
Subject: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Sun, 12 Jun 2011 23:15:45 GMT

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

Hi, I'm new to Ultimate++ and a friend asked me to see if I could port it to the Mac, and after some time, I was able to get theide to run.

I used the UNIX download, and I only had to modify the Makefile a little, and fiddle plugin/ndisasm a bit. The result of the build is: `_out/ide.out`

which does run from the command line!
It posts a GUI asking to create a package.

I'm on Mac OS X 10.6.7 using an 2010 iMac computer, which uses an intel 64 bit CPU. MacOS X has an X11 bridge, so all the needed X11 libs and headers are there, and the build apparently works.

I would be grateful for a few suggestions on next steps.

There is a file UPP.SPEC that seems to have installation steps; I'll see if I can install theide in the proper folder, and run some examples or tutorials.

There are other tools that need to be built also besides theide?

I'm happy to share the details, and rebuild with a newer version. I used 3211.

Thanks -

David Remba

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [dolik.rce](#) on Mon, 13 Jun 2011 05:34:12 GMT

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

Hi David!

That's great news! Actually it sounds suspiciously easy

Installing theide is pretty simple on most systems. I am not very familiar with mac, so you might have to supply some knowledge of your own, but it is basically just:

- 1) Moving the ide.out to some place where binary apps are usually stored (so the OS can find it) and rename it to "theide".
- 2) Creating a shortcut/launcher/menu item so user can simply launch the program.

That is the minimal setup

There are no additional tool necessary, you need just theide executable and a compiler (which you already have, since you compiled it). The configuration files etc. should be created when theide first starts. They will be most likely in \$HOME/.upp/theide.

To make theide usable, you will also have to create some assemblies and a build method. Have a look here and here for hints

It would be great if you could make a package for it, so we could upload it on the web!

Best regards and thanks for the efforts,
Honza

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Tue, 14 Jun 2011 03:40:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Honza,

The linux install to ~/upp worked ok,
make install

and as you said a build method is needed,
because the build attempts to link with
libs that aren't there (like GTK).

I did encounter one run-time problem so far,
which is that pull down menus and option menus
don't work.

I will investigate this, and there will be
probably some source code change.

I will keep you posted.

Here are the steps I did in detail:

1. to build theide

```
cd upp-x11-src-3211/uppsrc
```

make

but make these changes to Makefile first:

chgs:

a. in Makefile

added -DflagNOGTK

added -DHAVE_CONFIG_H (for ndisasm)

b. downloaded ndisasm and ran ./configure to get:

config.h (file added)

compiler.h (file replaced)

(these files appear to be generated by ndisasm

'configure' and are specific to the type of

system (hardware and software).

c. uppsrc/plugin/ndisasm/lib:

inttypes.h

LONG_MAX == 9223372036854775807L not working, hardcoded for MacOSX

d. in Makefile

turned-off compile and archive add for snprintf

because MacOSX has this function builtin

e. in Makefile

added to CFLAGS: -std=c99 -pedantic

(from comparing with ndisasm build)

f. in Makefile, final link step changes:

(differences in MacOSX gcc/ld from UNIX)

removed from LDFLAGS: -Wl,--gc-sections

removed -Wl,-O,2 (optimize) -Wl,-s (strip)

was: \$(LINKER) -o \$(OutFile) -Wl,-s \$(LIBPATH) -Wl,-O,2 \$(LDFLAGS) \

removed --start_group after Export.o

was: -Wl,--start-group \

removed --end_group after -lz

was: -Wl,--end-group

and use minimal library list:

-lX11 -lXrender -lz -lXext -lpng -lXft -lfontconfig -lfreetype -lm

(original lib list commented)

after successful build:

2. cp upp-x11-src-3211/uppsrc ../theide

3. doinstall mods

don't rm files for now

run in a terminal shell:

~/theide

note: double-click in finder will not work yet.

this will give:

error: Saturn cannot open files of this type.

4. runtime observations:

- a. pulldown menus & option menus do not appear
- b. errors in build (need to add a build method for MacOSX)

I'm working on these now ...

