully ready after make install.

Other problem I have found is that without setting full permissions to all upp folders under HOME,

the compiler complains because of lack of permissions. In summary:
- make does not work
- sudo make worka, but if TheIDE is not run with sudo, compiler complains because of lack of permissions

---

Subject: Re: Linux U++ libraries
Posted by deep on Sat, 28 Mar 2020 15:43:55 GMT
View Forum Message <> Reply to Message

If you do make and install steps as normal user then it works fine.

If do with sudo or root then you need to fix permissions.

Subject: Re: Linux U++ libraries
Posted by mirek on Sat, 28 Mar 2020 16:19:50 GMT
View Forum Message <> Reply to Message

koldo wrote on Sat, 28 March 2020 15:05Quote:I think the best is to avoid .bm completely and just setup these in theide when there is no .bm available.In this case, as in Windows, it could be an "Automatic build methods setup.." option, although I think TheIDE should have to be fully ready after make install.

Actually, this already works... And there is no difference really, if there are no .bm, ide makes them on start. You would not notice that happened.

Subject: [SOLVED] Re: Linux U++ libraries
Posted by koldo on Sat, 28 Mar 2020 17:13:32 GMT
View Forum Message <> Reply to Message

When theide started, Build methods was empty.
However, I have reviewed the commands used, and I have realised that the first time I did a "sudo make", so because of that I ruined the normal install.

Subject: Re: Linux U++ libraries
Posted by amrein on Sat, 28 Mar 2020 20:10:30 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 28 March 2020 17:19koldo wrote on Sat, 28 March 2020 15:05Quote:I think

---

the best is to avoid .bm completely and just setup these in theide when there is no .bm available.In this case, as in Windows, it could be an "Automatic build methods setup.." option, although I think TheIDE should have to be fully ready after make install.

Actually, this already works... And there is no difference really, if there are no .bm, ide makes them on start. You would not notice that happened.

The .bm files created with "domake" are mainly for umk command line users.
The script copy raw GCC.bm.in and CLANG.bm.in as GCC.bm and CLANG.bm.

Subject: Re: Linux U++ libraries
Posted by Lance on Sun, 29 Mar 2020 01:14:01 GMT
View Forum Message <> Reply to Message

I have exact the same experience as koldo. No automatic build method generation happened on Ubuntu 18.04 as of the most recent nightly build. I mannully created a gcc.bm using the content posted by deep and it fixed my problem.

File Attachments
1) Screenshot from 2020-03-28 21-04-37.png, downloaded 733 times


```
lance@basementtop:~/Downloads/upp-x11-src-14195$ make install
./doinstall --verbose
DOINSTALL INFO: >> Starting installation in HOME directory (U++ style)
DOINSTALL INFO: Variable values:
DOINSTALL INFO: DESTDIR        =
DOINSTALL INFO: prefix         = /home/lance
DOINSTALL INFO: Creating upp, upp.out, MyApps and .upp directories in /home/lance/
DOINSTALL INFO: Copying theide and umk
DOINSTALL INFO: Creating theide.desktop, copying default icon and creating user menu
/usr/bin/xdg-desktop-menu
xdg-desktop-menu install --mode user --novendor theide.desktop
DOINSTALL INFO: Copying U++ source code
DOINSTALL INFO: Cleaning source code
DOINSTALL INFO: Copying *.bm
cp: cannot stat '*.bm': No such file or directory
DOINSTALL INFO: Copying cs-cz.udc de-de.udc en-gb.udc en-us.udc es.udc fi.udc fr-fr.
c
DOINSTALL INFO: Generating default theide var files in /home/lance/.upp/theide/
DOINSTALL INFO: << End of installation
```

Subject: Re: Linux U++ libraries
Posted by mirek on Sun, 29 Mar 2020 13:55:55 GMT

Lance wrote on Sun, 29 March 2020 03:14I have exact the same experience as koldo. No automatic build method generation happened on Ubuntu 18.04 as of the most recent nightly build. I mannully created a gcc.bm using the content posted by deep and it fixed my problem.

That is as designed. They get created the first time theide runs.

Of course, install should not try to copy them, that is a mistake.

Subject: Re: Linux U++ libraries
Posted by mirek on Sun, 29 Mar 2020 14:31:09 GMT

OK, I can see the current experience is underwhelming. I am working on "linux experience overhaul" now...

One little idea that might (or might not) help us:

 https://www.ultimatepp.org/forums/index.php?t=msg&th=109 65&start=0&

My current plan is to

a) fix makefile based tarball

b) create a second more 'automated' U++ release that would work as close to Win32 or MacOS as possible. Single directory, static umk, script that builds theide. Maybe even could contain theide script file that calls that script to get replace by the binary. No more install, just keep files in that folder, just like with macos and win32 (except I am undecided whether to put config files there, or use ~/.upp). Building with umk on decent machine in Release/BLITZ takes 2 minutes, that is faster than installing most applications in windows...

Of course, install script should resolve dependencies as well...

