

OpenRTM-aist (C++) - バグ #2453

Feodra17 i386でrpmのパッケージビルドがエラーになる

2012/06/07 20:05 - n-ando

ステータス:	終了	開始日:	2012/06/07
優先度:	通常	期日:	
担当者:		進捗率:	100%
カテゴリ:		予定工数:	0.00時間
対象バージョン:			
説明 rtcd をリンクするときに undefined symbol __atomic_fetch_add_4 でエラーになる。			

関係しているリビジョン

リビジョン 2371 - 2012/06/07 18:57 - n-ando

rpmbuild in i386 sets -march=i386 in CFLAGS/CXXFLAGS. -march causes undef. ref. of __atomic_fetch_add. To avoid this problem, default CFLAGS/CXXFLAGS setting is changed in spec file. refs #2453

リビジョン 2371 - 2012/06/07 18:57 - n-ando

[->trunk] rpmbuild in i386 sets -march=i386 in CFLAGS/CXXFLAGS. -march causes undef. ref. of __atomic_fetch_add. To avoid this problem, default CFLAGS/CXXFLAGS setting is changed in spec file. refs #2453

リビジョン 2371 - 2012/06/07 18:57 - n-ando

[->trunk] rpmbuild in i386 sets -march=i386 in CFLAGS/CXXFLAGS. -march causes undef. ref. of __atomic_fetch_add. To avoid this problem, default CFLAGS/CXXFLAGS setting is changed in spec file. refs #2453

履歴

#1 - 2012/06/08 10:20 - n-ando

OpenRTM ML皆様

産総研 安藤です

どなたかFedora17 (32bit) 上で OpenRTM-aistのC++版のコンパイルを通したことがある方はいらっしゃいませんか？

実行形式にリンクする際に、軒並み以下のようなエラーがでて困っております。64bit版では出ていないみたいです。

```
../../../../src/lib/rtm/.libs/libRTC.so: undefined reference to `__atomic_fetch_add_4'  
collect2: error: ld returned 1 exit status
```

g++: gcc version 4.7.0 20120507 (Red Hat 4.7.0-5) (GCC)
OS: Fedora17, 32bit
OpenRTM: OpenRTM-aist-1.1.0-RELEASE

#2 - 2012/06/08 10:20 - n-ando

清水先生

安藤です

情報有り難うございます。

結論としては、コンパイルは通しましたが、原因はよくわかりませんでした（笑）一応もう少しで、1.1.0-RELEASEのFedora用rpmをリリースできる見込みです。

以下は、gcc 4.7, rpmパッケージ作成に興味のある方向け

Fedora17のgcc (4.7) では、-march=i386 オプションを付けてコンパイルすると、ある種の条件下（詳しくは調べてませんが）で__atomic_fetch_add_4 という関数（おそらくgccの組み込み関数）への呼び出しが暗黙的に追加されるようです。
オプションが -march=i468 i586 i686 だと追加されない。

しかしながら、色々探した限りでは__atomic_fetch_add_4 関数の宣言や実体が無いのと、Fedora14,15,16で作成した

libRTCやlibcoilにはそういうシンボルが一切見当たらないので、
まあ、なくても問題ないだろうと考えました。(あったらすみません。)

-march=i386オプションは、自分でソースからconfigure;makeする
時にはコンパイルオプションとして追加されないのですが、
rpmパッケージを作るためにrpmbuildを使うと、CFLAGS/CXXFLAGSに
のように勝手に追加されてしまいます。

CFLAGS=-O2 -g -march=i386 -mtune=i686
(これはFC14,15,16でも同様、でも問題は起こらない。)

ただし、x86_64版のrpmbuildは

CFLAGS=-O2 -g

というオプションのみです。なので、specファイルを少し書き換えて、
i386版も-O2 -gオプションのみになるように強制したところ、
一応パッケージの作成まで出来ました。

