

Robotics Systems Commons - Bug #1570

0.7 version does not build on MacOS

07/22/2013 05:37 AM - J. Moringen

Status:	Resolved	Start date:	07/22/2013
Priority:	Normal	Due date:	
Assignee:	S. Wrede	% Done:	100%
Category:	cmake	Estimated time:	0.00 hour
Target version:	rsb-0.7		
Description			
See https://ci.cor-lab.org/job/rsc-0.7-macos/			
Related issues:			
Related to Robotics Service Bus - Bug # 1571: 0.7 version does not build on M...		Resolved	07/22/2013

History

#1 - 07/22/2013 05:43 AM - J. Moringen

- Description updated

#2 - 07/22/2013 01:23 PM - J. Moringen

- Assignee changed from J. Wienke to S. Wrede

#3 - 07/22/2013 02:36 PM - J. Wienke

H, looks like a make clean is missing. I think we updated the boost version with the homebrew update.

#4 - 07/22/2013 02:53 PM - S. Wrede

Did you install boost with homebrew in the universal variant (32+64) bits?

From the log:

```
Linking CXX shared library ../build/librsc.dylib
```

```
Undefined symbols for architecture x86_64:
```

```
"boost::filesystem::path::operator/=(boost::filesystem::path const&)", referenced from:
  boost::filesystem::operator/(boost::filesystem::path const&, boost::filesystem::path const&) in Environment.cpp.o
"boost::this_thread::interruption_point()", referenced from:
  void boost::condition_variable_any::wait<boost::unique_lock<boost::recursive_mutex> >(boost::unique_lock<boost::recursive_mutex>&) in
  RepetitiveTask.cpp.o
  void boost::condition_variable_any::wait<boost::unique_lock<boost::recursive_mutex> >(boost::unique_lock<boost::recursive_mutex>&) in
  SimpleTask.cpp.o
"boost::this_thread::hidden::sleep_until(timespec const&)", referenced from:
  boost::this_thread::sleep(boost::posix_time::ptime const&) in PeriodicTask.cpp.o
  boost::this_thread::sleep(boost::posix_time::ptime const&) in ThreadedTaskExecutor.cpp.o
"boost::detail::once_epoch_cv", referenced from:
```

#5 - 07/25/2013 11:09 AM - J. Wienke

I didn't do anything special. Just the usual flags.

#6 - 07/27/2013 01:05 PM - S. Wrede

OK, I could reproduce the error on my machine with latest brew updates installed. It looks as if the linker command lacks the required boost dependencies but this needs further investigation.

#7 - 07/30/2013 08:18 PM - S. Wrede

- *Category set to cmake*
- *Status changed from New to Feedback*
- *% Done changed from 0 to 70*

The origin of this problem seems to be the call to `FIND_PACKAGE(boost)` in `FindBoostUUID.cmake`. For now, I reversed the order of `boost` and `boost uuid` checking in the `rsc CMakeLists`, which works on my machine and the Jenkins lion64 build slave.

Johannes: It would be great if `FindBoostUUID.cmake` can be implemented such that it does not expose this side effect.

#8 - 07/30/2013 09:13 PM - J. Wienke

This was something introduced in more recent versions of `cmake` with an update in `FindBoost`. I already tried several things to get around this. In any case, we can start to drop `FindBoostUUID` completely now, since `UUID` is included in `boost` for some time now.

#9 - 09/12/2013 04:33 PM - S. Wrede

- *Status changed from Feedback to Resolved*

Fixed (hypothesis: boost update on slave solved this).

#10 - 09/12/2013 04:33 PM - S. Wrede

- *% Done changed from 70 to 100*