Subject: Things we want from Linux/FreeBSD release archives
Posted by mirek on Tue, 31 Mar 2020 15:52:26 GMT

View Forum Message <> Reply to Message

I am starting this new thread to gather ideas about how to improve current Linux tarballs.

Here is my list:

- automatic dependencies installation - detect existence of apt-get / yum / zypper / pacman, perhaps even distro, install things during setup. If that fails, make user aware and ask if script should continue

- use 7z (or at least zip), stop changing the name, just single "upp" folder

- stop moving things around, stop writing outside "upp" folder - basically same behaviour as Win32 and MacOS. theide will get generated inside unpacked archive and will stay there. I think users can and will move it where they want later. (That said, I am not quite sure at this point about ~/.upp. Easy to do for theide, not so sure about compiled apps).

- ship with umks and use it if possible. If not, use Makefile to create umk

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by dolik.rce on Tue, 31 Mar 2020 18:46:28 GMT

View Forum Message <> Reply to Message

Hi Mirek,

Quote:detect existence of apt-get / yum / zypper / pacman

Good idea, but there should also be a simple way to disable this. This will be appreciated by packagers who would like to create deb/rpm/PKGBUILD/... because in that case the tooling has its own ways to install prerequisites.

Quote:- use 7z (or at least zip), stop changing the name, just single "upp" folder

What's wrong with *.tar.gz? Or *.tar.xz which is actually what every major distro starts to use lately... The 7z extractor is not installed by default on many distributions (checked on Arch and Debian), so it is not really linux friendly...

Also I'm not sure what you mean by changing the name, but I'm pretty sure that the "x11" should be dropped from the filename.

Quote:- stop moving things around, stop writing outside "upp" folder - basically same behaviour as Win32 and MacOS. theide will get generated inside unpacked archive and will stay there. I think users can and will move it where they want later. (That said, I am not quite sure at this point about ~/.upp. Easy to do for theide, not so sure about compiled apps).
Definitely a good idea. I think the proper place for the things currently written in ~/.upp is actually

in ~/.config. Also, TheIDE should store compiled object in ~/.cache, since they ca be easily rebuilt. It should be possible to configure it, since U++ already uses XDG Base Directory specification. So the Makefile in the source package should set the necessary XDG_* environment variables to some local directory, e.g. upp/build/.

Quote:- ship with umks and use it if possible. If not, use Makefile to create umk
Better not put it directly into the source package. Downloading it during the process if needed is IMHO better. The Makefile could also optionally support to use already existing umk (autodetecting it's presence on $PATH or by explicit configuration option).

Few more points I would like to suggest:

 - Clean the files a bit, before packaging. I think the archive is pretty big and it wouldn't hurt to only include necessary files. E.g. 32MB of tppi could be omitted, as those are automatically created by TheIDE (if I understand it correctly). Perhaps they should not even be in the repository... There also might be some packages, that are no longer needed or maintained, or could be in separate nest, since they are not needed by regular users (yes TCore, I'm looking at you and your friends :)).

 - Follow common conventions as much as possible. It should be possible to build it just by running ./configure # this might not even be needed
make
make install
This is what most experienced user will try even before reading the README file. Which reminds me: there should be a README file :) Also, the "make install" part should honour the standard $DESTDIR and $PREFIX variables, to make it easier for packagers.

Best regards,
Honza

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Tue, 31 Mar 2020 21:24:17 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Tue, 31 March 2020 20:46Hi Mirek,

Quote:detect existence of apt-get / yum / zypper / pacman

Good idea, but there should also be a simple way to disable this. This will be appreciated by packagers who would like to create deb/rpm/PKGBUILD/... because in that case the tooling has its own ways to install prerequisites.

Well, in general, I just mean there should be quick and easy way to get U++ running.

There should still be Makefile and "normal" way.

Quote:- use 7z (or at least zip), stop changing the name, just single "upp" folder

What's wrong with *.tar.gz? Or *.tar.xz which is actually what every major distro starts to use lately... The 7z extractor is not installed by default on many distributions (checked on Arch and Debian), so it is not really linux friendly...
[/quote]

tar.gz compression sucks. LZMA seems to produce archives that are about 50% of size. But tar.xz is probably better than 7z.

