Subject: Release 11.10
Posted by mirek on Fri, 30 Sep 2011 13:10:37 GMT
View Forum Message <> Reply to Message

Hi,

it was some time since the last "official" release, partly because nightly builds and svn mirror with policy of "stable trunk" serve community needs good enough.

With such slowed down release frequence, I believe it might be a good time to get back to "season number" releases and I guess it is a good time now to release 11.10.

Now the question is what is missing. I guess we should fix to bugfixing mode and resolve some release policy issues.

One question is what to do with rainbow. It is a great addition to U++ (I hope), but it has little practical uses for release users.

Any idea? Anything else?

Mirek

Subject: Re: Release 11.10
Posted by cbpporter on Fri, 30 Sep 2011 13:41:56 GMT
View Forum Message <> Reply to Message

Quite frankly, I do not know what one could do with it either. I played around with it trying to see if Irrlicht integration would be possible (too much work to fit in in my release schedule and Irrlicht lacks almost all drawing primitives) and it seems interesting to tinkerers, but it is not generally useful to end users. I see it more as a gateway to new platforms where U++ can provide what it did on the old platforms to the end users. WinFB is extremely cute though!

But the OpenGL backend could be something interesting for a larger pool of people. It is hard to provide a GUI inside a hardware accelerated area. I should know, I had to hack together a few.

As for what to add to U++, it is fairly complete from my point of view. It has been ages since I found a new bug I did not know about. Maybe do something with that half of a solution I proposed for IPv6.

Subject: Re: Release 11.10
Posted by Sender Ghost on Sat, 01 Oct 2011 02:01:29 GMT
View Forum Message <> Reply to Message

Hello, Mirek.

mirek wrote on Fri, 30 September 2011 15:10
Now the question is what is missing. I guess we should fix to bugfixing mode and resolve some release policy issues.

I hope, you will review latest redmine (#62) patch for OWC builder about correct support for .rc files, before new release.
mirek wrote on Fri, 30 September 2011 15:10
One question is what to do with rainbow. It is a great addition to U++ (I hope), but it has little practical uses for release users.

There are options:
1) Creation of rainbow assembly, along with bazaar, examples, reference, tutorial, uppsrc, etc.
2.1) Creation of Rainbow package inside rainbow assembly, which consist of various rainbow platforms:

description "Rainbow GUI platforms\377";

uses(SKELETON) Skeleton;

uses(WINALT) WinAlt;

uses(WINFB) WinFb;

uses(LINUXFB) LinuxFb;

uses(SDLFB) SDLFb;

uses(WINGL) WinGl;

file
 Rainbow.cpp;

2.2) Creation of RainbowApps assembly (new or mirror of MyApps assembly with rainbow directory), where developer could add Rainbow package for his|her applications and appropriate build flags.
3) Automatic recognition for "RAINBOW <RAINBOW_PLATFORM>" (or just "RAINBOW_<PLATFORM>") build flags inside TheIDE to add appropriate packages for GUI application. But this could be ambiguous.

I vote for 1 and 2 option.

Edit 2011-10-03:
Added rainbow changed files and diff for current 3947 revision, to clarify what I meant.

File Attachments

1) rainbow_r3947.zip, downloaded 604 times

Subject: Re: Release 11.10
Posted by kohait00 on Mon, 03 Oct 2011 17:03:07 GMT
View Forum Message <> Reply to Message

i'd vote for 2..

in such a way, the user has also a practical example on how to properly manage a (custom) set of backends..and how it works.

Subject: Re: Release 11.10
Posted by Sender Ghost on Wed, 05 Oct 2011 07:39:21 GMT
View Forum Message <> Reply to Message

Creation of Rainbow package templates is also possible.
You can find changed CtrlLib .upt files for Rainbow package in the attachments.

File Attachments
1) Rainbow_package_with_templates.zip, downloaded 598 times

Subject: Re: Release 11.10
Posted by mirek on Wed, 05 Oct 2011 14:59:48 GMT
View Forum Message <> Reply to Message

w.r.t. Rainbow, well, maybe the correct solution is to move WinFB to reference, as it is a reference example about how to use rainbow after all.

It would add just two or three packages to reference, which is OK IMO (Framebuffer, WinFB and some example, perhaps UWord)

Subject: Re: Release 11.10
Posted by Sender Ghost on Thu, 06 Oct 2011 05:08:12 GMT
View Forum Message <> Reply to Message

mirek wrote on Wed, 05 October 2011 16:59w.r.t. Rainbow, well, maybe the correct solution is to move WinFB to reference, as it is a reference example about how to use rainbow after all.

