Subject: upp-cocoa CtrlCore skeleton package to play
Posted by fudadmin on Mon, 24 Oct 2011 03:24:24 GMT
View Forum Message <> Reply to Message

due to popular requests and long overdue:
upp CtrlCore for Cocoa. Don't become too excited because its not much in there but just a lot of my time (It should be call blooddy cocoa, if you can guess what I mean). but i would have been over the moon if I had it one year ago.
from my notes inside the package:

Quote:
install this package into your MyApps.
or create separate
or can go it to the bazaar?
but then we are going to mess up the bazaar with a lot of cocoa tests?

Cocoa status and TODO:
for mirek:
1 Core IsNan problem as described on forums 3 times

2 png, z, freetype Link options for (if symbols not found...)
2.1 plugin/png package when COCOA : -lpng
2.2 Core package when COCOA : -lz
2.3 Draw package when COCOA : -lfreetype -lfontconfig
for aris et all: maybe need to fix my ide patches for the above autolinking to work?

3 rethink the whole building flags structure, having in mind, COCOA can run on mac and linux
tmp fix for Core config.h

```
#ifdef flagCOCOA
 #define PLATFORM_COCOA 1
#elif flagOSX11
 #define PLATFORM_OSX11 1  //aris002 why do we need this?
#else
 #define PLATFORM_X11 1
#endif
```

4 in the ide Launch after building - fix Terminal problem
4.1 for COCOA on mac, win and linux
4.2 for X11 on mac
(I have got some bad tmp fixes but, again, need to sort out the flags)
4.3 related to above (VERY IMPORTANT cause now *.app started from ide opens hidden below the ide)
- fix hidden upp menus for appleWM.
- Find how to check window manager name
- apply my hack from forums?

5 Cocoa installation on windows
5.1 minimal guide

5.1 adding libraries (nearly solved - flags again!!!)

6 test existing CtrlCoreCoc on windows! - needs opal and then cairo... hm

Random cocoa programming TODO:
The best case scenario is if someone, who knows cocoa well, joins my efforts... because
7 Image - Timeframe? I really can't imagine - maybe now the most difficult for me
  1 month for it? If i can grasp the ideas from the current upp?
  Got it working with pure cocoa but to connect it with upp
  needs rethinking of the structure of ImageDraw and SystemDraw.
  At the moment, I am trying
  based on win32 and comparing to X11 but maybe this is a wrong approach.
  Also, flipping is a headache...
7.1 test SetSurface and maybe implement those ideas for image? I think it was working correctly
  but I messed up and lost test packages after one of my mac crashes.
  Maybe to get rid of CG and use only NSImage?
8 draw functions - more to implement eg ellipse poly etc - not very difficult
9 keyboard events -I think I was able to catch them but need to integrate with the whole event loop
  and fix some codes (at the moment commented?). OTOH, Using mirek's scan tables maybe we could try
  our own keyboard events translation for apple and cocoa (Insert keycode and other missing keycodes
  problem on mac?)?
10 mouse events - similar model should apply as for keyboard? not explored
11 drawing/painting events - I think I am able to catch them from NSView correctly?
12 application delegate - used to start cocoa menus - find a way to set it directly to upp Ctrl methods?
13 menus - were more or less working but giving headaches by crashing. I guess, due to native threading
  and/or pool management. I removed them for now - should be not difficult.
14 *.app creation from ide - easy but I don't need it until 4.3 is resolved
15 text - I have no idea how difficult
16 timer - I have no idea how difficult

Would be nice to hear from someone who can test (I hope Nick has already bought his mac?) .
Also, I am playing with Cocoa on windows. Anyone has installed or knows how to install opal and/or cairo?

File Attachments

1) upp-cocoa1.zip, downloaded 626 times

Subject: Re: upp-cocoa CtrlCore skeleton package to play with
Posted by fudadmin on Tue, 13 Mar 2012 21:26:08 GMT

View Forum Message <> Reply to Message

As requested by Tom1 in another topic:
updated UNS package. Place eg in upp-cocoa. Should work on gnustep and mac.
Test like this:

```
#include <UNS/UNS.h>

using namespace Upp;

#define LLOG(x) LOG(x)

int main(int argc, const char *argv[])
{
 StdLogSetup(LOG_CERR);
 UppCocAutoReleasePool pool;
 LLOG("starting main");
// UppCocApplication app;
 CocWnd w;
 Rect r = RectC(0,0,300,300);
 CocWnd::CreateApp();
 w.CreateWnd(r);
 CocWnd::RunApp();
 LLOG("after cocoa run");
 return 0;
}
```

EDIT: UNS.zip updated for windows gnustep
To test compiling the flag should be just:
COCOA USEMALLOC

File Attachments

1) UNS.zip, downloaded 522 times

Subject: Re: upp-cocoa CtrlCore skeleton package to play with
Posted by Tom1 on Wed, 14 Mar 2012 09:02:08 GMT
View Forum Message <> Reply to Message

Hi Aris,

Thanks for uploading the UNS package. I'm in trouble with compiling, so I must have done something wrong:

1. I installed the stable GNUstep packages following directions from the GNUstep site. That probably went OK as it seems to work.

2. I checked out yesterday the current upp SVN repository to upp sub directory in my home

directory. Now I downloaded and extracted your UNS package to uppsrc directory.

3. I copied ide.exe (4193) to home/tom/bin and executed it from GNUstep shell. Then I created assembly 'cocoa' and made nests cocoa and uppsrc available to it, in addition to the output directory. I also set up MINGW build method where I had to add an include directory: C:\GNUstep\GNUstep\System\Library\Headers.

4. Then I created a new package 'cocoatest1' and added there dependency to UNS package, which brought in Core, Draw, RichText, plugin/png and plugin/z in addition to UNS.

5. Finally I added main.cpp file containing your sample above.

The result from trying to compile was not so good. Please see the attached compiler log. Any suggestions?

Best regards,

Tom

File Attachments

1) compilerlog.zip, downloaded 552 times

Subject: Re: upp-cocoa CtrlCore skeleton package to play with
Posted by fudadmin on Wed, 14 Mar 2012 13:34:47 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Wed, 14 March 2012 09:02Hi Aris,

Thanks for uploading the UNS package. I'm in trouble with compiling, so I must have done something wrong:

1. I installed the stable GNUstep packages following directions from the GNUstep site. That probably went OK as it seems to work.

2. I checked out yesterday the current upp SVN repository to upp sub directory in my home directory. Now I downloaded and extracted your UNS package to uppsrc directory.

3. I copied ide.exe (4193) to home/tom/bin and executed it from GNUstep shell. Then I created assembly 'cocoa' and made nests cocoa and uppsrc available to it, in addition to the output directory. I also set up MINGW build method where I had to add an include directory: C:\GNUstep\GNUstep\System\Library\Headers.

4. Then I created a new package 'cocoatest1' and added there dependency to UNS package, which brought in Core, Draw, RichText, plugin/png and plugin/z in addition to UNS.

5. Finally I added main.cpp file containing your sample above.

The result from trying to compile was not so good. Please see the attached compiler log. Any suggestions?

Best regards,

Tom

flags problem. see the other topic

Edit: lets keep this topic for things when you can run UNS..
