Subject: Re: Translation in static members
Posted by mirek on Fri, 18 Feb 2011 11:38:38 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Sat, 05 February 2011 12:29I'll answer myself

The simplest and probably correct solution is to use t_GetLngString() whenever using the static member, instead when initializing it:#include <Core/Core.h>
using namespace Upp;

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

//simple shorthand, to keep code nice looking
#define _t(X) t_GetLngString(X)

struct test{
 static const char* str;
};
const char* test::str=tt_("translation");

CONSOLE_APP_MAIN{
 SetLanguage(GetSystemLNG());
 test t;
 DUMP(_t(t.str)); //<- added _t() to translate the string at runtime
}
The _t macro is quite handy thing. It would actually work with t_ as well, but that confuses theide when syncing the translations. Maybe there could be some shorthand for t_GetLngString added in the U++, what do you think?

Honza

I guess calling t_GetLngString or GetLngString here is not a big problem, as IME static texts are not that frequent.

However, if we decided on shortcat synonyme, it would be better done as inline function - no need to use macro here

Mirek