
Subject: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [MrSarup](#) on Sun, 25 Dec 2016 10:19:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Mirek,

Mirek mentioned in the following thread here:

http://www.ultimatepp.org/forums/index.php?t=msg&th=9817&goto=47171&#msg_47171
mirek wrote on Sun, 25 December 2016 09:52

Personally, I would prefer removing .spec file altogether.

.rpm based distros are Spanish village for me. IMO, our responsibility for nightly builds / releases should stop at producing tarball with makefile which builds (perhaps after some package installs) on any most targets. Platform specific packages should be out of ultimatepp.org responsibility... There are too many target platforms to maintain directly...

Mirek

With your answer, you did surprise me a lot. To be able to make a good package requires placing the files in their appropriate directories on Linux platform. Many irresponsible developers do not read the following on Filesystem Hierarchy Standard and create chaos on each platform and system:

https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard

"Making a good package" also means compiling the source code against the right versions of libraries.

Whenever a different source code is released, this may require different library. "Making a good package" means this step is included in there.

Further, it assures that things will work because `_ONE_HAS_TO_DECLARE` dependencies with the right package names in that binary. It will also include menu entries, MIME handlers, documentation, startup scripts, etc. as well.

By providing a spec file, you also allow users to obtain all facilities of rpm with a difference that you did not build them yourself and compel this dirty work to be done by an user. That's fine. But the moment you declare that the spec file belongs to garbage, then many users may have problems.

Thus, I come up with a suggestion to provide cross-distribution of packages of UPP by using FPM technology. This is available here:

<https://github.com/jordansissel/fpm>

There is a good guide here:

<https://www.digitalocean.com/community/tutorials/how-to-use-fpm-to-easily-create-packages-in-multiple-formats>

If the IDE is cross-platform, why not packages for distributing its source code with cross-platform quality?

[EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++ (total votes: 5)

Yes, I agree 1/(20%)

No, it does not matter 1/(20%)

Perhaps it may be helpful 3/(60%)

Subject: Re: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [dolik.rce](#) on Sun, 25 Dec 2016 13:24:06 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi MrSarup!

MrSarup wrote on Sun, 25 December 2016 11:19: Hi Mirek,

Mirek mentioned in the following thread here:

[http://www.ultimatepp.org/forums/index.php?t=msg&th=9817 &goto=47171&#msg_47171](http://www.ultimatepp.org/forums/index.php?t=msg&th=9817&goto=47171&#msg_47171)
mirek wrote on Sun, 25 December 2016 09:52

Personally, I would prefer removing .spec file altogether.

.rpm based distros are Spanish village for me. IMO, our responsibility for nightly builds / releases should stop at producing tarball with makefile which builds (perhaps after some package installs) on any most targets. Platform specific packages should be out of ultimatepp.org responsibility... There are too many target platforms to maintain directly...

Mirek

I'll definitely side with Mirek on this. The developer and the packager shouldn't be the same person, as they actually need very different skillset. The developer needs deep knowledge about the software he writes, while the packager needs to know the platform he packages for. Also, each platform should have it's own packager, one person can not reasonably cover all of them - and I know what I'm talking about, I have been maintaining the packages for Debian based distributions and for Arch Linux for years, even tried to put together a .spec file for Fedora, CentOS and Scientific Linux (without much success, because I don't know those distros well enough).

In your poll above, the options don't make much sense to me. The correct answer for me would be "No, the source package is enough, the rest is up to packagers". If you are familiar with the RPM based distributions, it would be great if you could maintain packages for them.

Best regards,
Honza

Subject: Re: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [MrSarup](#) on Sun, 25 Dec 2016 13:44:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Honza,

After learning Fortran/Cobol on a 5 1/2 megabytes disk, I have seen number of developers today as well as the world of computers and internet growing! Amongst others, I have also seen young and intelligent developers as well as very adamant, conservative and narrow-minded developers.

The suggestion made by me is not actually from me but comes from academics. Most professors would teach this. You will find number of articles and books which preaches this.

U++ is something extremely special. Look at some other parallel works. I say this, because I liked U++ it.

However, it is not possible for EVERYONE to like such a smart project "at every costs". It takes such a lot of time in starting to learn and simply getting into the project.

Exactly this is the major draw back of every project, if this aspect is neglected.

Mind you, I do not care a damn if the developers accept my suggestion or not. This suggestion is merely in the interest of others, who may not need to follow the path I need, or was compelled to take.

About maintaining: I want to see first how far I can come up with U++. It is just the question of time to decide when I need to leave U++ project or how long should I stick with it.

While you may have an excellent experience, you may then need to appreciate such a master piece of work available in the last few years developed with FPM.

Especially from your experience, I would want to hear from you, or people like you, that FPM is one of the best projects seen in this area.

Or did you just reject and kept on making arguments to support Mirek's position? Did you understand the concept of FPM at all?

FPM does once exactly what you do many times on many platform.

So if you are maintaining many distros, why do you not yourself use it first and let us know your experience? Why prepare many binaries for many platforms, when one is sufficient?

Subject: Re: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [amrein](#) on Sun, 25 Dec 2016 18:42:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

A spec file for Fedora and other rpm based distribution is already provided and doesn't need much attention.

The bigger question about upp.spec file is how to handle ultimate++ version without using variables.

At present, to compile a new snapshot tarball, you need to run something like this:

```
source_file=upp-x11-src-10577M.tar.gz
release=${source_file%.tar.gz}
release=${release#*upp-x11-src-}
date=$(LC_TIME=en date '+%a %b %d %Y')
rpmbuild -tb --define "version $release" --define "date $date" ${source_file}
```

Most people using src.rpm are used to simply run this kind of command:

```
rpmbuild --rebuild upp-10577M-1.src.rpm
```

And so if someone wants to ease rpm build for everyone, he will need to make upp.spec less generic for each upp source snapshot.

