GROMACS - Bug #1284
syntax error in src/gromacs/legacyheaders/thread_mpi/atomic/xlc_ppc.h

06/16/2013 10:53 PM - Jeff Hammond

Status: Closed
Priority: Normal
Assignee: Sander Pronk
Category: mdrun
Target version: 4.6.3
Affected version - extra info:
Affected version: git master

Description
From patch description:
fix declaration that XLC rejected

i have no idea what "volatile char* volatile*" is supposed to do, but all that is required is "volatile void **" hence that is what i have used instead.

Associated revisions
Revision 7d5861d0 - 06/25/2013 04:03 PM - Jeff Hammond
Fix declaration that XLC rejected
Changed volatile void* volatile* declarator to volatile void* volatile.
Fixes #1284
Change-Id: I133f480b060ab4723988bd3059e5bd20ebc9df7a

History
#1 - 06/17/2013 12:21 AM - Mark Abraham
Looks like a regexp fail to me :-) Thanks for the series of fixes, which I have uploaded for review, e.g. https://gerrit.gromacs.org/#/c/2450/

#2 - 06/17/2013 03:15 PM - Sander Pronk
- Assignee set to Sander Pronk

This is unfortunately not a typo :-| It's what made the atomics work - in a previous version of the xlc compiler. The documentation of xlc now says that there are built-in atomic functions such as compare and swap, so I'll add a version check and then use those for the newer compilers.

#3 - 06/17/2013 04:23 PM - Jeff Hammond
I don't see how that is valid C syntax so it should have been filed as a bug with IBM rather than committed to Gromacs. If all else fails, one can write the code in GNU inline assembly, compile with GCC and link the object code into an XLC build.

I have access to XLC 9, 11 12 (BGP, POWER7 and BGQ, respectively) and can test this code against all of them and provide the appropriate preprocessor magic (i.e. get version-specific as necessary using IBMC - http://sourceforge.net/p/predef/wiki/Compilers/).

In any case, I agree that using the built-in atomics functions are a better idea anyways. I'll work on a patch for all of the aforementioned compilers. BGP and BGQ both have wrappers to LLSC assembly in the driver and I know that at least XLC 11+ supports the GCC macros for atomics.

#4 - 06/17/2013 04:46 PM - Sander Pronk
Actually,

volatile void* volatile p;

is valid C syntax, which means: treat both the pointer value and the item it points to as volatile. This is actually what we want, so perhaps all that needs to be done is to change lines 142&143 to
volatile char* volatile* oldv = oldval;
volatile char* volatile* newv = newval;

to

volatile char* volatile oldv = oldval;
volatile char* volatile newv = newval;

and make sure the same thing happens on line 86.

#5 - 06/17/2013 04:47 PM - Sander Pronk
BTW I don't have easy access to an xlc compiler so would you mind trying that?

#6 - 06/19/2013 07:05 PM - Mark Abraham
- volatile char* volatile* oldv = oldval;
- volatile char* volatile* newv = newval;
+ volatile char* volatile oldv = (char *) oldval;
+ volatile char* volatile newv = (char *) newval;

did compile on xlc 12.1 for BG/Q (and GCC 4.4.7 on linux frontend compiler)

#7 - 06/20/2013 05:06 PM - Mark Abraham
- Category set to mdrun
- Status changed from New to Fix uploaded
- Target version set to 4.6.3

https://gerrit.gromacs.org/#/c/2450/ looks good to me. Unless Jeff has any further comment, that will make GROMACS 4.6.3.

Working on a proper fix using built-ins will take more time.

#8 - 06/20/2013 06:31 PM - Jeff Hammond
I don't fully understand the casting here but if it compiles and runs correctly, that's all the proof I need. I will try to work on the built-ins version.

#9 - 06/25/2013 04:05 PM - Jeff Hammond
- Status changed from Fix uploaded to Resolved
- % Done changed from 0 to 100

Applied in changeset 7d5861d0f07e7b3ba8b954db863e660fceed6a5f.

#10 - 12/03/2013 04:16 PM - Rossen Apostolov
- Status changed from Resolved to Closed

Files
0005-fix-declaration-that-XLC-rejected.patch 1.16 KB 06/16/2013    Jeff Hammond