Subject: New experimental Linux/FreeBSD release
Posted by mirek on Thu, 02 Apr 2020 16:42:01 GMT
View Forum Message <> Reply to Message

Trying to make Linux experience nicer:

http://ultimatepp.org/downloads/upp-posix-14219.tar.xz

Unpack, cd to folder and run ./install (or if that does not work, sh install, but in that case let me know). If you are lucky, you will be in theide in about 2 minutes...

This is work in progress and at the moment will probably probably run flawlessly only on debian systems, although there is a try at fedora and openSUSE and related distros. I hope community will help to alter ./install to support more systems (just check the file, it is really easy to understand).

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Thu, 02 Apr 2020 19:06:15 GMT
View Forum Message <> Reply to Message

Hi Mirek

This king of tarball, using a script to auto-install dependencies and containing static binaries... it's so Windows like experience.
Most Linux developers will hate this.

Several issue for me:

* The extracted directory should have the same name as the tarball (if possible) and the directory name should contain the version number too

upp-posix-14207.tar.xz => upp-posix-14207

* No binary inside a source tarball, ever.

* No auto-install of dependencies.

Subject: Re: New experimental Linux/FreeBSD release
Posted by dolik.rce on Thu, 02 Apr 2020 19:50:18 GMT
View Forum Message <> Reply to Message

Hi Mirek,

I Agree with everything Amrein said... Also never use sudo in a script like this You should at least ask user if it is ok to run the packaging command, before showing password prompt In ideal world, you wouldn't even call the command if all the dependencies are already present.

That aside, even after ironing this out, the script will fail, unless you already have used U++ in past. Or if you don't have any build method called CLANG, as in my case. Some minimal build method should be inside the tarball, or should be generated by the script. It could use pkgconfig to fill in the include and library directories. Compiler can be set to c++, which is usually present regardless whether the system uses gcc or clang.

Best regards,
Honza

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Thu, 02 Apr 2020 20:46:57 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Thu, 02 April 2020 21:50
That aside, even after ironing this out, the script will fail, unless you already have used U++ in past. Or if you don't have any build method called CLANG, as in my case. Some minimal build method should be inside the tarball, or should be generated by the script. It could use pkgconfig to fill in the include and library directories. Compiler can be set to c++, which is usually present regardless whether the system uses gcc or clang.

CLANG should be autocreated by umk (umks32). Have you really tried to run it so that autocreation failed or you do infer this from the script?

(We are now actually building with empty include/library dirs, thats recent addition (U++ using pkg-config in builder)).

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Thu, 02 Apr 2020 20:47:47 GMT
View Forum Message <> Reply to Message

amrein wrote on Thu, 02 April 2020 21:06

* No binary inside a source tarball, ever.

* No auto-install of dependencies.

Do not think about this as source tarball.

EDIT: Or actually do, but then ignore ./install. You can ignore that, do dependecies yourself and just do 'make'. That should work. In other words, delete ./install, ./clean, ./umks32 and you have source tarball. Which will take much more time and effort to get working...

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by dolik.rce on Fri, 03 Apr 2020 04:37:32 GMT
View Forum Message <> Reply to Message

mirek wrote on Thu, 02 April 2020 22:46CLANG should be autocreated by umk (umks32). Have you really tried to run it so that autocreation failed or you do infer this from the script?

(We are now actually building with empty include/library dirs, thats recent addition (U++ using pkg-config in builder)).

Mirek

I have tried to run it. Here is the relevant part of the output (called with bash -x):
+ ./umks32
Usage: umk assembly main_package build_method -options [+flags] [output]

Examples: umk examples Bombs GCC -ab +GUI,SHARED ~/bombs
umk examples,uppsrc Bombs ~/GCC.bm -rv +GUI,SHARED ~/bin

See http://www.ultimatepp.org/appideumk$en-us.html for details
+ echo Using umks32 to build
Using umks32 to build
+ UMK=./umks32
+ ./umks32 ./uppsrc ide CLANG -brs ./theide
Invalid build method
+ ./umks32 ./uppsrc umk CLANG -brs ./umk
Invalid build method

Oh and by the way, you should definitely add "set -e" at the beginning of the script, so it fails on errors. This way, it tries to launch TheIDE which it failed to build

---

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 03 Apr 2020 05:35:39 GMT
View Forum Message <> Reply to Message

Here is what could be a well formed archive:

```
upp-posix-14205
|-cache
|    |-upp
|-config
|    |-upp
|       |-umk
|       |-theide
|           |-speller
|-extra
|    |-bazaar
|    |-examples
|    |-reference
|    |-tutorial
|-uppsrc
```

And how Makefile could be used:

```
make theide                          : build umk then use it to build theide with default parameters
make theide OPTIONS=-rbsv            : build umk then use it to build theide as release binary
make theide BUILD_METHOD=CLANG FLAGS=GUI,MT : build umk then use it to build theide
using CLANG and specific FLAGS
make umk                             : build umk only
make umk MAKE=gmake CXX=g++          : build umk using gmake and g++ (see
uppsrc/uMakefile.in for other possible options)
make clean                           : delete build directory and binaries
make build ASSEMBLY=examples PACKAGE=Button : build examples Button package from
extra directory
```

Please, test this tarball on Linux. It's well formed and Makefile is really faster:

http://dl.free.fr/mP39SKqYs

For Windows, the archive name could be upp-win64-14205.

For sandboxing umk and theide, it would be great to use $PWD/cache/upp and $PWD/config/upp/ and not use hidden directories
(and use ~/.config/upp and ~/.cache/upp only if umk and theide are in /opt or /usr)

And... another suggestion: add the option to clean the cache directory from command line and from inside theide.

Subject: Re: New experimental Linux/FreeBSD release

Posted by mirek on Fri, 03 Apr 2020 07:31:51 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Fri, 03 April 2020 06:37mirek wrote on Thu, 02 April 2020 22:46CLANG should be autocreated by umk (umks32). Have you really tried to run it so that autocreation failed or you do infer this from the script?

(We are now actually building with empty include/library dirs, thats recent addition (U++ using pkg-config in builder)).

Mirek

I have tried to run it. Here is the relevant part of the output (called with bash -x):
+ ./umks32
Usage: umk assembly main_package build_method -options [+flags] [output]

Examples: umk examples Bombs GCC -ab +GUI,SHARED ~/bombs
umk examples,uppsrc Bombs ~/GCC.bm -rv +GUI,SHARED ~/bin

See http://www.ultimatepp.org/appideumk$en-us.html for details
+ echo Using umks32 to build
Using umks32 to build
+ UMK=./umks32
+ ./umks32 ./uppsrc ide CLANG -brs ./theide
Invalid build method
+ ./umks32 ./uppsrc umk CLANG -brs ./umk
Invalid build method

Oh and by the way, you should definitely add "set -e" at the beginning of the script, so it fails on errors. This way, it tries to launch TheIDE which it failed to build

Thats really interesting. With freshly installed fedora, after fixing one wrong package name (gtk3-devel), all went as expected. It actually took longer to install dependecies than to compile theide...

Are you running as regular user? Do you have .config folder inside the unpacked archive? What is inside? There should be umks32 folder and inside autogenerated CLANG.bm (it is after I do this on ubuntu or fedora).

(Of course this is all work of progress, I will try to add most of suggestion today, while trying to make this work on Fedora, Suse and Arch).

