Subject: HelloWorld makefile
Posted by forlano on Thu, 28 Nov 2013 19:36:51 GMT
View Forum Message <> Reply to Message

Hello,

I would like to test emscripten https://github.com/kripken/emscripten/wiki . The Duetto competitor has no good (for me) instructions.

It seems I must start with a Makefile and then substitute gcc with emcc (https://github.com/kripken/emscripten/wiki/Building-Projects). I am not so optimistic about it. However it would be fantastic to have a U++ package working in the browser (by the way, the emscripten test case work with Firefox, but fail with Chromium).

Anyway I need a working makefile project to start with.
I run under lubuntu 13.04 U++ 6572-raring-i386 , got by Honza PPA (it worked without problem as usual   ... thanks).

I was able to run the examples/HelloWorld package and then made a Makefile by menu item "Build/Output mode". The problem: I run "make" in the directory of the Makefile and appeared

```
luigi@luigi:~/HelloWorld$ make
mkdir -p _out/HelloWorld/GCC4.7-Gcc-Gui-Linux-Main-Posix-Shared/
mkdir -p _out/HelloWorld/GCC4.7-Gcc-Gui-Linux-Main-Posix-Shared/
mkdir -p _out/CtrlLib/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/CtrlCore/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/PdfDraw/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/Draw/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/plugin/bmp/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/RichText/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/Core/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
mkdir -p _out/plugin/png/GCC4.7-Gcc-Gui-Linux-Posix-Shared/
make: ***  No rule to generate the target "HelloWorld/hello.cpp", necessary for "
_out/HelloWorld/GCC4.7-Gcc-Gui-Linux-Main-Posix-Shared/hello .o ".  Stop.
```

So it created only few directories and then stopped. It is since 10 years that I am not using make and never I have known how it worked.
Please send me anything able to build the HelloWorld package (whole library + HelloWorld) or give me some advice to get one working.

Many thanks,
Luigi

Subject: Re: HelloWorld makefile
Posted by dolik.rce on Thu, 28 Nov 2013 21:33:56 GMT
View Forum Message <> Reply to Message

Hi Luigi,

You can try with the universal makefile. Good place to start would be probably something like make -f uppbox/lpbuild2/mkfile PKG=HelloWorld CXX=emcc executed in the checked out svn trunk. Inside the makefile, you can find additional documentation.

I tried to run it of course, but it fails miserably But I haven't even read the instructions, so your results may be better...

I'm really interested in the results, emscripten is very interesting technology. Let us know how it works (or doesn't work). And of course, if you have trouble with the makefile, I can try to help...

Best regards,
Honza

PS: Thinking of it now, is the makefile really necessary? Why not just create emcc build method in TheIDE?

Subject: Re: HelloWorld makefile
Posted by mirek on Fri, 29 Nov 2013 07:06:31 GMT

View Forum Message <> Reply to Message

dolik.rce wrote on Thu, 28 November 2013 16:33
PS: Thinking of it now, is the makefile really necessary? Why not just create emcc build method in TheIDE?

I would welcome that!

Note however that right now, there is intrinsic issue with emscripten and U++: Based on Javascript, emscripten inherits HTML5 runtime model, which basically does not allow (long story short) "Ctrl::Run" to exist...

So if ever we would want to support it, we would need "loop-less" mode for U++.

(Plus of course, half of Core would have to be made inactive...)

Mirek

Subject: Re: HelloWorld makefile
Posted by forlano on Fri, 29 Nov 2013 15:06:53 GMT

View Forum Message <> Reply to Message

dolik.rce wrote on Thu, 28 November 2013 22:33
PS: Thinking of it now, is the makefile really necessary? Why not just create emcc build method in

TheIDE?

Very interesting but no success. It halt immediatly. With message "Unknown OS".

I'll try to make a Makefile.

