Subject: Endian specific source changes, for Universal support.
Posted by lundman on Fri, 08 Dec 2006 01:33:04 GMT
View Forum Message <> Reply to Message

On OsX in particular, you can produce Universal binaries, applications with both Intel and PPC compiled code. This is enabled by giving gcc "-arch intel -arch ppc".

However, there are parts of U++ that currently are in #ifdef ENDIAN.

Have you considered changing said code for endian independant code, like bit-shifts etc, or at worst, use OS given endian functions like ntohl() etc.

There do not appear to be too many endian related calls in UPP.

Subject: Re: Endian specific source changes, for Universal support.
Posted by lundman on Fri, 08 Dec 2006 05:34:47 GMT
View Forum Message <> Reply to Message

So it seem OsX's gcc supply some defines when compiling for both architectures. I can, for example, use:

```
 #ifdef __BIG_ENDIAN__
   #define flagPPC
        #undef flagX86
 #endif

 #ifdef __LITTLE_ENDIAN__
        #undef flagPPC
        #define flagX86
 #endif
```

in Core.h, and let it pick the CPU type automatically.

I have been playing around with "Build methods" which I have largely ignored until now, and I see the potential there. Instead of any code change, I can define GCC32 method, GCC32_Intel, and GCC32_Universal methods, and specify the compile options needed.

However, what does seem to be missing is the linking flags.

Very nice though.

Subject: Re: Endian specific source changes, for Universal support.
Posted by mirek on Fri, 08 Dec 2006 08:25:13 GMT
View Forum Message <> Reply to Message

I always thought that for universal binaries, sources are compiled twice....

Mirek

Subject: Re: Endian specific source changes, for Universal support.
Posted by lundman on Fri, 08 Dec 2006 08:32:30 GMT
View Forum Message <> Reply to Message

Correct, but specify both options and gcc will do both passes for you automatically. But it is possible to completely compile both archs separately and then join them at the end using lipo.

Subject: Re: Endian specific source changes, for Universal support.
Posted by mirek on Fri, 08 Dec 2006 10:12:58 GMT
View Forum Message <> Reply to Message

lundman wrote on Fri, 08 December 2006 03:32
Correct, but specify both options and gcc will do both passes for you automatically. But it is possible to completely compile both archs separately and then join them at the end using lipo.

I would rather vote for this option. While debugging, you will be using just one mode anyway....
And for release, this is not that problem, moreover TheIDE can use lipo as well (in future).

Mirek