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 07:41:44 GMT
View Forum Message <> Reply to Message

amrein wrote on Fri, 03 April 2020 07:35Here is what could be a well formed archive:

So if I play clueless user, this is what I get:

Quote:
[cxl@localhost upp-posix-14205]$ make theide
make[1]: Entering directory '/home/cxl/upp-posix-14205/uppsrc'
mkdir -p /home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main -Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Builders/GCCMK-Gcc-L inux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Core/GCCMK-Gcc-Linux -Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Android/GCCMK-Gcc-Li nux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Java/GCCMK-Gcc-Linux -Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/Esc/GCCMK-Gcc-Linux-Posi x-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/bz2/GCCMK-Gcc-Lin ux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/lzma/GCCMK-Gcc-Li nux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/lz4/GCCMK-Gcc-Lin ux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/zstd/GCCMK-Gcc-Li nux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/Core/GCCMK-Gcc-Linux-Pos ix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/pcre/GCCMK-Gcc-Li nux-Posix-Shared/
(date '+#define bmYEAR   %y%n#define bmMONTH   %-m%n#define bmDAY   %-d%n#define bmHOUR   %-H%n#define bmMINUTE %-M%n#define bmSECOND %-S%n#define bmTIME Time(%y, %-m, %-d, %-H, %-M, %-S)' && \
echo '#define bmMACHINE "'`hostname`'" && \
echo '#define bmUSER   "'`whoami`'") > build_info.h
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections -std=c++14 -I./ -DflagGCC -DflagSHARED -DflagPOSIX -DflagLINUX -DflagMAIN umk/MacOs.cpp -o /home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main -Posix-Shared/MacOs.o
make[1]: c++: Command not found
make[1]: *** [uMakefile.in:216: /home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main -Posix-Shared/MacOs.o] Error 127
make[1]: *** Waiting for unfinished jobs....
echo '#define bmSVN_REVISION "14205"' >> build_info.h
echo '#define bmSVN_URL "svn://www.ultimatepp.org/upp/trunk/uppsrc/umk"' >> build_info.h
make[1]: Leaving directory '/home/cxl/upp-posix-14205/uppsrc'
make: *** [Makefile:51: umk] Error 2

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 03 Apr 2020 08:13:04 GMT
View Forum Message <> Reply to Message

Quote:So if I play clueless user, this is what I get:
...
make[1]: c++: Command not found
...

Than you install a c++ compiler or read the readme.md file and install all required dependencies. Most developers do this.

Installing gcc-c++ is enough to build umk. Then umk should tell what dependencies are missing (pkg-config and others).

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 08:27:44 GMT
View Forum Message <> Reply to Message

amrein wrote on Fri, 03 April 2020 10:13Quote:So if I play clueless user, this is what I get:
...
make[1]: c++: Command not found
...

Than you install a c++ compiler or read the readme.md file and install all required dependencies. Most developers do this.

Installing gcc-c++ is enough to build umk. Then umk should tell what dependencies are missing (pkg-config and others).

Small issues to (eventually) fix:

buildrequires.fedora - not sure whether this is supposed to be used, but it is wrong. libgtk3.0-devel is actually gtk3-devel. Also adding c++ there would not hurt.

at the end of process, theide size 60MB compared to 13MB in case of ./install

OK, I have spoiled things for me by fixing and running buildrequires.fedora. Going to create a new clone and test with installing just c++.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 08:43:03 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 03 April 2020 10:27
OK, I have spoiled things for me by fixing and running buildrequires.fedora. Going to create a new clone and test with installing just c++.

So this is how it looks on freshly installed fedora, after installing just c++:

```
make theide
bash: make: command not found...
Install package 'make' to provide command 'make'? [N/y] y


 * Waiting in queue...
The following packages have to be installed:
 gc-7.6.4-6.fc31.x86_64 A garbage collector for C and C++
 guile22-2.2.6-2.fc31.x86_64 A GNU implementation of Scheme for application extensibility
 libatomic_ops-7.6.10-2.fc31.x86_64 Atomic memory update operations
 make-1:4.2.1-15.fc31.x86_64 A GNU tool which simplifies the build process for users
Proceed with changes? [N/y] y


 * Waiting in queue...
 * Waiting for authentication...
 * Waiting in queue...
 * Downloading packages...
 * Requesting data...
 * Testing changes...
 * Installing packages...
make[1]: Entering directory '/home/cxl/upp-posix-14205/uppsrc'
mkdir -p /home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Builders/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Core/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Android/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/ide/Java/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/Esc/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/bz2/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/lzma/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/lz4/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/zstd/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/Core/GCCMK-Gcc-Linux-Posix-Shared/
mkdir -p /home/cxl/upp-posix-14205/cache/upp/plugin/pcre/GCCMK-Gcc-Linux-Posix-Shared/
(date '+#define bmYEAR    %y%n#define bmMONTH   %-m%n#define bmDAY     %-d%n#define
bmHOUR    %-H%n#define bmMINUTE  %-M%n#define bmSECOND  %-S%n#define bmTIME
Time(%y, %-m, %-d, %-H, %-M, %-S)' && \
echo '#define bmMACHINE "'`hostname`'" && \
echo '#define bmUSER    "'`whoami`'") > build_info.h
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX -DflagMAIN  umk/MacOs.cpp -o
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/MacOs.o
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
```

```
-DflagPOSIX -DflagLINUX -DflagMAIN  umk/Console.cpp -o
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/Console.o
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX -DflagMAIN  umk/IdeContext.cpp -o
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/IdeContext.o
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX -DflagMAIN  umk/MakeBuild.cpp -o
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/MakeBuild.o
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX -DflagMAIN  umk/Export.cpp -o
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/Export.o
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX -DflagMAIN  umk/umake.cpp -o
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/umake.o
echo '#define bmSVN_REVISION "14205"' >> build_info.h
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX  ide/Builders/CppBuilder.cpp -o
/home/cxl/upp-posix-14205/cache/upp/ide/Builders/GCCMK-Gcc-Linux-Posix-Shared/CppBuilder.
o
echo '#define bmSVN_URL "svn://www.ultimatepp.org/upp/trunk/uppsrc/umk"' >> build_info.h
c++ -c -x c++ -O3 -ffunction-sections -fdata-sections  -std=c++14 -I./ -DflagGCC -DflagSHARED
-DflagPOSIX -DflagLINUX  ide/Builders/MakeFile.cpp -o
/home/cxl/upp-posix-14205/cache/upp/ide/Builders/GCCMK-Gcc-Linux-Posix-Shared/MakeFile.o
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ide/Builders/Builders.h:4,
                 from ide/Builders/MakeFile.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
compilation terminated.
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ./ide/Builders/Builders.h:4,
                 from umk/umake.h:4,
                 from umk/umake.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ide/Builders/Builders.h:4,
                 from ide/Builders/CppBuilder.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
compilation terminated.
In file included from ./Esc/Esc.h:4,
```

```
                 from ./ide/Core/Core.h:4,
                 from ./ide/Builders/Builders.h:4,
                 from umk/umake.h:4,
                 from umk/MacOs.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
compilation terminated.
compilation terminated.
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ./ide/Builders/Builders.h:4,
                 from umk/umake.h:4,
                 from umk/IdeContext.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
compilation terminated.
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ./ide/Builders/Builders.h:4,
                 from umk/umake.h:4,
                 from umk/Console.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
compilation terminated.
make[1]: *** [uMakefile.in:896:
/home/cxl/upp-posix-14205/cache/upp/ide/Builders/GCCMK-Gcc-Linux-Posix-Shared/MakeFile.o]
Error 1
make[1]: *** Waiting for unfinished jobs....
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ./ide/Builders/Builders.h:4,
                 from umk/umake.h:4,
                 from umk/Export.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
compilation terminated.
In file included from ./Esc/Esc.h:4,
                 from ./ide/Core/Core.h:4,
                 from ./ide/Builders/Builders.h:4,
                 from umk/umake.h:4,
                 from umk/MakeBuild.cpp:1:
./Core/Core.h:232:10: fatal error: zlib.h: No such file or directory
  232 | #include <zlib.h>
      |          ^~~~~~~~
```