Quote:
Also I'm not sure what you mean by changing the name, but I'm pretty sure that the "x11" should be dropped from the filename.

I mean that file name can be basically the same (withou x11), but after unpacking, it should be just upp

Quote:
Definitely a good idea. I think the proper place for the things currently written in ~/.upp is actually in ~/.config. Also, TheIDE should store compiled object in ~/.cache, since they ca be easily rebuilt. It should be possible to configure it, since U++ already

By not moving around I am actually thinking about putting everything to that "upp" folder after unpacking. Keep it "sandboxed", no writes outside that dir.

Quote:
 - Clean the files a bit, before packaging. I think the archive is pretty big and it wouldn't hurt to only include necessary files. E.g. 32MB of tppi could be omitted, as those are automatically created by TheIDE (if I understand it correctly).

Well, they are needed during make unfortunately... The logic would need to move to compile phase.

I think most of those MB are screenshot in documentation. Perhaps redoing them with "Gray" theme would help. Or using jpg compression.

Quote:
 Perhaps they should not even be in the repository... There also might be some packages, that are no longer needed or maintained, or could be in separate nest, since they are not needed by regular users (yes TCore, I'm looking at you and your friends :)).

TCore is not present for a long time now...

Quote:
 - Follow common conventions as much as possible. It should be possible to build it just by running [code]./configure # this might not even be needed
make
make install

Well, after changes I am planning I do not even know not only what ./configure would do, but even what make install is supposed to do.

After make, you get theide. Start theide and it configures itself.

Also, not really about release archive, but I think I will add nightly scrip that will add Makefile to svn so that it is buildable from git page.

Mirek

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Wed, 01 Apr 2020 04:30:07 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Tue, 31 March 2020 14:46since U++ already uses XDG Base Directory specification.
A quick question. Since when U++ is using XDG Base Directory specification?
In my local build all configuration files are located in ~/.upp

I vote for the XDG-Based directory structure.

It would be also great if ".upp" (or .config/upp) directory would be changeable. This should allow to create company/project specific configuration directories.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by dolik.rce on Wed, 01 Apr 2020 05:32:08 GMT
View Forum Message <> Reply to Message

Novo wrote on Wed, 01 April 2020 06:30A quick question. Since when U++ is using XDG Base Directory specification?
In my local build all configuration files are located in ~/.upp
Since about 2010 :) Unfortunately it is only used in some functions, e.g. GetDesktopFolder(). See Core/App.cpp for more examples... It should be relatively easy to use it also in GetHomeDirectory(), ConfigFile() and other functions.

Honza

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Wed, 01 Apr 2020 09:07:25 GMT
View Forum Message <> Reply to Message

So I would like this:


```
static String sConfigFolder;

void SetConfigDirectory(const String& s)
{
 sConfigFolder = s;
}

String  ConfigFile(const char *file) {
 if(sConfigFolder.GetCount())
  return AppendFileName(sConfigFolder, file);
#if defined(PLATFORM_WIN32)
 if(sHomecfg) {
  String p = GetHomeDirFile(GetExeTitle());
  ONCELOCK
   RealizeDirectory(p);
  return AppendFileName(p, file);
 }
 return GetExeDirFile(file);
#elif defined(PLATFORM_POSIX)
 static String cfgdir;
 ONCELOCK {
  String h = GetExeFolder();
  if(!sHomecfg)
   while(h.GetCount() > 1 && DirectoryExists(h)) {
    String pp = AppendFileName(h, ".config");
    if(DirectoryExists(pp)) {
     cfgdir = pp;
     break;
    }
    h = GetFileFolder(h);
   }
  if(IsNull(cfgdir))
   cfgdir = GetEnv("XDG_CONFIG_HOME");
  if(IsNull(cfgdir) || !DirectoryExists(cfgdir))
   cfgdir = GetHomeDirFile(".config");
 }
 String pp = AppendFileName(cfgdir, GetAppName());
 RealizeDirectory(pp);
 return AppendFileName(pp, file);
#else
 NEVER();
 return file;
```

```
#endif//PLATFORM
}
```

