Subject: Localisation clarifications
Posted by lundman on Thu, 21 Sep 2006 06:18:43 GMT
View Forum Message <> Reply to Message

I have read the thread:
http://www.arilect.com/upp/forum/index.php?t=msg&th=1017 &start=0&

So I added a .t file (read that too literally, so eventually renamed it to lang.t). Added Swedish (seSV) and Japanese (jaJP), and Synchronised. This all looked good.

For example, I ended up with:

T_("File")

svSE("Fil")

T_("About")

svSE("Beskriv")

I then use:

SetLanguage(LNG_('S','V','S','E'));

And I use it:

```
void UFxp::MainMenu(Bar& bar){
 bar.Add(t_("File"),THISBACK(FileMenu));
}
void UFxp::FileMenu(Bar& bar){
 bar.Add(t_("About"),THISBACK(do_About));
}
```

And for some reason, my "File" menu is still "File", but "About" actually turns up in Swedish. In Japanese I just get square boxes, presumably as it uses the wrong font.

What I am curious about is, why would you want to use SetLanguage in your code? Should it not use the Environment variables to decide which locate is preferred? If I always pick SetLanguage(ENGLISH), would that not break the idea that anyone can do .tr files to add their own locale to an application?

Is SetLanguage really supposed to be used if I have a Language> menu, and let users pick a language that way? Ie, I should leave it alone, and only set it if I chose to offer an option to set

languages.

Should I be able to set LANG to seSV.UTF-8 or jaJP.UTF-8 and have my application use that language?

Subject: Re: Localisation clarifications
Posted by unodgs on Thu, 21 Sep 2006 08:56:44 GMT
View Forum Message <> Reply to Message

Have you added sth like this at the beginning of your main cpp file??

#define TFILE <your_file.t>
#include <Core/t.h>

Do it if not

Subject: Re: Localisation clarifications
Posted by lundman on Thu, 21 Sep 2006 09:48:25 GMT
View Forum Message <> Reply to Message

Yep, sure have. It works for most things (swedish) but not for Japanese.

Not sure which one to focus right now. Using 609dev-1

What is amusing is theide lets you modify your .t file, but does not work with IME input (@XMODIFIERS) so I can't use it to edit Japanese anyway.

Another thing is, before I added locale, if I set LANG to jaJP, my application automatically changed to the "wider" font, which is capable of displaying Japanese. Event though I had only SetLanguage(english) in there.

Now I added jaJP to my .t, and it no longer pics the right font, so no Japanese display works. I also tried to change to SetLanguage(jaJP) and SetDefaultCharset(utf8).

The manual/localization is very sparse.

Subject: Re: Localisation clarifications
Posted by mirek on Thu, 21 Sep 2006 19:36:41 GMT
View Forum Message <> Reply to Message

Have you set .t to UTF-8? (In TheIDE, Project, File properties..)

Subject: Re: Localisation clarifications
Posted by lundman on Fri, 22 Sep 2006 00:28:59 GMT
View Forum Message <> Reply to Message

That I have:

file lang.t
lang.t: UTF-8 Unicode C++ program text

echo LC_CTYPE
ja_JP.UTF-8
# head lang.t
T_("Engine Connect")

svSE("Engine Anknyta")

But, I suspect it IS working, but because it no longer (automatically?) changes to use the correct font, it just shows square boxes. Why did it pick the right font when no jaJP was defined? (Even though the app itself had no Japanese, it worked when I printed Japanese file names). Once I add jaJP, it no longer does.

Before (notice the wide/2byte font)
http://netbsd.interq.or.jp/~lundman/UFxp5.jpg

After (notice thin normal font, which I could also get if I unsetenv LC_CTYPE in the old build)
http://netbsd.interq.or.jp/~lundman/UFxp6.jpg

Subject: Re: Localisation clarifications
Posted by mirek on Fri, 22 Sep 2006 09:04:24 GMT
View Forum Message <> Reply to Message

lundman wrote on Thu, 21 September 2006 02:18
What I am curious about is, why would you want to use SetLanguage in your code? Should it not use the Environment variables to decide which locate is preferred? If I always pick SetLanguage(ENGLISH), would that not break the idea that anyone can do .tr files to add their own locale to an application?

Acutally, we have left this decision to application designer. E.g. application can pick locale settings and use proper SetLanguage based on it.

Quote:
Is SetLanguage really supposed to be used if I have a Language> menu, and let users pick a language that way? Ie, I should leave it alone, and only set it if I chose to offer an option to set languages.

Yes, that is another option.

Mirek

Subject: Re: Localisation clarifications
Posted by mirek on Fri, 22 Sep 2006 09:07:49 GMT
View Forum Message <> Reply to Message

lundman wrote on Thu, 21 September 2006 20:28
That I have:

file lang.t
lang.t: UTF-8 Unicode C++ program text

echo LC_CTYPE
ja_JP.UTF-8
# head lang.t
T_("Engine Connect")

svSE("Engine Anknyta")

But, I suspect it IS working, but because it no longer (automatically?) changes to use the correct font, it just shows square boxes. Why did it pick the right font when no jaJP was defined? (Even though the app itself had no Japanese, it worked when I printed Japanese file names). Once I add jaJP, it no longer does.

Before (notice the wide/2byte font)
http://netbsd.interq.or.jp/~lundman/UFxp5.jpg

After (notice thin normal font, which I could also get if I unsetenv LC_CTYPE in the old build)
http://netbsd.interq.or.jp/~lundman/UFxp6.jpg

Well, this really looks like wrong font used. Frankly, I am becoming a kind of desperate about X.org / *X desktop environment font settings...

Anyway, you can try to solve the problem by Draw::SetStdFont - at least we will know what is going on...