make[1]: *** [uMakefile.in:800:
/home/cxl/upp-posix-14205/cache/upp/ide/Builders/GCCMK-Gcc-Linux-Posix-Shared/CppBuilder.o] Error 1
compilation terminated.
make[1]: *** [uMakefile.in:410:
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/IdeContext.o] Error 1
make[1]: *** [uMakefile.in:216:
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/MacOs.o] Error 1
make[1]: *** [uMakefile.in:313:
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/Console.o] Error 1
make[1]: *** [uMakefile.in:701:
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/umake.o] Error 1
make[1]: *** [uMakefile.in:604:
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/Export.o] Error 1
make[1]: *** [uMakefile.in:507:
/home/cxl/upp-posix-14205/cache/upp/umk/GCCMK-Gcc-Linux-Main-Posix-Shared/MakeBuild.o] Error 1
make[1]: Leaving directory '/home/cxl/upp-posix-14205/uppsrc'
make: *** [Makefile:51: umk] Error 2

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 08:46:34 GMT
View Forum Message <> Reply to Message

Now truth the be told, zlib issue is not really the fault of tarball, I should probably use pkg-config for zlib too, which I do not at the moment.

However, it all just demonstrates how hard things can be for clueless users... and how hard is it to create tarball that is actually easy to use.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 08:50:24 GMT
View Forum Message <> Reply to Message

amrein wrote on Fri, 03 April 2020 07:35

And... another suggestion: add the option to clean the cache directory from command line and from inside theide.

My archive has ./cleanup script.

theide should not be able to clean the whole .cache (what would other applications say to that? . However it has "Clear UPPOUT" command (to clear all cached object files and binaries) for like 15 years.

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 09:56:44 GMT
View Forum Message <> Reply to Message

amrein wrote on Fri, 03 April 2020 10:13

Installing gcc-c++ is enough to build umk. Then umk should tell what dependencies are missing (pkg-config and others).

So I wondered how you mean this. I hoped that your tarball takes care about it, but now I see that you expect to add functionality to 'umk' that stops on failed pkg-config?

But even so, that would only list the names of missing libraries. The names of packages are different...

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 03 Apr 2020 13:25:39 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 03 April 2020 11:56amrein wrote on Fri, 03 April 2020 10:13

Installing gcc-c++ is enough to build umk. Then umk should tell what dependencies are missing (pkg-config and others).

So I wondered how you mean this. I hoped that your tarball takes care about it, but now I see that you expect to add functionality to 'umk' that stops on failed pkg-config?

But even so, that would only list the names of missing libraries. The names of packages are different...

Mirek

Yes the name of package are different and distribution dependent.
It means that if you want to simplify the use of U++ tarball, you will have to know each package name for each distributions.

Can you try again to play as clueless user?
Here are two files. One is an up to date Makefile. The second one is a list of dependencies for a few Linux distribution used by this Makefile.

What do you think?

edit: Still, umk and theide should be able to tell when there are missing dependencies. They fail without explicit reasons. Same thing for current uppsrc/Makefile.in and uppsrc/uMakefile.in.

File Attachments

1) MakefileAndBuildRequires.tar.gz, downloaded 349 times

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 14:59:45 GMT
View Forum Message <> Reply to Message

[quote title=amrein wrote on Fri, 03 April 2020 15:25]mirek wrote on Fri, 03 April 2020 11:56
What do you think?

Well, at that point, w.r.t. dependencies, this is just overengineered equivalent of ./install.

Thanks for finding the correct names though Now we need pacman variants and freebsd...

BTW, I am 99% sure that "xorg-x11-server-devel freetype-devel expat-devel" are not needed.

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by dolik.rce on Fri, 03 Apr 2020 15:00:23 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 03 April 2020 09:31Are you running as regular user? Do you have .config folder inside the unpacked archive? What is inside? There should be umks32 folder and inside autogenerated CLANG.bm (it is after I do this on ubuntu or fedora).

There is only GCC.bm in .config/umks32/. Note that I'm using Arch and I do not have clang installed. The genrated GCC.bm seems to work just fine, it's just that the script is hardcoded to

use CLANG.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 03 Apr 2020 15:17:56 GMT
View Forum Message <> Reply to Message

[quote title=mirek wrote on Fri, 03 April 2020 16:59]amrein wrote on Fri, 03 April 2020 15:25mirek wrote on Fri, 03 April 2020 11:56
What do you think?

Well, at that point, w.r.t. dependencies, this is just overengineered equivalent of ./install.

Thanks for finding the correct names though Now we need pacman variants and freebsd...

BTW, I am 99% sure that "xorg-x11-server-devel freetype-devel expat-devel" are not needed.

Mirek

Perhaps. I'm playing with it right now. I'm trying to find Fedora 31 dependencies. Apparently, there is more dependencies than just gcc-c++ and make for umk.
You talked about it in another thread: zlib-devel bzip2-devel

Is this Makefile ok for you? I mean, shoudl I continue working on it than commit the change after test on BSD and Linux?

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 16:02:45 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Fri, 03 April 2020 17:00mirek wrote on Fri, 03 April 2020 09:31Are you running as regular user? Do you have .config folder inside the unpacked archive? What is inside? There should be umks32 folder and inside autogenerated CLANG.bm (it is after I do this on ubuntu or fedora).

There is only GCC.bm in .config/umks32/. Note that I'm using Arch and I do not have clang installed. The genrated GCC.bm seems to work just fine, it's just that the script is hardcoded to use CLANG.

Ah, that explains it. Thank you.

Script indeed is hardcoded CLANG as CLANG is build requirement. Original idea was to install it by that automatic thing but I do not yet have arch support.

As for hardcoding CLANG, for now it is for simple reason that it is faster compiler. It is also supported everywhere anyway. In future, I would like to add llvm debugging libraries to theide, so then it will be "hard" requirement anyway...

Plus Win32 is now clang based as well as MacOS, so...

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 03 Apr 2020 16:24:24 GMT
View Forum Message <> Reply to Message

[quote title=amrein wrote on Fri, 03 April 2020 17:17]mirek wrote on Fri, 03 April 2020 16:59
Is this Makefile ok for you? I mean, shoudl I continue working on it than commit the change after test on BSD and Linux?

I want to be able to ./install U++ (actualy, maybe instead of ./install, the better name is ./config). Actually, at this point, even if I am persuaded by community to drop it, I will probably keep maintaining it for myself.

What you are trying to do is formally correct, but it is just a new slightly improved iteration of current tarball.

Instead of trying that, I think it would be better to invest time to provide binary packages (again). For tarball, simple Makefile with readme is enough. I plan to start autogenerating Makefile to repo really soon.

Oh, and BTW, pkg-config is not enough to identify missing packages, fresh fedora does not even have 'make' installed.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 03 Apr 2020 20:20:41 GMT
View Forum Message <> Reply to Message

[quote title=mirek wrote on Fri, 03 April 2020 18:24]amrein wrote on Fri, 03 April 2020 17:17mirek wrote on Fri, 03 April 2020 16:59
Is this Makefile ok for you? I mean, shoudl I continue working on it than commit the change after test on BSD and Linux?

I want to be able to ./install U++ (actualy, maybe instead of ./install, the better name is ./config). Actually, at this point, even if I am persuaded by community to drop it, I will probably keep maintaining it for myself.

