Subject: Re: Thread::GetCurrentThreadId() and Thread::GetCurrentThreadHandle() new methods
Posted by tojocky on Sun, 09 Jan 2011 21:28:29 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 08 January 2011 14:07Ok, added this:

ThreadId does not make sense for me now (IMO: it is Win32 specific and not really related to Thread).

Mirek,

I think that you are not right according by:
http://suacommunity.com/dictionary/pthread_self-entry.php

Quote:In the Windows threading model each created thread has both a HANDLE and a system-wide unique id. As a result the GetCurrentThreadId Windows function returns the same logical information as the POSIX call.

The method DWORD WINAPI GetCurrentThread(void) returns the pseudo-handle, that is not same with the result _beginthreadex(...).

In the other had, you are right, because in POSIX you can manage with the result pthread_self.

In the end, I need an unique Thread ID.

mirek wrote on Sat, 08 January 2011 14:07Ok, added this:

ThreadId does not make sense for me now (IMO: it is Win32 specific and not really related to Thread).

Note: The faster alternative to all this might be checking the pointer to TLS variable.
About Your Note, can you give me an example, please?

Thank you in advance!

Added:
By TLS variable you mean: Thread-local storage variable?

Like this:

```
thread__ bool sThreadId;

qword GetCurrentThreadIdCustom(){
 return (qword (&sThreadId));
}
```