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Wed, 01 Apr 2020 16:17:24 GMT
View Forum Message <> Reply to Message

Is it possible to add "static String sAppName" in addition to sConfigFolder?
In this case for PLATFORM_WIN32 we will get
String p = GetHomeDirFile(Nvl(sAppName, GetExeTitle()));
And for PLATFORM_POSIX
String pp = AppendFileName(cfgdir, Nvl(sAppName, GetAppName()));
Otherwise we are limited by a name of an executable.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Wed, 01 Apr 2020 16:21:43 GMT
View Forum Message <> Reply to Message

Novo wrote on Wed, 01 April 2020 18:17Is it possible to add "static String sAppName" in addition
to sConfigFolder?
In this case for PLATFORM_WIN32 we will get
String p = GetHomeDirFile(Nvl(sAppName, GetExeTitle()));
And for PLATFORM_POSIX
String pp = AppendFileName(cfgdir, Nvl(sAppName, GetAppName()));
Otherwise we are limited by a name of an executable.

Already done 6 hours ago. SetAppName/GetAppName...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Wed, 01 Apr 2020 16:26:47 GMT
View Forum Message <> Reply to Message

Sorry. I missed that ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Wed, 01 Apr 2020 16:34:08 GMT
View Forum Message <> Reply to Message

I'm just curious, why are you using GetExeTitle() instead of GetAppName() with WIN32?
This seems to be inconsistent ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Wed, 01 Apr 2020 16:54:53 GMT
View Forum Message <> Reply to Message

Novo wrote on Wed, 01 April 2020 18:34I'm just curious, why are you using GetExeTitle() instead of GetAppName() with WIN32?
This seems to be inconsistent ...

Because I prefer not to change win32 sources while developing in Linux? :)

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Thu, 02 Apr 2020 17:10:49 GMT
View Forum Message <> Reply to Message

My ~/.config gets quickly littered by dozens of my dev apps.
Another problem is conflicting names. ~/.config is a system-wide namespace.

I propose to add a sAppNamePrefix variable in addition to the sAppName.
If it ends with the slash, then it will be a folder. (I propose to make it "upp/" by default)
Without the slash this will be a regular prefix. "upp" -> "upptheide", for example.

If I'm shipping several apps, I can put all configs in one folder, and name it after a company, for example.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Thu, 02 Apr 2020 17:33:24 GMT
View Forum Message <> Reply to Message

I just realized that I can create my local .config folder, but I still think that it would be useful to have the sAppNamePrefix variable ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Thu, 02 Apr 2020 18:45:30 GMT
View Forum Message <> Reply to Message

Novo wrote on Thu, 02 April 2020 19:33I just realized that I can create my local .config folder, but I still think that it would be useful to have the sAppNamePrefix variable ...

Maybe we can just allow slash in AppName and default to ".upp/" + GetExeTitle().

Anyway, I would like more opinions on this. After all, I have moved to .config based on suggestions here... :)

Mirek

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Thu, 02 Apr 2020 22:32:20 GMT
View Forum Message <> Reply to Message

mirek wrote on Thu, 02 April 2020 14:45Novo wrote on Thu, 02 April 2020 19:33I just realized that I can create my local .config folder, but I still think that it would be useful to have the sAppNamePrefix variable ...

Maybe we can just allow slash in AppName and default to ".upp/" + GetExeTitle().

Anyway, I would like more opinions on this. After all, I have moved to .config based on suggestions here... :)

Mirek
GetAppName() is used in many places. I haven't check all of them, but this doesn't seem to be a good idea.

