

---

Subject: MPL

Posted by [mirek](#) on Fri, 15 Aug 2008 20:46:31 GMT

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

---

Speaking about licensing...

As one of things that is seriously missing from U++ is a good vector graphics support and one possible way how to fix this is Cairo, I wonder how much MPL is really permissive...

From my reading, it appears as you are not obliged to release sources of rest of your application, even if you combine Cairo code with anything else. And you have to release sources of Cairo only if you make changes to it. Not advertising clauses either.

That is sufficiently permissive IMO. Am I correct?

Mirek

---

---

Subject: Re: MPL

Posted by [kodos](#) on Fri, 15 Aug 2008 21:39:31 GMT

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

---

Speaking of cairo: I am currently working on the integration in U++ . I'm not finished yet but I have a CairoCtrl that works on Windows and a compatibility layer to draw UPP::Drawings with cairo. If there is interest I can post it in bazaar. But there are a few changes needed in the png package (i had to remove some compiler flags).

---

---

Subject: Re: MPL

Posted by [cbpporter](#) on Fri, 15 Aug 2008 21:52:51 GMT

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

---

Great, you have to be a lawyer to make heads or tails out of this licensing issue. I hope we are not getting over our heads with mixed licenses, using BSD, MPL and what not. I think we should ask some help from somebody who has experience in deciphering license descriptions.

As for Cairo, I believe it is a good idea. AGG 2.4 would be great from the license point of view, but working with AGG is IMO extremely hard. You need a lot of code to get basic stuff done, and it is not very intuitive. On the other hand Cairo is easy to use.

I wonder how MPL clauses apply if we keep Cairo sources intact, but change the build method to produce static linking? Having self-contained executable is one of U++ advantages. If the license is acceptable and we can get static linking, and if the performance is good for basic operations we could replace the current platform dependent draw primitives with Cairo calls, with the added advantage of getting Mac drawing for free. This could make Mac port easier, but we still need Objective C for windowing. And AFAIK, BeOS too! A Haiku port is not planned, but it is good to

know that it is technically possible.

---

---

Subject: Re: MPL

Posted by [emr84](#) on Sat, 16 Aug 2008 00:18:31 GMT

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

---

I believe that this book can be a good aid to understand about different licenses:

<http://oreilly.com/catalog/osfreesoft/book/>

I hope that it is useful (it seems that the MPL can cause problems)

---

---

Subject: Re: MPL

Posted by [mirek](#) on Sat, 16 Aug 2008 07:39:47 GMT

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

---

cbpporter wrote on Fri, 15 August 2008 17:52

I wonder how MPL clauses apply if we keep Cairo sources intact, but change the build method to produce static linking?

Actually, if it is cairo, we will not leave sources intact in any case

OTOH, I have not seen anything about static linking, just combined work. And in that case you are about to reveal the modified code of cairo only. Plus only WE need to do that, U++ user will get already modified cairo, with sources available, so there is nothing he has to worry about.

At least, that is my understanding.

Quote:

Having self-contained executable is one of U++ advantages. If the license is acceptable and we can get static linking, and if the performance is good for basic operations we could replace the current platform dependent draw primitives with Cairo calls, with the added advantage of getting Mac drawing for free.

I do not think this is really possible. There still will have to be platform dependent draw for performance reasons.

Cairo, AFAIK, is basically a software renderer. Which, BTW, is exactly what Vista or Mac OS X do. OTOH, the most basic rendering operations ("DrawRect, DrawText, DrawImage") tend to be very well accelerated by HW, at least in Linux.

Quote:

This could make Mac port easier, but we still need Objective C for windowing.

Yeah, that sucks, does not it?

Mirek

---

---

Subject: Re: MPL

Posted by [mirek](#) on Sat, 16 Aug 2008 08:53:33 GMT

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

---

kodos wrote on Fri, 15 August 2008 17:39 Speaking of cairo: I am currently working on the integration in U++ . I'm not finished yet but I have a CairoCtrl that works on Windows and a compatibility layer to draw UPP::Drawings with cairo. If there is interest I can post it in bazaar. But there are a few changes needed in the png package (i had to remove some compiler flags).