-Dave

daveremba@yahoo.com

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [dolik.rce](#) on Tue, 14 Jun 2011 05:14:44 GMT

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

daveremba wrote on Tue, 14 June 2011 05:40Hi Honza,

The linux install to ~/upp worked ok,
make install

and as you said a build method is needed,
because the build attempts to link with
libs that aren't there (like GTK).

Adding NOGTK flag to the build method should help in this particular case (as you already found out when building theide)...

daveremba wrote on Tue, 14 June 2011 05:40

I did encounter one run-time problem so far,
which is that pull down menus and option menus
don't work.

I will investigate this, and there will be
probably some source code change.

Similar problem was discussed in this thread some time ago... IIRC we didn't find any fully
satisfying solution, but it should give you the idea where to look.

The instructions are nice and clear, most of it should be easy to solve once and forever by slightly
modifying the sources and/or makefile generator.

Honza

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Wed, 15 Jun 2011 05:56:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Honza,

Thanks for the link to the message thread.
It really helped.

After reading all the tests -
I inserted code from the end of the thread,
and the code at the end worked!
(the whole X windows and window manager
stuff is too complicated!)

theide seems to be fully operational
on MacOSX from what I can tell!
(except for the build method)

In my code version, there is now a
PLATFORM_MACOSX definition,
so whatever worked for Windows/Linux,
etc. should not be affected
by these changes. (and when someone
who really knows X gets to it the
code could be merged better).

Two questions for you:

1. I cannot build apps yet from inside theide,
but I can from the shell, using gcc and the libraries.
HelloWorld worked.

To add a MacOSX build method, do
I modify these files? (to add a MacOSX bm like OSX11)

```
CtrlCore/CtrlCore.upp:26:library(OSX11) "X11 Xft fontconfig Xrender freetype expat";  
Draw/Draw.upp:19:library(OSX11) "fontconfig freetype expat";
```

and do I also modify file GCC.bm?

FYI -
the command line DEBUG mode build
for testing helloworld was:

compile:

```
c++ -c -x c++ -g3 -O0 -ffunction-sections -fdata-sections -I./ -I../uppsrc \  
-I/usr/X11R6/include -I/usr/X11R6/include/freetype2 \  
-DflagGUI -DflagGCC -DflagLINUX -DflagPOSIX -DflagNOGTK -DflagDEBUG \  
hello.cpp -o hello.o
```

link:

```
c++ -o hello hello.o \  
... list of U++ .a files ...  
-L"/usr/X11R6/lib" -L"/usr/lib" \  
-lX11 -lXrender -lz -lXext -lpng -lXft -lfontconfig -lfreetype -lm
```

2. Since I'm a newbie to this project, I'd prefer not to touch the SVN/GIT repository, so maybe you could please assist, if I give you a zip with the changed files? I marked each source file with "zv" where-ever there is a change.

Best,

Dave

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Wed, 15 Jun 2011 19:36:56 GMT

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

re: question #2, maybe I will put a zip file in the Bazaar.

It will be a working port to Mac OS X: with the runnable binary of theide, the archive libraries, a readme, and changed files.

Later, I will try out SVN and see if I can do a merge into it.

There is more work needed to make this a Mac application and runnable by clicking from the Finder. For now, one must run it from the shell:

```
~/theide
```

(which isn't so bad since users are programmers).

-Dave

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Wed, 15 Jun 2011 20:11:18 GMT

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

re: question #1

I'll start with GCC.bm and see if I can build from inside theide as well as command line.

I see that the uppsrc/Makefile is generated, so perhaps I ought to look at that code also.

I'll keep looking and learning for now.

-Dave

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [mirek](#) on Thu, 16 Jun 2011 07:54:26 GMT

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

daveremba wrote on Wed, 15 June 2011 01:56

In my code version, there is now a PLATFORM_MACOSX definition, so whatever worked for Windows/Linux,

Originally, it was meant to be PLATFORM_MACOSX11 for X11 under Mac...