Situation with Mac is slightly different. Apps have to be installed into the /Applications, otherwise they won't work correctly, AFAIK.
So, .config has to go somewhere in /Applications/XXX.app/Contents/MacOS/, but in this case ide and umk won't share configuration.
If we want to share stuff, it will go into ~/.config, and this will litter this folder ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Thu, 02 Apr 2020 23:36:30 GMT
View Forum Message <> Reply to Message

BTW, log-files are still stored in ~/.upp/app/app.log ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Fri, 03 Apr 2020 07:23:08 GMT
View Forum Message <> Reply to Message

[quote title=Novo wrote on Fri, 03 April 2020 00:32]mirek wrote on Thu, 02 April 2020 14:45
Situation with Mac is slightly different. Apps have to be installed into the /Applications, otherwise they won't work correctly, AFAIK.

IMO not quite true. As far as I can tell, I am running theide.app from the ~ and everything works just fine (right now, it is using out and .upp)

Mirek

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Fri, 03 Apr 2020 18:57:59 GMT
View Forum Message <> Reply to Message

You are right. I have outdated information.
Soft link in /Applications works as well.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Sat, 04 Apr 2020 13:08:51 GMT
View Forum Message <> Reply to Message

Novo wrote on Thu, 02 April 2020 19:10My ~/.config gets quickly littered by dozens of my dev apps.
Another problem is conflicting names. ~/.config is a system-wide namespace.

I propose to add a sAppNamePrefix variable in addition to the sAppName.
If it ends with the slash, then it will be a folder. (I propose to make it "upp/" by default)
Without the slash this will be a regular prefix. "upp" -> "upptheide", for example.

If I'm shipping several apps, I can put all configs in one folder, and name it after a company, for example.

Done: SetAppGroup, default is "u++". So unless you use SetAppGroup, all is now stored into .config/u++/GetAppName()

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Sat, 04 Apr 2020 13:33:19 GMT
View Forum Message <> Reply to Message

Thanks a lot!

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Sat, 04 Apr 2020 16:46:59 GMT
View Forum Message <> Reply to Message

log-files are still stored in ~/.upp :)
Unsure that they should go into .config, but definitely not into ~/.upp ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Zbych on Sat, 04 Apr 2020 19:06:33 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 04 April 2020 15:08
Done: SetAppGroup, default is "u++". So unless you use SetAppGroup, all is now stored into
.config/u++/GetAppName()

Does it affect only MacOS or Linux apps as well?
And if both, is there a way to get back to old path (~/.upp/AppName)?

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Sat, 04 Apr 2020 22:49:26 GMT
View Forum Message <> Reply to Message

This does affect Mac and Linux apps.
You can call SetConfigDirectory("~/.upp/AppName"). Probably, you also need to
expand/normalize "~/.upp/AppName" first.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Zbych on Sun, 05 Apr 2020 09:12:13 GMT
View Forum Message <> Reply to Message

Novo wrote on Sun, 05 April 2020 00:49This does affect Mac and Linux apps.

What is technical reason of this change?
To be honest I pissed off by changes like this. I have many embedded apps that keep their
settings in .upp directory and the last thing I need is to loose their settings, because default
ConfigFile() has changed.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by amrein on Sun, 05 Apr 2020 09:46:47 GMT
View Forum Message <> Reply to Message

Hi

I don't understand why we use ~/.config and ~/.cache and not directories in the current source
directory when U++ is inside user home directory.
I mean, the main idea of sandboxing is to keep everything in the same directory, right? :(

Note: using ~/.config and ~/.cache is incompatible with deb or rpmbuild packaging if we use umk to build other packages. :?

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Sun, 05 Apr 2020 11:03:17 GMT
View Forum Message <> Reply to Message

Zbych wrote on Sun, 05 April 2020 11:12Novo wrote on Sun, 05 April 2020 00:49This does affect Mac and Linux apps.

What is technical reason of this change?
To be honest I pissed off by changes like this. I have many embedded apps that keep their settings in .upp directory and the last thing I need is to loose their settings, because default ConfigFile() has changed.