なんかすっきりしませんが、gcc 4.7 が主流になってくれば
何かわかるかもしれません。

他に情報お持ちの方がおられましたら、教えてもらえませんか？
よろしく願います。

#3 - 2012/06/08 10:20 - n-ando

安藤様

清水です。

標記の件、少し調べてみました。
問題は解決しませんでした。以下分かったことです。

(1) Fedora17でomniORBをコンパイル
-march=i386オプションを付けて、omniORBをソースからコンパイルしてみました。
結果、問題なくコンパイルできました。
作成されたバイナリをreadelfで見ましたが、
__atomic_fetch_add_4 という参照はありませんでした。
どうやら、CORBAの問題では無さそうです。

(2) Ubuntu12.04でOpenRTM-1.1.0をコンパイル
-march=i386オプションを付けて、RTM1.1.0をソースからコンパイルしてみました。
g++のバージョンは、4.6.3です。
結果、以下のエラーでコンパイルできませんでした。

```
../../src/lib/rtm/.libs/libRTC.so: undefined reference to `__sync_fetch_and_add_4'
```

(3) __sync_fetch_and_add_4 のエラー原因の検索
上記のエラーメッセージをググってみました。
すると、いろいろな情報が見つかりました。

__sync_fetch_and_add_4 がlegacyなインタフェースで、
C++ では、std::atomic<>に置き換わったので、
__sync_fetch_and_add_4 を使わないようにソースを
書き換えれば良いという情報がありました。

上記が正しいとすれば、__atomic_fetch_add_4 もおそらく同じだと思います。

また、簡単な回避法としては、-march=i486 以上をオプションに指定すれば良い
との情報もありました。
i386だけダメなようです。

ところで、根本的な疑問として、RTMではatomic関係のインタフェースを
どこで使っているのでしょうか？
src/lib/rtm以下のすべての*.oファイルに__atomic_fetch_add_4 への
参照が含まれてしまっています。

以上、全然すっきりしませんが、わかった事です。
さらなる情報をご存知の方は教えて頂けると幸いです。

#4 - 2012/06/08 10:21 - n-ando

清水先生

安藤です

調査ありがとうございます。

安藤様

清水です。

標記の件、少し調べてみました。
問題は解決しませんでした、以下分かったことです。

(1) Fedora17でomniORBをコンパイル
-march=i386オプションを付けて、omniORBをソースからコンパイルしてみました。
結果、問題なくコンパイルできました。
作成されたバイナリをreadelfで見ましたが、
__atomic_fetch_add_4 という参照はありませんでした。
どうやら、CORBAの問題では無さそうです。

そうですか。となると、OpenRTM側の問題の可能性が大きいですね。

(2) Ubuntu12.04でOpenRTM-1.1.0をコンパイル
-march=i386オプションを付けて、RTM1.1.0をソースからコンパイルしてみました。
g++のバージョンは、4.6.3です。
結果、以下のエラーでコンパイルできませんでした。

```
../src/lib/rtm/.libs/libRTC.so: undefined reference to `__sync_fetch_and_add_4'
```

4.7 だけ、というわけでもなさそうですね。

(3) __sync_fetch_and_add_4 のエラー原因の検索
上記のエラーメッセージをググってみました。
すると、いろいろな情報が見つかりました。

__sync_fetch_and_add_4 がlegacyなインタフェースで、
C++では、std::atomic<>に置き換わったので、
__sync_fetch_and_add_4 を使わないようにソースを
書き換えれば良いという情報がありました。

なるほどそうでしたか。これってC++0x11あたりで導入される機能ですよ。
旧C++とC++0x11が混ざって悪さしているとかですかね？

上記が正しいとすれば、__atomic_fetch_add_4 もおそらく同じだと思います。

また、簡単な回避法としては、-march=i486 以上をオプションに指定すれば良い
との情報もありました。
i386だけダメなようです。

そうですね。i386だけダメのようです。
__sync_fetch_and_add_4 や __atomic_fetch_add_4 に相当する機能が
i386にはなく、i486以降には存在するので、ソースのどこかで
__sync_fetch_and_add_4() や __atomic_fetch_add_4()関数を使っている関数が
呼び出されているが、i386でコンパイル時にはこれらに対応するコードを
生成することができずエラーになる、ということでしょうね。
undefined referenceというエラーは正しくない、といっている人も居ました。