Mirek

Subject: Re: Linux U++ libraries
Posted by koldo on Sun, 29 Mar 2020 15:14:39 GMT

Thank you, Mirek. It's great that you can do this. Continuing with this line, it would be excellent that we have a clear line about which is the best way to distribute open source applications, like in

GitHub, in which the ease of use for the user is maximum (maybe the user knows nothing at all about C++), in order to:

- Distribute binaries
- The user can compile the binaries in the simplest way on both Windows and Linux or Mac.

Subject: Re: Linux U++ libraries
Posted by mirek on Sun, 29 Mar 2020 17:35:51 GMT

View Forum Message <> Reply to Message

koldo wrote on Sun, 29 March 2020 17:14Thank you, Mirek. It's great that you can do this. Continuing with this line, it would be excellent that we have a clear line about which is the best way to distribute open source applications, like in GitHub, in which the ease of use for the user is maximum (maybe the user knows nothing at all about C++), in order to:

- Distribute binaries
- The user can compile the binaries in the simplest way on both Windows and Linux or Mac.

Actually, if we wanted to be crazy, we can upload static umk to source tree with install script... But thats maybe too crazy.

But what should be possible is some form of single installation file that would work on given CPU architecture. It would have to run in console, but other than that... Combined with BLITZ, this sounds like a nice idea....

Mirek

Subject: Re: Linux U++ libraries
Posted by mirek on Sun, 29 Mar 2020 17:38:55 GMT

View Forum Message <> Reply to Message

amrein wrote on Sat, 28 March 2020 21:10mirek wrote on Sat, 28 March 2020 17:19koldo wrote on Sat, 28 March 2020 15:05Quote:I think the best is to avoid .bm completely and just setup these in theide when there is no .bm available.In this case, as in Windows, it could be an "Automatic build methods setup.." option, although I think TheIDE should have to be fully ready after make install.

Actually, this already works... And there is no difference really, if there are no .bm, ide makes them on start. You would not notice that happened.

The .bm files created with "domake" are mainly for umk command line users.
The script copy raw GCC.bm.in and CLANG.bm.in as GCC.bm and CLANG.bm.

Good point.

That said, I think we do not really need .in files now. Empty (mostly) GCC.bm and CLANG.bm should work just fine..

Subject: Re: Linux U++ libraries
Posted by mirek on Sun, 29 Mar 2020 18:43:52 GMT
View Forum Message <> Reply to Message

amrein wrote on Sat, 28 March 2020 21:10mirek wrote on Sat, 28 March 2020 17:19koldo wrote on Sat, 28 March 2020 15:05Quote:I think the best is to avoid .bm completely and just setup these in theide when there is no .bm available.In this case, as in Windows, it could be an "Automatic build methods setup.." option, although I think TheIDE should have to be fully ready after make install.

Actually, this already works... And there is no difference really, if there are no .bm, ide makes them on start. You would not notice that happened.

The .bm files created with "domake" are mainly for umk command line users.
The script copy raw GCC.bm.in and CLANG.bm.in as GCC.bm and CLANG.bm.

I have just changed umk to create default build methods just like theide. No more need to create these during install. .var should be autoinstalled in theide as well.

Mirek

Subject: Re: Linux U++ libraries
Posted by Lance on Sun, 29 Mar 2020 18:49:41 GMT
View Forum Message <> Reply to Message

Quote:That is as designed. They get created the first time theide runs.

Except the build method is not created upon the first run. Something is missing. I have deleted the .upp folder before make install and have to fight with an ide not configured with any build method. If build method generation happens at all, like it does in windows, nobody will notice anything and make complaints.

Subject: Re: Linux U++ libraries
Posted by amrein on Mon, 30 Mar 2020 10:55:49 GMT

Snapshots are created each morning. The last snapshot "upp-x11-src-14201.tar.gz" works perfectly.

Subject: Re: Linux U++ libraries
Posted by amrein on Mon, 30 Mar 2020 10:59:52 GMT

mirek wrote on Sun, 29 March 2020 20:43amrein wrote on Sat, 28 March 2020 21:10mirek wrote on Sat, 28 March 2020 17:19koldo wrote on Sat, 28 March 2020 15:05Quote:I think the best is to avoid .bm completely and just setup these in theide when there is no .bm available.In this case, as in Windows, it could be an "Automatic build methods setup.." option, although I think TheIDE should have to be fully ready after make install.

Actually, this already works... And there is no difference really, if there are no .bm, ide makes them on start. You would not notice that happened.

The .bm files created with "domake" are mainly for umk command line users.
The script copy raw GCC.bm.in and CLANG.bm.in as GCC.bm and CLANG.bm.

I have just changed umk to create default build methods just like theide. No more need to create these during install. .var should be autoinstalled in theide as well.

Mirek

Ok so creating and copying .bm files are not needed anymore then?
There are used in several build scripts.

Subject: Re: Linux U++ libraries
Posted by amrein on Mon, 30 Mar 2020 11:11:00 GMT

I removed .bm files from nightly tarball generation.

No more bm file handling in tarball build scripts too.