What you are trying to do is formally correct, but it is just a new slightly improved iteration of current tarball.

Instead of trying that, I think it would be better to invest time to provide binary packages (again). For tarball, simple Makefile with readme is enough. I plan to start autogenerating Makefile to repo really soon.

Oh, and BTW, pkg-config is not enough to identify missing packages, fresh fedora does not even have 'make' installed.

Yes. Pkg-config is not enough. This is why the last Makefile search for make pkg-config and c++ binaries.
I need to test this makefile on BSD and MacOS to see how it works. It's certainly too much gmake dependent.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Sat, 04 Apr 2020 13:17:54 GMT
View Forum Message <> Reply to Message

Link in the first post updated to new version, it now asks more before doing "naughty" things...

Should now work with debians and rpms (to be tested...)

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by deep on Mon, 06 Apr 2020 07:53:32 GMT
View Forum Message <> Reply to Message

Hi

I get following error in compilation.

ubuntu 16.04 , http://ultimatepp.org/downloads/upp-posix-14219.tar.xz

run ./install

CppBuilder.cpp
In file included from
/home/deepak/src/upp/.cache/upp.out/ide/Common/CLANG.Blitz.Gui.Shared/ide/Common$blitz.cpp:3:
In file included from /home/deepak/src/upp/uppsrc/ide/Common/ComDlg.cpp:1:

```
In file included from /home/deepak/src/upp/uppsrc/ide/Common/Common.h:5:
In file included from /home/deepak/src/upp/uppsrc/CtrlLib/CtrlLib.h:4:
In file included from /home/deepak/src/upp/uppsrc/CtrlCore/CtrlCore.h:1657:
/home/deepak/src/upp/uppsrc/CtrlCore/GtkAfter.h:3:2: error: unknown type name
'GdkDrawingContext'; did you mean 'GdkDragContext'?
        GdkDrawingContext *ctx;
        ^~~~~~~~~~~~~~~~~
        GdkDragContext
/usr/include/gtk-3.0/gdk/gdktypes.h:137:39: note: 'GdkDragContext' declared here
typedef struct _GdkDragContext        GdkDragContext;
                                      ^
1 error generated.
```

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Mon, 06 Apr 2020 08:01:45 GMT
View Forum Message <> Reply to Message

Here is an updated buildrequires.posix file Mirek.

I tested umk and theide dependencies in those POSIX distributions.

File Attachments
1) buildrequires.posix, downloaded 431 times

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Mon, 06 Apr 2020 08:18:45 GMT
View Forum Message <> Reply to Message

amrein wrote on Mon, 06 April 2020 10:01Here is an updated buildrequires.posix file Mirek.

I tested umk and theide dependencies in those POSIX distributions.

Thanks, this is very helpfull. I think we should add openssl too - while not required for theide at the moment, many examples and a lot of code needs it right away.

I believe it is "openssl-devel" in rpms.

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Mon, 06 Apr 2020 08:33:46 GMT

deep wrote on Mon, 06 April 2020 09:53Hi

I get following error in compilation.

ubuntu 16.04 , http://ultimatepp.org/downloads/upp-posix-14219.tar.xz

run ./install

```
CppBuilder.cpp
In file included from
/home/deepak/src/upp/.cache/upp.out/ide/Common/CLANG.Blitz.Gui.Shared/ide/Common$blitz.c
pp:3:
In file included from /home/deepak/src/upp/uppsrc/ide/Common/ComDlg.cpp:1:
In file included from /home/deepak/src/upp/uppsrc/ide/Common/Common.h:5:
In file included from /home/deepak/src/upp/uppsrc/CtrlLib/CtrlLib.h:4:
In file included from /home/deepak/src/upp/uppsrc/CtrlCore/CtrlCore.h:1657:
/home/deepak/src/upp/uppsrc/CtrlCore/GtkAfter.h:3:2: error: unknown type name
'GdkDrawingContext'; did you mean 'GdkDragContext'?
        GdkDrawingContext *ctx;
        ^~~~~~~~~~~~~~~~~
        GdkDragContext
/usr/include/gtk-3.0/gdk/gdktypes.h:137:39: note: 'GdkDragContext' declared here
typedef struct _GdkDragContext        GdkDragContext;
                                      ^
1 error generated.
```

Thats actually a bug in CtrlCore. Should be fixed tomorrow; or you can fix CtrlCore/GtkAfter.h

```
class ViewDraw : public SystemDraw {
#if GTK_CHECK_VERSION(3, 22, 0) // <---- wrong version here
 GdkDrawingContext *ctx;
#endif
public:
 ViewDraw(Ctrl *ctrl);
 ~ViewDraw();
};
```

Thanks for reporting!

Subject: Re: New experimental Linux/FreeBSD release
Posted by deep on Mon, 06 Apr 2020 09:52:54 GMT

Hi Mirek

Thanks. I changed the file as per your message.

It is working now.

I have now .config and .cache folders.

Could not locate log file. which used to be in .upp folder.

Serialised data "cfg" file is in .config/u++/.... folder.

EDIT : Now I have log files. On first run there was no log file created.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Mon, 06 Apr 2020 11:45:06 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 06 April 2020 10:18amrein wrote on Mon, 06 April 2020 10:01Here is an updated buildrequires.posix file Mirek.

I tested umk and theide dependencies in those POSIX distributions.

Thanks, this is very helpfull. I think we should add openssl too - while not required for theide at the moment, many examples and a lot of code needs it right away.

I believe it is "openssl-devel" in rpms.

Mirek

* openssl-devel on Fedora, on Redhat and on Mandriva.
* libopenssl-devel on OpenSuse
* libssl-dev on Debian and Ubuntu
* Nothing to install on FreeBSD (installed by default) but libcrypto.pc, libssl.pc and openssl.pc are missing

---

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Mon, 06 Apr 2020 11:59:48 GMT
View Forum Message <> Reply to Message

amrein wrote on Mon, 06 April 2020 13:45
* openssl-devel on Fedora, on Redhat and on Mandriva.
* libopenssl-devel on OpenSuse

* libssl-dev on Debian and Ubuntu
* Nothing to install on FreeBSD (installed by default) but libcrypto.pc, libssl.pc and openssl.pc are missing

Thanks a lot. Looks like not moving everything to pkg-config was a good idea...

So I guess we have blind spot remaining now - pacman / ArchLinux and derivatives.... It unfortunately keeps crashing in virtualbox for me....

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Mon, 06 Apr 2020 13:05:22 GMT
View Forum Message <> Reply to Message

Last fixes on buildrequires.posix (mandriva => openmandriva + dependencies and package manager fixes):
OpenMandriva use clang by default...

openmandrivalinux.umk.dependencies: clang make zlib-devel bzip2-devel glibc-devel
openmandrivalinux.umk.pkg-config: zlib bzip2
openmandrivalinux.theide.dependencies: gtk+3-devel libnotify-devel # gtk+3-devel provided by lib64gtk+3.0-devel
openmandrivalinux.theide.pkg-config: gtk+-3.0 libnotify # x11 freetype2
openmandrivalinux.extra.dependencies: openssl-devel # openssl-devel provided by lib64openssl-devel
openmandrivalinux.extra.pkg-config: libcrypto libssl openssl # All are in openssl
openmandrivalinux.install.command: dnf install

Subject: Re: New experimental Linux/FreeBSD release
Posted by dolik.rce on Mon, 06 Apr 2020 14:18:05 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 06 April 2020 13:59So I guess we have blind spot remaining now - pacman / ArchLinux and derivatives.... It unfortunately keeps crashing in virtualbox for me....
This should work:

arch.umk.dependencies: gcc make zlib bzip2
arch.umk.pkg-config: zlib bzip2
arch.theide.dependencies: gtk3 libnotify
arch.theide.pkg-config: gtk+-3.0 libnotify # x11 freetype2
arch.install.command: sudo pacman -Sy --needed

I didn't actually test it, just checked the package names on my machine... But it should work Also, gcc could be changed to clang, but gcc is probably already installed on most developers systems, since it is in base-devel group (unlike clang).

