

Robotics Service Bus - Bug #423

Fix Boost Compile/Link behavior

07/15/2011 11:40 AM - M. Rolf

Status:	Resolved	Start date:	07/15/2011
Priority:	Normal	Due date:	
Assignee:	J. Wienke	% Done:	100%
Category:	C++	Estimated time:	0.00 hour
Target version:			
Description			
<p>If the configured boost-uuid path contains a full boost installation (instead of just UUID), RSB is completely compiled against this boost-installation, but linked to the standard boost installation.</p> <p>Test configuration on Lucid: standard boost 1.40 in /usr full boost 1.42 in /usr/local/boost1.42 Configured BOOSTUUID_INCLUDE_DIRS=/usr/local/boost1.42/include</p> <p>Result: Compiled (entirely) against 1.42 Linked against 1.40</p> <p>=> Nothing works</p> <p>Probable fix: permutate -I directives for compilation? In any way it needs to compile and link against same location</p>			

Associated revisions

Revision fed419c0 - 09/23/2011 03:13 PM - J. Wienke

be sure that our configure paths have precedence over system ones

refs #423

History

#1 - 09/12/2011 04:57 PM - J. Wienke

- Status changed from New to In Progress
- Assignee set to J. Wienke

I don't see a real solution to this, except that we now provide an internal uuid version.

If we exchange the -I directives we get the same error the other way around. Suddenly boost.uuid will be used from the boost location that contains uuid and not your custom boost.uuid version. This is the drawback of the linux scheme with a common prefix for multiple programs.

#2 - 09/12/2011 04:57 PM - J. Wienke

- Category changed from Configuration to C++

#3 - 09/23/2011 03:14 PM - J. Wienke

- *Status changed from In Progress to Resolved*

- *% Done changed from 0 to 100*

I now at least ensure that custom paths have precedence over system paths. Moreover the include order should already be boost, boost.uuid... Hence, boost should be taken from the first path not the uuid one.