Subject: Format NFormat and so on.
Posted by lundman on Tue, 09 May 2006 07:18:48 GMT
View Forum Message <> Reply to Message

It's always the trivial things I have to ask for

What is the deal with Format and NFormat. I have not really found any documentation giving details to its use. But looking at sources, it seems %` (percent-backtick) prints date/time, but always in US order. No help for SetDateFormat, and only t.cpp has an example that does not change the format (when used with %`).

%5.2g and %5.2f give me asserts. The docs seem to favour %n instead, and sometimes %5.2n works, but with more than 2 digits in the decimal place, sometimes asserts. %5n seems to work ok, just not the way I was hoping.

What is the difference between NFormat and Format? The examples seem to use both.

Subject: Re: Format NFormat and so on.
Posted by mirek on Tue, 09 May 2006 08:52:29 GMT
View Forum Message <> Reply to Message

lundman wrote on Tue, 09 May 2006 03:18
It's always the trivial things I have to ask for

I do not think this is trivial....

Quote:
What is the deal with Format and NFormat. I have not really found any documentation giving details to its use. But looking at sources, it seems %` (percent-backtick) prints date/time, but

%` is "empty" - default format. In fact, ` can be used as delimiter of format code; the trouble is that unlike classic printf, Format allows you to define your own formats with id's, means you sometimes need delimiter to tell apart id and text..

Quote:
always in US order. No help for SetDateFormat, and only t.cpp has an example that does not change the format (when used with %`).

%5.2g and %5.2f give me asserts. The docs seem to favour %n instead, and sometimes %5.2n works, but with more than 2 digits in the decimal place, sometimes asserts. %5n seems to work ok, just not the way I was hoping.

Not sure where is the problem:

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

CONSOLE_APP_MAIN
{
 LOG(Format("%'", GetSysDate()));
 SetLanguage(LNG_('C','S','C','Z'));
 LOG(Format("%'", GetSysDate()));
 SetLanguage(LNG_('E','N','U','S'));
 SetDateFormat("%2:R/%1:d %4:Day");
 LOG(Format("%'", GetSysDate()));
 LOG(Format("%6.4g", 1.23));
}
```

.log:

```
* e:\out\MSC71.Debug_full\Format.exe 09.05.2006 10:47:16, user: cxl

05/09/2006
09.05.2006
V/2006 Tuesday
  1.23
```

If troubles persist, send the code!

Quote:
What is the difference between NFormat and Format? The examples seem to use both.

They are the same. You should use straight Format.

(Two names are just some echoes of the past - originally, Format behaved like Sprintf, using C standard library. Then we decided to extend it for any Values and new was named NFormat to keep BWC while deprecating Format. When our code was Format clean, we hijacked Format for NFormat behaviour. Happened 2 years ago anyway

Mirek

Subject: Re: Format NFormat and so on.
Posted by lundman on Tue, 09 May 2006 10:15:25 GMT

View Forum Message <> Reply to Message

Quote:
%` is "empty" - default format. In fact, ` can be used as delimiter of format code; the trouble is that unlike classic printf, Format allows you to define your own formats with id's, means you sometimes need delimiter to tell apart id and text..

Yeah I suspected as much reading Format.cpp, but alas, without Documentation, or examples, of how to use that code...

Quote:

If troubles persist, send the code!

Only example I found was

```
uppsrc/Core/t.cpp:	SetDateFormat(t_("date-format\a%2:02d/%3:02d/%1:4d"));
```

and not having any specs on it, I swapped %2 with %3 thinking that would do it, but it did not.

Subject: Re: Format NFormat and so on.
Posted by lundman on Wed, 10 May 2006 02:09:43 GMT

View Forum Message <> Reply to Message

Cut out the extra junk I had from the example, and used yours and it works fine.

Left with:

```
// "Wednesday 10/05/2006 10:23:43"
SetDateFormat("%4:Day %3:02d/%2:02d/%1:4d");
```

Which will do.. should really give the user the option to chose this, but it would be too complicated to expect a user to be able to enter a magic string like that, when even I don't know fully what the deal is

The double problem is probably more something to do with that I should cast it to date/time to print it.

```
 double size, cps;

 return Format("%5.2n MB Remaining at %5.2n KB/s should take %` seconds",
  size/1024.0, cps, size / 1024.0 / cps);
```

```
54793.43359 MB Remaining at 1715.27 KB/s should take 31 seconds
```