It would add just two or three packages to reference, which is OK IMO (Framebuffer, WinFB and some example, perhaps UWord)
Ok, now I understand what you mean - size matters (for release).
Sure, complete reference examples will show how to use selected Rainbow backends for applications.
But I don't think that reference assembly is right place for Framebuffer package on long term. It could be placed into uppsrc directory, in this case.

Subject: Re: Release 11.10
Posted by unodgs on Thu, 06 Oct 2011 05:42:06 GMT
View Forum Message <> Reply to Message

mirek wrote on Wed, 05 October 2011 10:59w.r.t. Rainbow, well, maybe the correct solution is to move WinFB to reference, as it is a reference example about how to use rainbow after all.

It would add just two or three packages to reference, which is OK IMO (Framebuffer, WinFB and some example, perhaps UWord)
I think this is wrong. Framebuffer (and winfb + linuxfb) is a software renderer backend for upp's painting engine and it's place is in the Rainbow package. I think that win32 and x11 renderers should be detached from ctrlcore and moved to Rainbow package as well. My point is that all available renderers should be in one place. In examples should be only classic UWord and in theide one chooses a proper flag (maybe in a separate drop list to make it easier) for rendering engine (default is GUI).

Subject: Re: Release 11.10
Posted by mirek on Thu, 06 Oct 2011 08:44:07 GMT
View Forum Message <> Reply to Message

unodgs wrote on Thu, 06 October 2011 01:42mirek wrote on Wed, 05 October 2011 10:59w.r.t. Rainbow, well, maybe the correct solution is to move WinFB to reference, as it is a reference example about how to use rainbow after all.

It would add just two or three packages to reference, which is OK IMO (Framebuffer, WinFB and some example, perhaps UWord)
I think this is wrong. Framebuffer (and winfb + linuxfb) is a software renderer backend for upp's painting engine and it's place is in the Rainbow package. I think that win32 and x11 renderers should be detached from ctrlcore and moved to Rainbow package as well. My point is that all available renderers should be in one place. In examples should be only classic UWord and in theide one chooses a proper flag (maybe in a separate drop list to make it easier) for rendering engine (default is GUI).

I agree that Framebuffer should, in long term, be moved to uppsrc. Putting it to reference is just the solution for the next release.

I also agree that in theory, there should be some rainbow subdirectory of uppsrc. But I vote against putting win32/x11 outside CtrlCore - simply for practical reasons...

Mirek

Subject: Re: Release 11.10
Posted by kohait00 on Sat, 08 Oct 2011 19:32:25 GMT
View Forum Message <> Reply to Message

what about leaving x11/win32 backend in uppsrc, but placing it in a seperate package there,

which gets selected by default...

this would still isolate the backend stuff for itself (in the same manner as in rainbow), but leave it in the place where it as used per default belongs to (according to mirek)..

moving Framebuffer in uppsrc wouldnt make much sense for a simple reason..as standalone it doesnt serve much..it always needs at least one final FB backend to compile..

putting it together i'd say:

* rainbow is the destination for all backend packages..no examples of how to use it..
* default arch drawing backends can be 'moved' to uppsrc and selected as per default, for simplicity reasons
* WinFb is an important development backend for Fb emulation during design phase.. just as SDLFb, leave it in rainbow, together with Framebuffer
* use examples on rainbow should go to examples or reference..we should mayba append a suffix to those examples or make a special subassembly under reference/examples.

Subject: Re: Release 11.10
Posted by mirek on Sat, 08 Oct 2011 19:47:18 GMT
View Forum Message <> Reply to Message

kohait00 wrote on Sat, 08 October 2011 15:32what about leaving x11/win32 backend in uppsrc, but placing it in a seperate package there, which gets selected by default...

Correct, but impractical. It would add two packages two packages to any application - more packages -> less overview.

Quote:
moving Framebuffer in uppsrc wouldnt make much sense for a simple reason..as standalone it doesnt serve much..it always needs at least one final FB backend to compile..

So what? The only reason to keep Framebuffer from uppsrc is because it is not mature enough IMO.

In every case, I am now only resolving the current release - there I would go reference path. In future we can decode otherwise.

Mirek