Thanks
Luigi

Subject: Re: HelloWorld makefile
Posted by forlano on Sat, 30 Nov 2013 10:57:57 GMT

View Forum Message <> Reply to Message

forlano wrote on Fri, 29 November 2013 16:06

I'll try to make a Makefile.

... and even the makefile temptive failed.

I remember that several years ago I was able to have for free and easily, from Theide, a working makefile of my own project. Now seems hard to get one good for a reference package.

Perhaps depends by the distribution and where the lib/include are...

From other side I have not understood if the Mirek's words mean that the issue prevent U++ to work via emscripten or it is only a potential problem.

Luigi

Subject: Re: HelloWorld makefile
Posted by mirek on Sat, 30 Nov 2013 12:44:13 GMT

View Forum Message <> Reply to Message

forlano wrote on Sat, 30 November 2013 05:57

From other side I have not understood if the Mirek's words mean that the issue prevent U++ to work via emscripten or it is only a potential problem.

Luigi

You can think about C++ running in emscripten as C++ running on some alien OS. U++ at this moment does not support that OS (just like it does not support e.g. iOS).

Now, it would probably be possible to add support for 'emscripten OS (eOS)' (and I would like to do that), but then there is one particular problem: there is no way how to implement explicit message loop in eOS. This in practice means that Ctrl::Run is impossible.

In the end, we could still have U++/eOS, but as Ctrl::Run is not possible, U++ applications for eOS would have to be specifically altered (quite heavily) to conform with this. E.g. this would not be possible anymore:

```
if(PromptYesNo("Do you want to perform this operation?")) {
    // perform the operation
}
```

instead, you would need to write

```
void MyApp::PerformOperation() {
    // perform the operation
}

PromptYesNo("Do you want to perform this operation?", THISBACK(PerformOperation))
```

Mirek

Subject: Re: HelloWorld makefile

Posted by forlano on Sun, 01 Dec 2013 11:54:28 GMT

View Forum Message <> Reply to Message

> mirek wrote on Sat, 30 November 2013 13:44
> In the end, we could still have U++/eOS, but as Ctrl::Run is not possible, U++ applications for eOS would have to be specifically altered (quite heavily) to conform with this. E.g. this would not be possible anymore:
>
> ```
> if(PromptYesNo("Do you want to perform this operation?")) {
>     // perform the operation
> }
> ```
>
> instead, you would need to write
>
> ```
> void MyApp::PerformOperation() {
>     // perform the operation
> ```

```
}

PromptYesNo("Do you want to perform this operation?", THISBACK(PerformOperation))
```

Mirek

Thanks for reply.
I am not sure to have completely understood. But if the problem is only to rewrite the Prompt_like() function of the user application, I am ready for my own app. Perhaps most of us are ready too.
However I am not aware if the whole library needs to be rewritten to accomodate such changes or just a new general function Prompt_eOS() should be added to overcome the issue.

I see huge advantages (both economics (no needs to own Mac & iOS device) and time saved (no new programming language)) that widely compensate the troubbles to rewrite the code of existing apps. Of course for a new app the problem does not exists.

Luigi

Subject: Re: HelloWorld makefile
Posted by mirek on Sun, 01 Dec 2013 12:01:27 GMT

View Forum Message <> Reply to Message

> forlano wrote on Sun, 01 December 2013 06:54
> I am not sure to have completely understood. But if the problem is only to rewrite the Prompt_like() function of the user application, I am ready for my own app. Perhaps most of us are ready too.

That is unfortunately just a tip of iceberg. Worse, as Run (and Execute) does not exist, you cannot write

```
void PerformDialog {
    MyDialog dlg;
    if(dlg.Execute() == IDOK)
        DoSomething();
}
```

> Quote:
> I see huge advantages (both economics (no needs to own Mac & iOS device) and time saved (no new programming language)) that widely compensate the troubbles to rewrite the code of existing apps. Of course for a new app the problem does not exists.

Yes, that is true.

Mirek