How? Using templates I guess.

Those commands are enough to create a snapshot spec from the current upp.spec (without adding release history):

```
release=10577M
date=$(LC_TIME=en date '+%a %b %d %Y')
sed -e 's|/%version|'$release'|g' -e 's|/%date|'$date'|g' upp.spec > snapshot-upp.spec
```

Any upp tarball containing this snapshot-upp.spec file would be able to be compiled with a simple command "rpmbuild -tb upp-x11-src-\$(release).tar.gz".

You can even build a src.rpm from it like this "rpmbuild -ts upp-x11-src-\$(release).tar.gz" and compile the resulting src.rpm as previously said "rpmbuild --rebuild upp-\$(release)-1.src.rpm"

Subject: Re: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [dolik.rce](#) on Sun, 25 Dec 2016 19:38:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

MrSarup wrote on Sun, 25 December 2016 14:44: Especially from your experience, I would want to hear from you, or people like you, that FPM is one of the best projects seen in this area.

Or did you just reject and kept on making arguments to support Mirek's position? Did you understand the concept of FPM at all?

FPM does once exactly what you do many times on many platform.

So if you are maintaining many distros, why do you not yourself use it first and let us know your experience? Why prepare many binaries for many platforms, when one is sufficient?

Seems I was reading your post bit too fast and reacting to something slightly different then what you wrote. I haven't checked the links you posted and mixed up FPM with Flatpak :blush:. Sorry for the confusion this might have caused.

Anyway, my point still holds, the packaging can and should be separated from the development. FPM (and actually even Flatpak :)) is just another platform. You are right that it comes for lower price, since the effort is not repeated for multiple distributions. I haven't actually heard about FPM before, but I will definitely check it - it might be worth a try.

It is also worth noting that the packaging itself is not the only thing to consider. The packaging infrastructure is also important, e.g. for Ubuntu, the packages can be built and hosted for free on launchpad, for OpenSUSE (and many others) there is OBS. This can be a great simplification, especially for small projects. Of course, if this approach would integrate with FPM, it would be even better. Can tell if that is possible right now, since I haven't read enough about FPM yet.

tl;dr version: I'll try to check out the FPM and perhaps try to implement U++ packages with it. I think your suggestion is definitely valid and worth investigation. But I also believe that it is not Mireks work or responsibility.

Honza

Subject: Re: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [MrSarup](#) on Sun, 25 Dec 2016 20:21:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Honza,

dolik.rce wrote on Sun, 25 December 2016 20:38

Seems I was reading your post bit too fast and reacting to something slightly different then what you wrote.

...

But I also believe that it is not Mireks work or responsibility.

Honza

Confusion between things has no impact. If you actually meant it, that FPM is something useless or unwanted or foolish, then I would really doubt your professional competence. Good that you made this clear.

No one told Mirek that he should make 10.800+ posts here. Why did he do so? Honza, no one commanded you to make 1.700+ post here. Why did you do that? Who paid the developers to create U++ and invest a gigantic time phase of their life for the benefit of the U++ community? Why did they do that?

C++ is there, extremely matured. What was the reason to create a new forking movement of U++? What are we talking about then? Are we talking about having something better, or at least something different?

U++ is different and has its own personality and identity.

In the first hour, I was able to compile a GUI on windows that had some basis of examples and references. I combined, played with it and could make some fun coding.

What is the aim of this movement? What is the difference U++ developers are trying to make in the world of programming then? To me it looks like one wants to make things not only different but also simpler, easier and efficient. That's U++. Will someone disagree on this?

Who made laws and who made decisions to what is or what not to be? Who said that the contribution `_MUST_` end when Mirek makes an ugly tarball for distribution? Did someone think that it would be in the interest of new comers if the entire installation routines are made much easier and simpler, keeping the cross-platform in front of our eyes.

Then, the U++ community will grow faster and it becomes must popular.

Thus, we are then talking about U++ community politics, where members participates and distributes work.

Not making packages is a decision that challenges the entire Linux community, where packages and modules built with binaries is one of the predominant source of building applications, services and servers. Denying to recognize, that packaging concept is a piece of garbage, shows ignorance and stupidity.

The developers need to firstly define the aim of U++ and always keep it before their eyes: Easy, simple and efficient. This applies at all levels.

Linux means headaches.

I remember days in 1996, when I sat crying before my half dead server in the data center trying to install Redhat baby and make it work. What happened today? Where is Redhat now and how comfortable Linux became?

It is a rubbish idea, if one leaves a fragment of a process difficult while keeping the rest as easy as possible.

To me, U++ installation must be as easy as everything. This includes Linux installs. This applies to docs as well. Once this foundation is laid upon, many things becomes simpler.

Subject: Re: [EFFING PACKAGE MANAGEMENT] Provide cross-distribution of packages of U++

Posted by [mirek](#) on Wed, 28 Dec 2016 16:03:04 GMT

[View Forum Message](#) <> [Reply to Message](#)

MrSarup wrote on Sun, 25 December 2016 14:44

After learning Fortran/Cobol on a 5 1/2 megabytes disk, I have seen number of developers today as well as the world of computers and internet growing! Amongst others, I have also seen young and intelligent developers as well as very adamant, conservative and narrow-minded developers.

Just for the record, Fortran/Cobol with 5 1/2 megabytes of disk, you must be younger than me with much more comfortable beginnings. I was learning on Z80 with 48KB of RAM, tape recorder and selfwritten assembler...

Quote:

FPM does once exactly what you do many times on many platform.

Great. Anybody to do this for U++?

Mirek