Honza

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Mon, 06 Apr 2020 14:47:24 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Mon, 06 April 2020 16:18mirek wrote on Mon, 06 April 2020 13:59So I guess we have blind spot remaining now - pacman / ArchLinux and derivatives.... It unfortunately keeps crashing in virtualbox for me....
This should work:

arch.umk.dependencies: gcc make zlib bzip2
arch.umk.pkg-config: zlib bzip2
arch.theide.dependencies: gtk3 libnotify
arch.theide.pkg-config: gtk+-3.0 libnotify # x11 freetype2
arch.install.command: sudo pacman -Sy --needed

I didn't actually test it, just checked the package names on my machine... But it should work Also, gcc could be changed to clang, but gcc is probably already installed on most developers systems, since it is in base-devel group (unlike clang).

Honza

Thanks a lot. clang name is "clang", right?

Subject: Re: New experimental Linux/FreeBSD release
Posted by dolik.rce on Mon, 06 Apr 2020 14:50:37 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 06 April 2020 16:47Thanks a lot. clang name is "clang", right?

Yes, clang packge is called clang. Oh, and I forgot about the openssl, which is in openssl package Arch tries to keep it simple

Subject: Re: New experimental Linux/FreeBSD release
Posted by pvictor on Tue, 07 Apr 2020 08:28:07 GMT
View Forum Message <> Reply to Message

In the latest build (14263) the first line in Makefile is:
UPPDIR1 = /home/cxl/upp.src/uppsrc/

I had to replace it with:

UPPDIR1 = uppsrc/
in order to run make.

Victor

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Tue, 07 Apr 2020 12:35:30 GMT
View Forum Message <> Reply to Message

Should be now fixed.

Subject: Re: New experimental Linux/FreeBSD release
Posted by Oblivion on Wed, 08 Apr 2020 08:41:08 GMT
View Forum Message <> Reply to Message

Hello Mirek,

There seems to be another problem on Linux with the latest code.

Logs are written to.cache/upp.out, but I get garbled log file names (sometimes even free mem blocks (FreeFreeFree) leak to names).
Contents of log files are fine, but TheIDE can't open them as the file names are trashed.

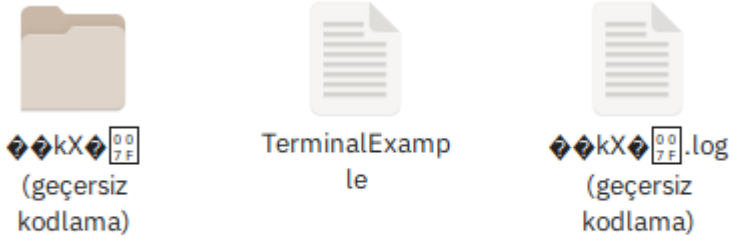
Screenshot:

System: ArchLinux/Gnome 3.36, both GCC and CLANG, Flags: GUI, Encoding UTF-8.

Best regards,
Oblivion

File Attachments

1) Garbled_Log_filenames.png, downloaded 1280 times



Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 08:58:22 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Wed, 08 April 2020 10:41Hello Mirek,

There seems to be another problem on Linux with the latest code.

Logs are written to.cache/upp.out, but I get garbled log file names (sometimes even free mem blocks (FreeFreeFree) leak to names).
Logs are fine but, naturally, TheIDE can't open them as the file names are trashed.

Screenshot:

System Arch/Linux/Gnume 3.36, both GCC and CLANG, Flags: GUI, Encoding UTF-8).

Best regards,
Oblivion

Thanks, sorry for trouble. Just a quick question, what is the name of the application? Do you have non-ASCII names in it?

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 09:08:19 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Wed, 08 April 2020 10:41Hello Mirek,

There seems to be another problem on Linux with the latest code.

Logs are written to.cache/upp.out, but I get garbled log file names (sometimes even free mem blocks (FreeFreeFree) leak to names).

That is already wrong, it should be .config/u++/{AppName}

I have now found and fixed a problem when LOGs started before MAIN. Could that be related?

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Wed, 08 Apr 2020 09:25:39 GMT
View Forum Message <> Reply to Message

Great. Now theide and umk uses sandboxed .config and .cache, and they use ~/.config/ and ~/.cache if there are not there.

Suggestion:

* use cache/upp/ and config/upp in tarball (instead of hidden directories)
* use config/upp/theide, config/upp/umk instead of config/u++/theide...
* share compiler .bm files between theide and umk (in config/upp/ and not in both config/upp/theide and config/upp/umk)
* fix code where ".upp" is still used (you can find some using "grep -R \\.upp uppsrc")
* move bazaar, examples, reference, tutorial in an extra directory (see layout at the end of this message)
* use share/upp in sandboxed upp directory or ~/.share/upp if the directory doesn't exist (for dictionaries and other future data)

I'm thinking right non on how to improve your current script and Makefiles. I will perhaps upload a few patches to review.

Directory layout:

```
cache
config
extra
  |--bazaar
  |--examples
  |--reference
  \--tutorial
share
  |--upp
       |--theide
            |--speller
uppsrc
```

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Wed, 08 Apr 2020 09:37:50 GMT
View Forum Message <> Reply to Message

I don't know if it happens to everyone but:

* I always got /home/user/bazaar in all assembly, even if this directory doesn't exist and even if I remove all configuration folders.

* I have /home/user/uppsrc in all assembly, if ~/upp.src doesn't exist.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 09:41:50 GMT
View Forum Message <> Reply to Message

amrein wrote on Wed, 08 April 2020 11:25Great. Now theide and umk uses .config and .cache inside the tarball and ~/.config/ or ~/.cache if there are not there.

Suggestion:

* use cache/upp/ and config/upp in tarball (instead of hidden directories)

On some level, I think this makes sense, but having these names the same has a lot of sense too. So I would vote for .config.

Quote:
* use config/upp/theide, config/upp/umk instead of config/u++/theide...

I suspected that u++ might not be liked, but is that really a problem? IMO it looks cool Also nameclash is very unlikely.

Quote:
* share compiler .bm files between theide and umk (in config/upp/ and not in both config/upp/theide and config/upp/umk)

Will do, but will probably mean putting it to .config/theide

Quote:
* fix code where ".upp" is still used (you can find some using "grep -R \\.upp uppsrc")

OK, but not all ".upp" are config related, in most cases it is extension for package files

Quote:
* move bazaar examples reference tutorial in an extra directory (see layout at the end of this message)

I do not like that.

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by Oblivion on Wed, 08 Apr 2020 10:27:38 GMT
View Forum Message <> Reply to Message

Edit 1: I'll investigate further, this may be a problem on my side. I'll let you know.
Edit 2: This occurs if TIMING macro is in Paint();
Edit 3: This occurs only on GTK. NOGTK mode is not affected. (A lock issue? What's changed in GTK backend, lately?)

Quote:Just a quick question, what is the name of the application? Do you have non-ASCII names in it?

It is "TerminalExample", nothing more.

By the way, I've checked out SVN, cleared the cache, and manually set up .config/u++. Unfortunately, it didn't change the situation.

