

brainvisa-development - Defect #17763

boost detection with multiple config prevents from setting its path

04/01/2018 03:27 PM - Riviere, Denis

Status:	Closed	Start date:	04/01/2018
Priority:	Normal	Due date:	
Assignee:		% Done:	100%
Category:	brainvisa-cmake	Estimated time:	0:00 hour
Target version:	brainvisa-4.6	Spent time:	0:00 hour
Resolution:			
Description			
<p>We have this problem on CentOS 5.11 for instance: boost include is installed in /usr/include (plus possibly in /usr/local/include), and in /i2bm/brainvisa/CentOS-5.11-x86_64/boost/include (a non-standard path). This latest path is normally added in cmake search paths, but it finds anyway one of the two previous ones, and setting the value manually in the cache does not change anything: the manual value is reset to /usr/include or /usr/local/include. In consequence, we cannot use our boost, which has the required version.</p>			

Associated revisions

Revision 80324 - 04/01/2018 04:46 PM - Riviere, Denis

setup Boost config variables before 1st call to find_package(Boost)
#17763

Revision 80324 - 04/01/2018 04:46 PM - Riviere, Denis

setup Boost config variables before 1st call to find_package(Boost)
#17763

Revision 80325 - 04/01/2018 04:59 PM - Riviere, Denis

i2bm-specific: set Boost_INCLUDE_DIR to /i2bm/brainvisa/<OS>/boost/include if found
with <https://github.com/neurospin/highres-cortex/commit/a356d0826e1860faf85816da9b45b2bd0f20c04f>
fixes: #17763

Revision 80325 - 04/01/2018 04:59 PM - Riviere, Denis

i2bm-specific: set Boost_INCLUDE_DIR to /i2bm/brainvisa/<OS>/boost/include if found
with <https://github.com/neurospin/highres-cortex/commit/a356d0826e1860faf85816da9b45b2bd0f20c04f>
fixes: #17763

History

#1 - 04/01/2018 03:30 PM - Riviere, Denis

The problem is due (or we could say triggered) to multiple cmake options for boost.
In soma-io, it is used the default way:

```
find_package( Boost REQUIRED )
```

Then in highres-cortex, it is used differently:

```
# Do not use BoostConfig.cmake from boost-cmake, because its behaviour may be
# different from regular FindBoost.cmake.
set(Boost_NO_BOOST_CMAKE ON)
# All suitable Boost versions should be included here, if they are not already
# contained in the FindBoost.cmake module provided with CMake. Hence, new
# versions should be added here as they are released...
set(Boost_ADDITIONAL_VERSIONS
  "1.65.1" "1.65.0" "1.65"
  "1.64.0" "1.64" "1.63.0" "1.63" "1.62.0" "1.62" "1.61.0" "1.61"
  "1.60.0" "1.60" "1.59.0" "1.59" "1.58.0" "1.58" "1.57.0" "1.57"
  "1.56.0" "1.56" "1.55.0" "1.55" "1.54.0" "1.54" "1.53.0" "1.53"
  "1.52.0" "1.52" "1.51.0" "1.51" "1.50.0" "1.50" "1.49.0" "1.49")
find_package(Boost 1.49 REQUIRED)
```

It appears that setting Boost_NO_BOOST_CMAKE or Boost_ADDITIONAL_VERSIONS after find_boost() has already been called resets the paths in the cache. So we cannot specify one manually.

#2 - 04/01/2018 05:00 PM - Riviere, Denis

- *Status changed from New to Closed*

- *% Done changed from 0 to 100*

Appliqué par commit [brainvisa:r80325](#).