It was suggested right in this thread by Posix gurus. I am just trying to deliver...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Sun, 05 Apr 2020 11:07:56 GMT
View Forum Message <> Reply to Message

amrein wrote on Sun, 05 April 2020 11:46Hi

I don't understand why we use ~/.config and ~/.cache and not directories in the current source directory when U++ is inside user home directory.
I mean, the main idea of sandboxing is to keep everything in the same directory, right? :(

You have missed the description of the process. It is quite more complicated then just using ~/.config / ~.cache.

```
String  ConfigFile(const char *file) {
 if(sConfigFolder.GetCount())
  return AppendFileName(sConfigFolder, file);
#if defined(PLATFORM_WIN32)
 if(sHomecfg) {
  String p = GetHomeDirFile(GetAppName());
  ONCELOCK
   RealizeDirectory(p);
  return AppendFileName(p, file);
 }
```

```
 return GetExeDirFile(file);
#elif defined(PLATFORM_POSIX)
 static String cfgdir;
 ONCELOCK {
  String h = GetExeFolder();
  if(!sHomecfg)
   while(h.GetCount() > 1 && DirectoryExists(h)) {
    String pp = AppendFileName(h, ".config");
    FindFile ff(pp);
    if(ff && ff.IsFolder() && ff.CanWrite()) {
     cfgdir = pp;
     break;
    }
    h = GetFileFolder(h);
   }
  if(IsNull(cfgdir))
   cfgdir = GetEnv("XDG_CONFIG_HOME");
  if(IsNull(cfgdir) || !DirectoryExists(cfgdir))
   cfgdir = GetHomeDirFile(".config");
  if(sConfigGroup.GetCount())
   cfgdir = AppendFileName(cfgdir, sConfigGroup);
 }
 String pp = AppendFileName(cfgdir, GetAppName());
 RealizeDirectory(pp);
 return AppendFileName(pp, file);
#else
 NEVER();
 return GetExeDirFile(file);
#endif//PLATFORM
}
```

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Sun, 05 Apr 2020 11:10:10 GMT
View Forum Message <> Reply to Message

Zbych wrote on Sun, 05 April 2020 11:12Novo wrote on Sun, 05 April 2020 00:49This does affect
Mac and Linux apps.

What is technical reason of this change?
To be honest I pissed off by changes like this. I have many embedded apps that keep their
settings in .upp directory and the last thing I need is to loose their settings, because default
ConfigFile() has changed.

Thinking about it... I can actually change ConfigFile further so that it will use ~/.upp config if it finds

it instead of ~/.config.

Do we want that?

Mirek

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by dolik.rce on Sun, 05 Apr 2020 11:22:03 GMT
View Forum Message <> Reply to Message

mirek wrote on Sun, 05 April 2020 13:10Zbych wrote on Sun, 05 April 2020 11:12Novo wrote on Sun, 05 April 2020 00:49This does affect Mac and Linux apps.

What is technical reason of this change?
To be honest I pissed off by changes like this. I have many embedded apps that keep their settings in .upp directory and the last thing I need is to loose their settings, because default ConfigFile() has changed.

Thinking about it... I can actually change ConfigFile further so that it will use ~/.upp config if it finds it instead of ~/.config.

Do we want that?

Mirek
Well, it would solve the backwards compatibility issue, quite smoothly. Only new apps or new installations of existing apps would start to use the old directory. Possible alternative would be to make the code migrate the files from ~/.upp to ~/.config, but that seems quite invasive and could result in some broken apps, e.g. when both new and older version would be run on same machine.

Honza

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Mon, 06 Apr 2020 04:33:41 GMT
View Forum Message <> Reply to Message

Ide is still opening log-file from ~/.upp :(
Although it is stored in .config/u++ now ...

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Mon, 06 Apr 2020 08:12:14 GMT

Novo wrote on Mon, 06 April 2020 06:33Ide is still opening log-file from ~/.upp :(
Although it is stored in .config/u++ now ...

Have you recompiled it?

Mirek

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Mon, 06 Apr 2020 13:31:00 GMT

mirek wrote on Mon, 06 April 2020 04:12Novo wrote on Mon, 06 April 2020 06:33Ide is still opening log-file from ~/.upp :(
Although it is stored in .config/u++ now ...

Have you recompiled it?