However, I've now noticed that this occurs only when TIMING macros are used (specifically, RTIMING).
Also DuMPS and LOGS seem to be acting weirdly (not dumping, or logging) when TIMING macros are used.
Not to mention each RTIMING macro adds ~8 ms. delay to overall result. (Maybe related, maybe not, I don't know.)

Best regards,
Oblivion

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 11:51:25 GMT
View Forum Message <> Reply to Message

OK, that rings some bells...

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 12:09:42 GMT
View Forum Message <> Reply to Message

amrein wrote on Wed, 08 April 2020 11:37I don't know if it happens to everyone but:

* I always got /home/user/bazaar in all assembly, even if this directory doesn't exist and even if I remove all configuration folders.
* I have /home/user/uppsrc in all assembly, if ~/upp.src doesn't exist.

What is "all assembly" ?

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 12:17:57 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Wed, 08 April 2020 12:27
Not to mention each RTIMING macro adds ~8 ms. delay to overall result. (Maybe related, maybe not, I don't know.)

BTW, not quite sure what you mean by that, but maybe that is expected behaviour. In overall result, the time spent by time measuring code is included, then it is (approximately) subtracted in time per single pass.

(The implementation simply measures how long it takes to do a large amount of timings of NOP and then uses this value in all calculations).

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by Oblivion on Wed, 08 Apr 2020 12:35:03 GMT
View Forum Message <> Reply to Message

Quote:In overall result, the time spent by time measuring code is included, then it is (approximately) subtracted in time per single pass.

Sure, I understand that. It is just that I'm having a spike in delay added by timing macros too (in Paint()), after I last updated U++, a day ago. I've pointed it out, because I thoulght it might be related to the main issue I'm having with timing macros on GTK backend.

Best regards,
Oblivion

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 12:50:13 GMT
View Forum Message <> Reply to Message

So my theory about issues with log/TIMING is that the problem is that some static Strings are already destructed by the time .log is created.

I have now tried to "fortify" the code so that this does not happen, can you retest with current trunk please?

Subject: Re: New experimental Linux/FreeBSD release
Posted by Oblivion on Wed, 08 Apr 2020 12:58:03 GMT
View Forum Message <> Reply to Message

Yep, the problem disappeared. timing macros seem to work as expected now.

Thank you very much!

Best regards,
Oblivion

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Wed, 08 Apr 2020 14:05:39 GMT
View Forum Message <> Reply to Message

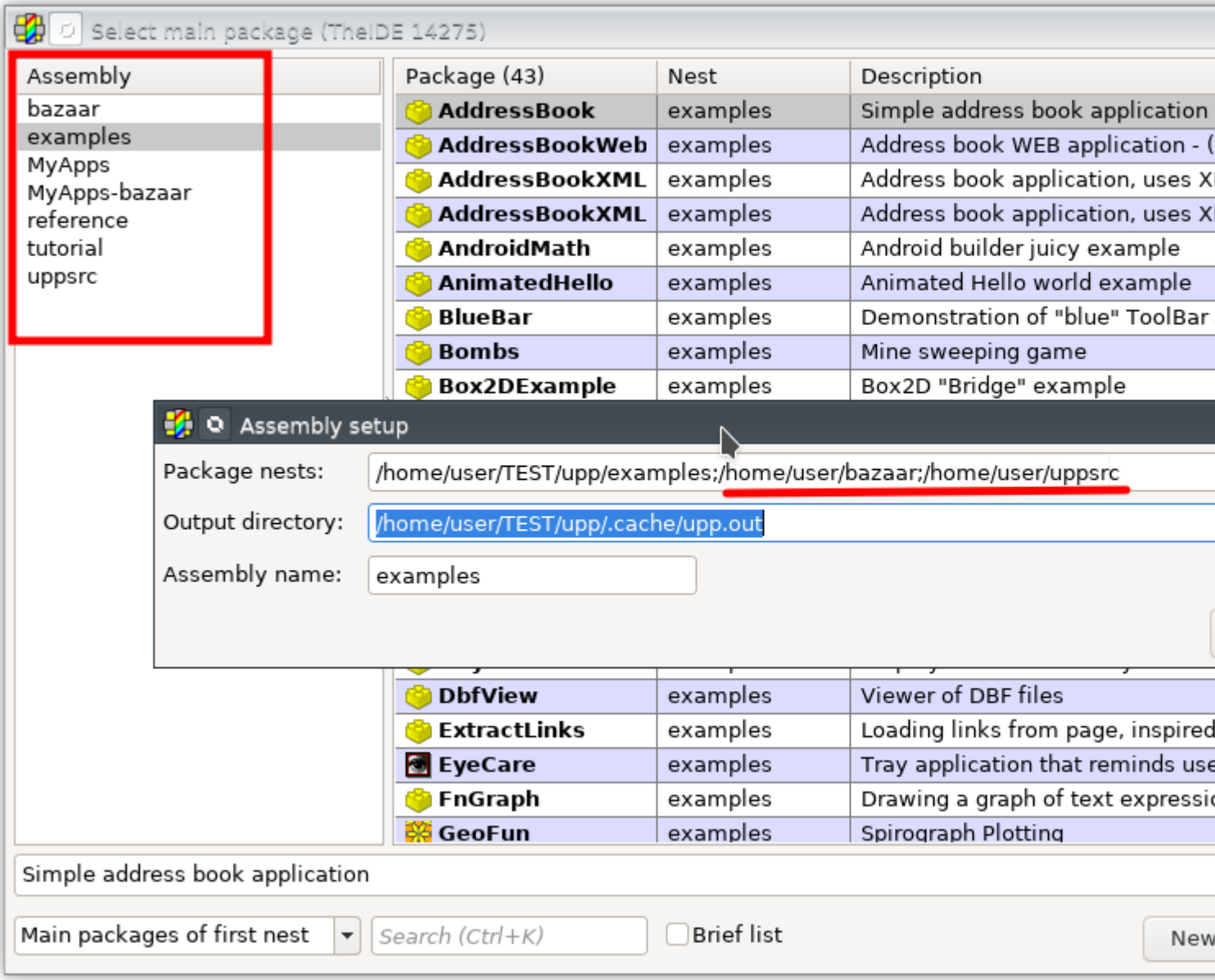
mirek wrote on Wed, 08 April 2020 14:09amrein wrote on Wed, 08 April 2020 11:37I don't know if it happens to everyone but:

* I always got /home/user/bazaar in all assembly, even if this directory doesn't exist and even if I remove all configuration folders.
* I have /home/user/uppsrc in all assembly, if ~/upp.src doesn't exist.

What is "all assembly" ?

All assembly on the left (inside the red rectangle) use "/home/user/bazaar;/home/user/uppsrc". Only uppsrc assembly uses the correct nest (/home/user/TEST/upp/uppsrc).

File Attachments
1) Screenshot_20200408_154954.png, downloaded 1364 times



Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 08 Apr 2020 16:24:44 GMT
View Forum Message <> Reply to Message

amrein wrote on Wed, 08 April 2020 11:37I don't know if it happens to everyone but:

* I always got /home/user/bazaar in all assembly, even if this directory doesn't exist and even if I remove all configuration folders.
* I have /home/user/uppsrc in all assembly, if ~/upp.src doesn't exist.

Thanks, needed a big more tweaking, should be now fixed.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Wed, 08 Apr 2020 19:10:38 GMT
View Forum Message <> Reply to Message

Yes. It's fixed.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Wed, 08 Apr 2020 19:19:14 GMT
View Forum Message <> Reply to Message

I can't find where this bug come from (I guess it's a bug).