Have you succeeded in "plugining" the Cairo? I mean, do you have it as source package?

Mirek

---

---

Subject: Re: MPL

Posted by [cas\\_](#) on Sat, 16 Aug 2008 10:57:35 GMT

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

---

luzr wrote on Sat, 16 August 2008 09:39

OTOH, I have not seen anything about static linking, just combined work. And in that case you are about to reveal the modified code of cairo only.

MPL FAQ seems to confirm that your understanding is correct

---

---

Subject: Re: MPL

Posted by [kodos](#) on Sat, 16 Aug 2008 10:58:51 GMT

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

---

luzr wrote on Sat, 16 August 2008 10:53

Have you succeeded in "plugining" the Cairo? I mean, do you have it as source package?

Mirek

Yes I have it as source package. I had to rename some

`#include <>` to `#include ""`

and I add some dummy headers that link cairo with the included png and zlib .

---

---

Subject: Re: MPL

Posted by [amrein](#) on Sat, 16 Aug 2008 11:54:39 GMT

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

---

MPL licence is like LGPL + an exception to be able to link statically with proprietary software + no patent grant to you or your users if you use a modified version of the software. If you modify the MPL covered code, you must provide the modified MPL source in a working form with all additions. The final user can ask the distributor for the MPL licensed source code.

A module covered by the GPL and a module covered by the MPL cannot legally be linked together.

Cairo source can also be under LGPL. U++ is available with its source code so no problem but:

- With static linking, if someone else use your software + cairo under LGPL, the final user can ask the distributor for the cairo source code (modified or not) + your source. Your source must be LGPL or GPL. If the final distributor link statically to your software, he must release the wall source using GPL or LGPL. With dynamic linking, the final distributor will just have to add in its docs, apps, about menu, ... that his software use LGPL licensed code and must provide the source of the LGPLed code with its software (in the media!).
- If they link statically with U++ and U++ link dynamically with the cairo LGPL covered source, the distributor can use a proprietary licence for their own software but still need to tell about the LGPL licence of the dynamic library.

Quote:Great, you have to be a lawyer to make heads or tails out of this licensing issue.

Sure!

Most of the FOSS licences let you include other source code covered by the same licence into your own source. When you select an open source licence, you should carefully study the advantages, drawbacks and risks. With a not enough spread one, you will have to reinvent the wheel each time you want a new feature. The more licence we have, the more fragmented the community is, the more difficult producing new FOSS become.

Really, before any work on including other libraries, can we have a definitive statement about the final U++ licence choice?

---

---

Subject: Re: MPL

Posted by [mr\\_ped](#) on Sat, 16 Aug 2008 21:20:41 GMT

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

---

Quote:A module covered by the GPL and a module covered by the MPL cannot legally be linked together.

GPL is of no problem/interest for U++, this is BSD-like (or LGPL + exception for static linking) type of project, so actually MPL sounds good for U++.  
(As far as I'm aware there's no GPL code in U++ installer?)

---

---

Subject: Re: MPL  
Posted by [mirek](#) on Sat, 16 Aug 2008 23:35:32 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

amrein wrote on Sat, 16 August 2008 07:54  
Really, before any work on including other libraries, can we have a definitive statement about the final U++ licence choice?

Is not it a bit fast, after 4 years with "BSD-like" license?

Anyway, I believe MIT is the most reasonable choice. Alternatively, 2 clause BSD (which IS GPL compatible) or just public domain (SQLite seems to do pretty well

Mirek

---

---

Subject: Re: MPL  
Posted by [amrein](#) on Sun, 17 Aug 2008 20:51:45 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Why not create a poll with different license:  
MIT, modified BSD, MPL, LGPL, GPL (with direct links to OSI web page for each of them)?

---

---

Subject: Re: MPL  
Posted by [mirek](#) on Sun, 17 Aug 2008 21:51:53 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, do it.

Mirek

---

---

Subject: Re: MPL  
Posted by [amrein](#) on Mon, 18 Aug 2008 07:03:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Done

<http://www.ultimatepp.org/forum/index.php?t=msg&th=3730&start=0&>

---