Mirek
Double-checked. Full rebuild. Alt-l opens a log-file located in ~/.upp
GIT rev. 74249e7
It is easy to check. Timestamp is "04.04.2020 15:30:37"

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Tue, 07 Apr 2020 03:34:05 GMT

Another problem with Mac.
It looks like GetExeFolder() doesn't resolve links on Mac.
The way I check that: I created a link to umk in ~/.local/bin
~/.local/bin/umk -> ~/dvlp/cpp/upp/out/uppsrc/CLANG.Blitz.Shared/umk
Config dir is located in ~/dvlp/cpp/upp/
This perfectly works on Linux but Mac.
On Mac I need to create a link to config dir in ~/.local/
I guess the problem is with GetExeFolder() -> GetExeFilePath() ...
Not sure who is supposed to resolve links, OS or Upp ...
Debugger doesn't work on Mac, so, it is a little bit difficult to track an origin of the problem.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Tue, 07 Apr 2020 12:36:06 GMT

Novo wrote on Mon, 06 April 2020 15:31mirek wrote on Mon, 06 April 2020 04:12Novo wrote on

Mon, 06 April 2020 06:33Ide is still opening log-file from ~/.upp :(
Although it is stored in .config/u++ now ...

Have you recompiled it?

Mirek
Double-checked. Full rebuild. Alt-l opens a log-file located in ~/.upp
GIT rev. 74249e7
It is easy to check. Timestamp is "04.04.2020 15:30:37"

Bug in theide, should be now fixed.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Tue, 07 Apr 2020 13:48:20 GMT
View Forum Message <> Reply to Message

mirek wrote on Tue, 07 April 2020 08:36Bug in theide, should be now fixed.
Thanks. It is fixed now.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Tue, 07 Apr 2020 14:28:22 GMT
View Forum Message <> Reply to Message

Zbych wrote on Sun, 05 April 2020 11:12Novo wrote on Sun, 05 April 2020 00:49This does affect
Mac and Linux apps.

What is technical reason of this change?
To be honest I pissed off by changes like this. I have many embedded apps that keep their
settings in .upp directory and the last thing I need is to loose their settings, because default
ConfigFile() has changed.

OK sorry again. To make the deal sweeter, I have now adopted dolik's suggestion and settings
get "migrated" (really copied) from .upp to .config/u++ as long as configuration in .upp exists and
in .config does not.

Mirek

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Fri, 10 Apr 2020 15:34:50 GMT
View Forum Message <> Reply to Message

Novo wrote on Tue, 07 April 2020 05:34Another problem with Mac.

It looks like GetExeFolder() doesn't resolve links on Mac.
The way I check that: I created a link to umk in ~/.local/bin
~/.local/bin/umk -> ~/dvlp/cpp/upp/out/uppsrc/CLANG.Blitz.Shared/umk
Config dir is located in ~/dvlp/cpp/upp/
This perfectly works on Linux but Mac.
On Mac I need to create a link to config dir in ~/.local/
I guess the problem is with GetExeFolder() -> GetExeFilePath() ...
Not sure who is supposed to resolve links, OS or Upp ...
Debugger doesn't work on Mac, so, it is a little bit difficult to track an origin of the problem.

Hopefully fixed in the trunk. Looks like BSD magic did not really work in MacOS... :)

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Fri, 10 Apr 2020 16:02:17 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 10 April 2020 11:34
Hopefully fixed in the trunk. Looks like BSD magic did not really work in MacOS... :)
Thanks a lot! It works now.

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Fri, 10 Apr 2020 16:06:25 GMT
View Forum Message <> Reply to Message

#include <mach-o/dyld.h>
This won't compile outside of Mac ... :)

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by mirek on Fri, 10 Apr 2020 16:37:16 GMT
View Forum Message <> Reply to Message

Ops, thanks, hopefully fixed (but please retest that GetExeFilePath fix still works...)

Subject: Re: Things we want from Linux/FreeBSD release archives
Posted by Novo on Fri, 10 Apr 2020 17:16:02 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 10 April 2020 12:37Ops, thanks, hopefully fixed (but please retest that GetExeFilePath fix still works...)
Thanks!
It works on Mac and compiles on Linux. Full rebuild ...