When I use Makefiles, I get only one file named .cache/upp.out/build_info.h
When I use umk or umks32, I get several files named build_info.h in .cache/upp.out/

Quote:
rm -fr .cache/upp.out/*
./umk ./uppsrc umk CLANG -brs ./umk
find .cache/upp.out/ -name build_info.h

Result:

Quote:
.cache/upp.out/Core/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/pcre/CLANG.Shared/build_info.h
.cache/upp.out/plugin/zstd/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/lz4/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/lzma/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/bz2/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/Esc/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/umk/CLANG.Blitz.Main.Shared/build_info.h
.cache/upp.out/ide/Java/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/ide/Android/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/ide/Core/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/ide/Builders/CLANG.Shared/build_info.h

Those files don't have the same content (the defined variables inside don't have the same value most of the time).

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Thu, 09 Apr 2020 08:22:42 GMT
View Forum Message <> Reply to Message

amrein wrote on Wed, 08 April 2020 21:19I can't find where this bug come from (I guess it's a

bug).

When I use Makefiles, I get only one file named .cache/upp.out/build_info.h
When I use umk or umks32, I get several files named build_info.h in .cache/upp.out/

Quote:
rm -fr .cache/upp.out/*
./umk ./uppsrc umk CLANG -brs ./umk
find .cache/upp.out/ -name build_info.h

Result:

Quote:
.cache/upp.out/Core/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/pcre/CLANG.Shared/build_info.h
.cache/upp.out/plugin/zstd/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/lz4/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/lzma/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/plugin/bz2/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/Esc/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/umk/CLANG.Blitz.Main.Shared/build_info.h
.cache/upp.out/ide/Java/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/ide/Android/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/ide/Core/CLANG.Blitz.Shared/build_info.h
.cache/upp.out/ide/Builders/CLANG.Shared/build_info.h

Those files don't have the same content (the defined variables inside don't have the same value most of the time).

Let us say "feature".

From utility perspective, it is better when build_info is in all package output files. This however was not yet implemented for Makefile and is causing little problems that way yet. I will file this as low-priority issue in redmine.

EDIT: Also, these defines should be treated by the code as "optional additional information" (with #ifdef)...

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Sat, 11 Apr 2020 08:01:04 GMT
View Forum Message <> Reply to Message

Little fix for your Makefiles

Summary:

* replace hard coded reference of .cache/upp.out by $(UPPOUT)
* secure the line that remove $(UPPOUT) (clean target)
* remove unneeded line, the last line that remove build_info.h (this file is in $(UPPOUT))

Here is the patch. Of course, you will certainly fix umk/theide code instead I guess.

```
$ diff Makefile Makefile.ide
3,4c3,4
< UPPOUT = .cache/upp.out/
< CINC   =  -I$(UPPDIR1) `pkg-config --cflags freetype2` `pkg-config --cflags x11` `pkg-config
--cflags fontconfig` `pkg-config --cflags xcb` `pkg-config --cflags expat` `pkg-config --cflags
xinerama` `pkg-config --cflags xrender` `pkg-config --cflags xft` `pkg-config --cflags xdmcp`
`pkg-config --cflags xext` `pkg-config --cflags gtk+-3.0` `pkg-config --cflags libnotify` `pkg-config
--cflags libpng` -I./ -I.cache/upp.out
---
> UPPOUT = .cache/upp.out/
> CINC   =  -I$(UPPDIR1) `pkg-config --cflags freetype2` `pkg-config --cflags x11` `pkg-config
--cflags fontconfig` `pkg-config --cflags xcb` `pkg-config --cflags expat` `pkg-config --cflags
xinerama` `pkg-config --cflags xrender` `pkg-config --cflags xft` `pkg-config --cflags xdmcp`
`pkg-config --cflags xext` `pkg-config --cflags gtk+-3.0` `pkg-config --cflags libnotify` `pkg-config
--cflags libpng` -I./ -I$(UPPOUT)
94c94
< OutFile = ./ide
---
> OutFile = ide.out
103c103
<       echo '#define bmUSER    "'`whoami`'"') > ".cache/upp.out/build_info.h"
---
>       echo '#define bmUSER    "'`whoami`'"') > "$(UPPOUT)build_info.h"
72308,72309c72308,72309
72308,72309c72308,72309
<       if [ -d "$(UPPOUT)" ]; then rm -rf "$(UPPOUT)" ; fi
<       if [ -f build_info.h ]; then rm -f build_info.h ; fi
---
>       if [ "$(UPPOUT)" != "" -a "$(UPPOUT)" != "/" -a -d "$(UPPOUT)" ] ; then rm -fr "$(UPPOUT)"
; fi
>
```

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Sat, 11 Apr 2020 08:30:36 GMT
View Forum Message <> Reply to Message

amrein wrote on Sat, 11 April 2020 10:01Little fix for your Makefiles

Summary:

* replace hard coded reference of .cache/upp.out by $(UPPOUT)
* secure the line that remove $(UPPOUT) (clean target)
* remove unneeded line, the last line that remove build_info.h (this file is in $(UPPOUT))

Here is the patch. Of course, you will certainly fix umk/theide code instead I guess.

Do you realize these are generated by umk?

Sending Makefile patches is meaningless. It is ide/Builders code that needs patching...

(I might do that soon though...)

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Sat, 11 Apr 2020 10:51:05 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 11 April 2020 10:30amrein wrote on Sat, 11 April 2020 10:01Little fix for your Makefiles

Summary:

* replace hard coded reference of .cache/upp.out by $(UPPOUT)
* secure the line that remove $(UPPOUT) (clean target)
* remove unneeded line, the last line that remove build_info.h (this file is in $(UPPOUT))

Here is the patch. Of course, you will certainly fix umk/theide code instead I guess.

Do you realize these are generated by umk?

Sending Makefile patches is meaningless. It is ide/Builders code that needs patching...

(I might do that soon though...)

Mirek

Of course I know . The patch gives a clear idea to where the changes need to occur.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Sun, 12 Apr 2020 09:58:26 GMT
View Forum Message <> Reply to Message

[quote title=amrein wrote on Sat, 11 April 2020 12:51][quote title=mirek wrote on Sat, 11 April 2020 10:30]amrein wrote on Sat, 11 April 2020 10:01
Of course I know . The patch gives a clear idea to where the changes need to occur.

Hopefully implemented in trunk. Please check.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Sun, 12 Apr 2020 11:03:47 GMT
View Forum Message <> Reply to Message

All done. It works perfectly as expected.

Small issue: if umk create a Makefile from inside uppsrc, it will use "OutFile = packagename" instead of "OutFile = packagename.out"

Example: "OutFile = umk" instead of "OutFile = umk.out"

The consequence is an error at link time because the directory name (umk or ide for example) already exist.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Sun, 12 Apr 2020 21:09:13 GMT
View Forum Message <> Reply to Message

amrein wrote on Sun, 12 April 2020 13:03All done. It works perfectly as expected.

Small issue: if umk create a Makefile from inside uppsrc, it will use "OutFile = packagename" instead of "OutFile = packagename.out"

Example: "OutFile = umk" instead of "OutFile = umk.out"

The consequence is an error at link time because the directory name (umk or ide for example) already exist.

Afaik it uses this always. But it definitely makes sense to change to umk.out if dir there exists..

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Tue, 14 Apr 2020 12:23:09 GMT
View Forum Message <> Reply to Message

Today, I share an early version of the new Makefile to get developers opinions.

I'm sharing the Makefile and an empty domake that shows how Makefile will call it.
This domake doesn't do anything so you can type 'make' without worrying. Only 'make clean' target is 'dangerous' here.

Type 'make' to see how it works.

