Subject: Re: Porting (Mac OS X) and "reference application" idea
Posted by mirek on Mon, 11 Jul 2011 12:59:11 GMT

daveremba wrote on Mon, 11 July 2011 00:52
What kind of port would be reasonable and useful
to the UPP community?
Probably reinventing
the entire Apple tool chain is not desirable.

I always tend to answer myself: Firefox.

Despite looks, Firefox is emulated GUI on Win32 and Linux, just like U++. I believe it is emulated GUI on MacOSX to, hint is e.g. here:

 http://mxr.mozilla.org/firefox/source/widget/src/cocoa/nsNat iveThemeCocoa.h

Quote:
is there a 1:1 correspondence of widgets to UPP?
(probably not)

Well, the desired result is that any single existing example in examples and reference runs on MacOSX in native mode and feels native. We are not aiming at 100% look&feel. 97% is just enough

Quote:
The UPP community would want an app that runs
on Linux and Windows to look similar, but also
conform to Apple look and feel, right?
(without much or any re-coding effort, like
Qt etc, but a better result than Qt)

Exactly. And we already know it is possible Linux vs Windows wise.

Quote:
There are several technical paths to chose from;
and picking the best one depends on what
the community expects from a porting effort
of UPP to Mac.

Actually, I do not think so. There is only one possible path and it is the same as current Linux and Windows backends...

I have outlined steps in previous posts.

Just to make things clear, we will NOT be using Cocoa widgets. We will be emulating them.

So the whole xcode/nib thing is mostly irrelevant to us. We will only use as little so that MacOSX recognizes our app as native. Nothing more.

To MacOSX it will be just windows with simple view and no subwidgets...

Quote:
off-topic:
A universal mobile UI framework that claims 1 million
developers for mobile devices and that I'm working
with is Sencha, which runs in Javascript.

The libarary is called ExtJS. Here is the
link:
http://www.sencha.com/products/extjs/

Sencha currently lacks a good data-object
connection to server (Direct-EXT is incomplete).
I recommend UPP developers take a look at it,
how might it relate to the Rainbow (or future UPP)
projects?

For another project, I am considering using UPP for the
server backend (since it is so fast compared to PHP, and
a much better language (C++ & good design)
compared to PHP). I'd need JSON/REST in UPP
to get my project working however.

Any comments on this would be appreciated
(maybe in another thread).

Interesting. I see rainbow and MacOSX as our responsibility, something that simply needs to be done, but after that, I definitely want to work quite a lot on U++ as web framework. Maybe even embrace Javascript in theide in some way...

That said, U++ is already being used as backend quite intensely in my work. But we are still missing "higher level" web framework...