Subject: Re: Simple class to handle variables used by different threads
Posted by mirek on Mon, 24 Aug 2015 17:13:37 GMT
View Forum Message <> Reply to Message

Well, depends on type and CPU...

Usually, if type is directly supported by CPU, it works.

Example of type that does not work is e.g. int64 in 32-bit mode. It is because it is, on CPU level, compound type...

Anyway, MT is not hard because of transfering data between threads. If you think about it, they have to get transfered at some point... The real issues are order:

if one thread does

a = a1;
b = b1;

then another thread can see b updated first. This is what barriers are for.

(Then, of course, there is an issue that you cannot never tell WHEN the change is visible in another thread, but that is really not a problem...)

And then, of course, serialization, which is about transactions:

a = a + 1;

now this, on CPU level is something like

read a into register
increment register // remember this point A
write register into a

Now imagine if another thread starts incrementing a at point A...

Well, there really is complex theory about lock-less multithreading, it is pretty tough stuff. I would not dare suggesting new MT tools before reading it all :)

Mirek