File Attachments
1) makefile-and-domake-20200414.zip, downloaded 387 times

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Thu, 16 Apr 2020 15:10:57 GMT
View Forum Message <> Reply to Message

For what is worth, I have tested nightly posix release with Raspberry PI 3 1GB and raspbian and all works flawlessly. 25 minutes to get theide running.

Building UWord takes about 7:30....

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Sat, 18 Apr 2020 10:11:39 GMT
View Forum Message <> Reply to Message

Bug. With the new umk, there is no more difference between generated Makefiles:

umk ../uppsrc ide CLANG +POSIX,LINUX -rsM ; mv -f Makefile Makefile.ide.linux
umk ../uppsrc ide CLANG +POSIX,OSX11 -rsM ; mv -f Makefile Makefile.ide.osx11
umk ../uppsrc ide CLANG +POSIX,WIN32 -rsM ; mv -f Makefile Makefile.ide.win32

md5sum $(ls Makefile.* | sort)

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Sun, 19 Apr 2020 15:17:52 GMT
View Forum Message <> Reply to Message

Thats actually bugfix

Previously umk did not differntiate between output binary name and makefile.

To specify makefile name, you now can use M=makefile
(https://www.ultimatepp.org/appideumk$en-us.html).

(BTW, I started work on improved commandline, however I expect final product after 2020.1 release).

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Mon, 20 Apr 2020 05:59:18 GMT
View Forum Message <> Reply to Message

I tested the last rev-14335 on Linux. The -M= option works according to specification but the generated makefiles are still the same.

There are identical, regardless of passed flags (LINUX, X11, WIN32, OSX11):

```
umk ../uppsrc ide CLANG +POSIX,LINUX     -rs -M=Makefile.ide.LINUX
umk ../uppsrc ide CLANG +POSIX,LINUX,X11 -rs -M=Makefile.ide.LINUX-X11
umk ../uppsrc ide CLANG +POSIX,WIN32     -rs -M=Makefile.ide.WIN32
umk ../uppsrc ide CLANG +POSIX,OSX11     -rs -M=Makefile.ide.OSX11

md5sum Makefile.ide.*
```

Résultat:

```
ade7b1a69e50e605b4a113fe761c0d0b  Makefile.ide.LINUX
ade7b1a69e50e605b4a113fe761c0d0b  Makefile.ide.LINUX-X11
ade7b1a69e50e605b4a113fe761c0d0b  Makefile.ide.OSX11
ade7b1a69e50e605b4a113fe761c0d0b  Makefile.ide.WIN32
```

With an older version, umk rev-14205, the resulting files are differents:

```
umk.old ../uppsrc ide CLANG +POSIX,LINUX     -rsM ; mv -f Makefile Makefile.ide.LINUX
umk.old ../uppsrc ide CLANG +POSIX,LINUX,X11 -rsM ; mv -f Makefile Makefile.ide.LINUX-X11
umk.old ../uppsrc ide CLANG +POSIX,WIN32     -rsM ; mv -f Makefile Makefile.ide.WIN32
umk.old ../uppsrc ide CLANG +POSIX,OSX11     -rsM ; mv -f Makefile Makefile.ide.OSX11

md5sum Makefile.ide.*
```

12ee3d49848f69d4034381bebe91a210 Makefile.ide.LINUX
458b67aa2e9f7d652aa6ef4a40cd822c Makefile.ide.LINUX-X11
7b0ce34377577d122d34ac644d01e99f Makefile.ide.OSX11
002fd3fecfb6dc4948b497cc7e835fc2 Makefile.ide.WIN32

I searched but couldn't find the bug.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Mon, 20 Apr 2020 08:39:17 GMT
View Forum Message <> Reply to Message

I am sorry for misunderstanding your bug report. Should be now fixed in the trunk.

I have also started printing flags in verbose mode...

Mirek

Subject: Re: New experimental Linux/FreeBSD release
Posted by pvictor on Wed, 22 Apr 2020 16:28:14 GMT
View Forum Message <> Reply to Message

Missing } at the end of CtrlCore/UtilX11.cpp (Upp 14346)

Victor

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 22 Apr 2020 17:40:39 GMT
View Forum Message <> Reply to Message

Thank you, fixed...

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Wed, 22 Apr 2020 21:35:20 GMT
View Forum Message <> Reply to Message

Experimental Linux/FreeBSD/OpenBSD/OSX11 release : http://dl.free.fr/oLYL88VWq

Not official. Only for testing.

This is what I was working on.

It's not so complex and it should work on every POSIX OS. At least on Linux, FreeBSD and OpenBSD.
It also support rpm distribution (with 'rpmbuild -tb upp-14352MS.tar.xz')

(Patch for OpenBSD Package Organizer configs included).

I stop there because I think Mirek wants something simpler, smaller and easier to maintain.
At least, you can use the buildrequires.posix file to enhance the current posix release tarball.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Wed, 22 Apr 2020 22:56:01 GMT
View Forum Message <> Reply to Message

amrein wrote on Wed, 22 April 2020 23:35Experimental Linux/FreeBSD/OpenBSD/OSX11 release : http://dl.free.fr/oLYL88VWq

Not official. Only for testing.

This is what I was working on.
It's not so complex and it should work on every POSIX OS. At least on Linux, FreeBSD and OpenBSD.
It also support rpm distribution (with 'rpmbuild -tb upp-14352MS.tar.xz')

(Patch for OpenBSD Package Organizer configs included).

I stop there because I think Mirek wants something simpler, smaller and easier to maintain.
At least, you can use the buildrequires.posix file to enhance the current posix release tarball.

Thank you!

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 24 Apr 2020 09:57:29 GMT
View Forum Message <> Reply to Message

There's a strange error when I use umk to build theide.
At link time, at the end of the wall build process, it fails saying:

...
Linking...
clang-9: error: unsupported option '--libs'
clang-9: error: unsupported option '--libs'
...

clang-9: error: no such file or directory: '`pkg-config'
clang-9: error: no such file or directory: 'freetype2`'
clang-9: error: no such file or directory: '`pkg-config'
clang-9: error: no such file or directory: 'x11`'
clang-9: error: no such file or directory: '`pkg-config'
clang-9: error: no such file or directory: 'fontconfig`'
...

My command line (fresh build umk rev 14364):

./umk uppsrc ide CLANG -rbs ./theide

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 24 Apr 2020 14:17:02 GMT
View Forum Message <> Reply to Message

amrein wrote on Fri, 24 April 2020 11:57There's a strange error when I use umk to build theide. At link time, at the end of the wall build process, it fails saying:

...
Linking...
clang-9: error: unsupported option '--libs'
clang-9: error: unsupported option '--libs'
...
clang-9: error: no such file or directory: '`pkg-config'
clang-9: error: no such file or directory: 'freetype2`'
clang-9: error: no such file or directory: '`pkg-config'
clang-9: error: no such file or directory: 'x11`'
clang-9: error: no such file or directory: '`pkg-config'
clang-9: error: no such file or directory: 'fontconfig`'
...

My command line (fresh build umk rev 14364):

./umk uppsrc ide CLANG -rbs ./theide

Sorry, my mistake. I was adding response file support because of win32 commandline limitation, clashed with ` ` expansions. Will fix ASAP.

Subject: Re: New experimental Linux/FreeBSD release
Posted by mirek on Fri, 24 Apr 2020 16:00:04 GMT
View Forum Message <> Reply to Message

Hopefully fixed.

Subject: Re: New experimental Linux/FreeBSD release
Posted by amrein on Fri, 24 Apr 2020 17:59:53 GMT
View Forum Message <> Reply to Message

It works again.