Subject: Re: BUG: Serious problem with Core/CharSet
Posted by tojocky on Fri, 04 Jun 2010 05:28:41 GMT
View Forum Message <> Reply to Message

kov_serg wrote on Fri, 04 June 2010 01:22Upp2361/WinXP

Testing code shows "error 128=31"

```
#include <Core/Core.h>

using namespace Upp;
CONSOLE_APP_MAIN {
 wchar_t w[]={0x0410,0}; // must be converted to 0x80,0 see definition of CHRTAB_CP866
 String r=FromUnicode(w,CHARSET_CP866);
 int rc=r[0]&255;
 Cout()<<((rc==0x80)?"ok ":"error 128=")<<rc<<"\n";
}
```

CHARSET_CP866 has code 62 but if we use 60 instead 62 it will works.

The bug hides here:

Core\CharSet.cpp:2252

```
...
byte  ResolveCharset(byte charset)
{
 return charset ? charset : DefaultCharset;
}

inline
static CharSetData& s_cset(byte charset)
{
 return s_map()[ResolveCharset(charset)]; // <<<< why charset code used here instead of real
index String("CP866") ???
}
```

s_map() returns ArrayMap<String, CharSetData>& and shuld be indexed by string "CP866" not by charset code 62. Because it real index is 60.

Look at Core\CharSet.cpp:2206

```
 int q = s_map().GetCount();
```

And try to ASSERT(q==systemcharset); and this assert fail almost always.

May be at beginig charset codes was equal to s_map item index, but now this is not true. And this

is a problem function FromUnicode works wrong.

Some days ago I had problems with charset. but the problem was not with u++. u++ works perfectly. the chars tables and conversions are perfectly.