Mirek

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [harmac](#) on Thu, 16 Jun 2011 09:38:31 GMT

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

I don't know the Mac environment but isn't an X11 application supposed to run on any X server? It doesn't mean to run the application natively on Mac, does it? I don't know if somebody has run a

U++ application for X11 on Windows using Xming or Cygwin/X, but it should work likewise, no? I then wouldn't call it running natively on Windows, though.

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Fri, 17 Jun 2011 02:34:56 GMT

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

[msg deleted]

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Fri, 17 Jun 2011 02:40:45 GMT

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

It seems that MacOS X is a slightly different flavor of UNIX (BSD I think, instead of the other major branch).

Yes, while X11 appears to be somewhat compliant, the compiler and linker flags are different. U++ does handle this difference in Linux with the OS flags (OSX11, SOLARIS, etc).

Dave

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [kohait00](#) on Tue, 05 Jul 2011 14:31:27 GMT

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

did i get it right? this port is somewhat 'X11' related on MacOSX?
what about the native port (cocoa, objective c)? anyone doing it?

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Fri, 08 Jul 2011 00:32:51 GMT

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

yes, it runs an X11 emulator over MacOSX. But I am finding some visual problems with the example applications. More info to follow ...

--

There are two problems with drawing that I can see:

1. buttons are flat (there is no border)
2. the area behind menus and certain controls shows an uninitialized bitmap (a backing store problem?)

Does not having GTK matter? (I think not).
There is a beta GTK for MacOS (Gimp uses it).

Either these draw problems could be tracked down in the X11 draw code, or the approach taken by some other frameworks is to draw using OpenGL.
In either case dealing with fonts is a hassle.
In the current implementation using X11 it already done, so that's a plus.

So maybe the first step is just to fix the current X11 draw problems, and I think that will give a usable U++ on MacOS.

p.s.

I do not think the nice rounded buttons of the Apple UI will be achieved through Quartz 2D; Cocoa would have to be used instead (using the C to Obj-C bridge).
This also seems pretty involved, and some developers here started on that.

Also layout might be slightly different between square Windows/Linux buttons and Apple buttons.

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [mirek](#) on Fri, 08 Jul 2011 06:21:26 GMT

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

daveremba wrote on Thu, 07 July 2011 20:32yes, it runs an X11 emulator over MacOSX.
But I am finding some visual problems with the example applications. More info to follow ...

--

There are two problems with drawing that I can see:

1. buttons are flat (there is no border)

Sounds like missing call to ChStdInit...

Quote:

Does not having GTK matter? (I think not).
There is a beta GTK for MacOS (Gimp uses it).

Well, I think installing GTK on Mac for now is worth trying.

I believe that following steps need to be achieved while developing U++/MacOS:

- Get theide working in MacOS / x11
- Alter theide so that it is capable of building Cocoa apps
- Develop SystemDraw. One particular complication there could be font metrics - not sure whether MacOSX is using freetype in base system
- Develop base CtrlCore
- Develop clipboard, drag&drop
- Develop MacOSX chameleon. Or maybe just a skin - not sure now how much skinning is usual on MacOSX

Quote:

I do not think the nice rounded buttons of the Apple UI will be achieved through Quartz 2D; Cocoa would have to be used instead (using the C to Obj-C bridge).

If you can DrawImage, you can draw rounded buttons

Mirek

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Sat, 09 Jul 2011 01:36:30 GMT

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

OK, here's where things are at:

- Get theide working in MacOS / x11

this is done - theide seems to work properly.
I can build apps, the editor works, the compiler,

linker, disassembler, package manager, menus,
all work correctly.

A generated Makefile for standalone building also
works correctly.

Also, internal debugging works.

Very little changes were necessary to the 3211
release to get to this point.

The NOGTK and OSX11 flags are needed for all
projects to build.

(how can this be made a platform default?)

External debugging does not work - Zero debugger is not installed.

Opening an external shell does not currently work.

fixing button & menu problems:

Problem does not happen in theide, only in
built applications (all).

I'll check for a missing call to ChStdInit.

Thanks.