となると、coilで特定のCPUアーキテクチャに依存したコードはとくに書いた
覚えはないので、標準ライブラリに古いコードが残っていて、-march=i386
オプションが指定されているにもかかわらず、それが生きてしまって
エラーになっているとも考えられますかね？ そうなると、gcc周りの
問題かな、という気がしますね。

ところで、根本的な疑問として、RTMではatomic関係のインタフェースを
どこで使っているのでしょうか？
src/lib/rtm以下のすべての*.oファイルに __atomic_fetch_add_4 への
参照が含まれてしまっています。

RTMというより、まずcoilの方じゃないかと思います。Fedora17では
stringutil.oとかProcess.oにすでに、atomic_fetch_add_4が
含まれてました。

Mutex関係なら理解できるんですが、とくにatomicな処理とか関係なさそうな stringutilでこの関数への参照が含まれているので、特定の関数ではなくて原因はgccのオプションかな、と思った次第です。

includeしているヘッダかなとも思ったのですが、関係ありそうなヘッダは無いですし。あと、グローバルにインスタンス化しているオブジェクトも関係あるかなと思ったのですが、特にそういうのも無いですしね。
iostreamか？

以上、全然すっきりしませんが、わかった事です。
さらなる情報をご存知の方は教えて頂けると幸いです。

ありがとうございました。OpenRTM/coilのコードも時間があるとき調べてみたいと思います。

#5 - 2012/06/08 10:22 - n-ando

安藤様

ジェフです。

2012/6/8 Ando Noriaki <n-ando@aist.go.jp>

(3) _sync_fetch_and_add_4 のエラー原因の検索
上記のエラーメッセージをググってみました。
すると、いろいろな情報が見つかりました。

_sync_fetch_and_add_4 がlegacyなインタフェースで、
C++ では、std::atomic<> に置き換わったので、
_sync_fetch_and_add_4 を使わないようにソースを
書き換えれば良いという情報がありました。

なるほどそうでしたか。これってC++0x11あたりで導入される機能ですよ。
旧C++とC++0x11が混ざって悪さしているとかですかね？

std::atomicはC++11から導入されたからまだオフィシャルなコンパイラのサポートはゼロです。オフィシャルではないならVC2011とgcc 4.5(?)以上にありますが、フラグを設定することが必要です。(そして、C++11からのフィーチャーならlegacyの方はまだあるはずですが...)

したがって、使うことに注意が必要でしょう。coilにstd::atomicを既定として導入するとVC2010でもコンパイルできなくなります。オプションにした方がいいと思います。以下のファイルの68行目は参照になると思います。

https://github.com/gbiggs/tide/blob/api_reorg/include/tide/tide_config.h.in

#6 - 2012/06/08 10:22 - n-ando

安藤です

coil本体ではC++0x11やboostなどはしばらく使わない予定です。
coil自体が移植性を向上させるものなので使えないOS・処理系が主流である間は新しい機能に手を出すつもりはありませんのでご安心を。
拡張モジュールでは一部 boost 使ったりもしていますが。

#7 - 2012/06/08 10:24 - n-ando

- ステータスを新規から解決に変更

- 進捗率を0から100に変更

以下のようにして-march=i386を回避

Index: OpenRTM-aist/packages/rpm/OpenRTM-aist.spec.in

```
-----
--- OpenRTM-aist/packages/rpm/OpenRTM-aist.spec.in      (リビジョン 2370)
+++ OpenRTM-aist/packages/rpm/OpenRTM-aist.spec.in      (リビジョン 2371)
@@ -78,7 +78,7 @@
#-----
# build section
%build
-%configure --prefix=/usr
+%configure --prefix=/usr CFLAGS="-O2 -g" CXXFLAGS="-O2 -g"
%{_make}
#-----
```

#8 - 2012/06/08 10:25 - n-ando

- ステータスを解決から終了に変更