- Alter theide so that it is capable of building Cocoa apps

well, Cocoa uses MVC pattern and will need a
C to Obj-C bridge. Not sure if this is an easy path.

Xcode writes out a NIB file; there used to be a
tool called nib2cib for just such purposes.

Another (inferior) path is to use Xcode to develop UI,
and provide access to UPP as a library (but then
code is not portable).

- Develop SystemDraw.

One particular complication there could be font metrics - not sure whether MacOSX is using
freetype in base system

you mean SystemDraw for Cocoa apps?

SystemDraw I think works
correctly for X11 apps (except for the flat button
and menu problems listed above on build apps).

- Develop base CtrlCore

again, for Cocoa via Obj-C bridge?

the various widgets in CtrlCore seem to draw correctly
inside the layout editor. If they appear correctly

in theide, I think they also would work and appear correctly in built applications?

- Develop clipboard, drag&drop

the functionality you have now for Linux/X11 should also work in MacOS/X11, but I'll test with other applications; e.g. text pasted to/from another application. I think it will work under X11

- Develop MacOSX chameleon.

Or maybe just a skin - not sure now how much skinning is usual on MacOSX

unlike Windows and Linux, MacOS doesn't really support different themes - there is a single Apple "look and feel".

probably getting the Apple look will only work through a Cocoa application.
Also, the X11 emulator is very basic, and has a minimal window manager.

For example, separate applications do not appear switchable with ALT-TAB (all the X apps currently running are grouped under the X server process). Apps are separate processes at the OS level, but at the Apple UI "Finder" like Windows "Explorer" individual X11 apps are not directly accessible. So that would be a another good reason to get a native Cocoa path working.

...

I do not think the nice rounded buttons of the Apple UI will be achieved through Quartz 2D; Cocoa would have to be used instead (using the C to Obj-C bridge).

If you can DrawImage, you can draw rounded buttons

Mirek

yes, that would make the app look better.
Thank you for your reply.

Dave

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [mirek](#) on Sat, 09 Jul 2011 06:36:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Quote:

- Alter theide so that it is capable of building Cocoa apps

[i]well, Cocoa uses MVC pattern and will need a C to Obj-C bridge. Not sure if this is an easy path.

Well, there is at least one 'proof of concept' - if uncertain, look at how Firefox manages it

Quote:

- Develop SystemDraw.

One particular complication there could be font metrics - not sure whether MacOSX is using freetype in base system

[i]you mean SystemDraw for Cocoa apps?

Yes.

Quote:

- Develop base CtrlCore

[i]again, for Cocoa via Obj-C bridge?

Yes.

Quote:

- Develop MacOSX chameleon.

Or maybe just a skin - not sure now how much skinning is usual on MacOSX

[i]unlike Windows and Linux, MacOS doesn't really support different themes - there is a single Apple "look and feel".

That is very good. In that case, all we need is to make some screenshots

Quote:

probably getting the Apple look will only work through a Cocoa application.

You can get Apple look for U++ even in Win32 quite easily...

Quote:

individual X11 apps are not directly accessible.
So that would be a another good reason to
get a native Cocoa path working.

Obviously, by MacOSX support I mean "native" support, which means Cocoa...

Quote:

I do not think the nice rounded buttons of the Apple UI
will be achieved through Quartz 2D; Cocoa would have
to be used instead (using the C to Obj-C bridge).

Actually, perhaps we do not even need bridge. Obj-C++ is still C++, right?

Mirek

Subject: Re: possibly working theide on Mac OS X using tiny mods to UNIX dist
Posted by [daveremba](#) on Mon, 11 Jul 2011 04:24:04 GMT

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

I think there ought to be more discussion in the forum
of a general nature, so let me reply in the
this thread instead.

The simple port of
TheIDE is working on MacOSX, and I think that
was the focus of this thread. There are
some problems with the resulting code,
perhaps resulting from not having GTK yet,
or maybe requiring a full port to Cocoa.

The question now is what exactly should it build
on MacOSX? How does that tie into Rainbow, etc?

-Dave
