

---

Subject: Re: Cmake end with an error message.  
Posted by [Anonymous Poster](#) on Fri, 30 Nov 2007 11:49:21 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Florian,

Florian Uhlig wrote on Fri, 30 November 2007 09:07: Hi Andrey

That you already come up with a solution for the UNIQUE macro saves me a lot of work. Thanks for that.

You are always welcome.

Quote:

Your second suggestion is not needed.

The SET commando

```
SET (PATH ${ROOTSYS}/bin ${PATH})
```

adds \${ROOTSYS}/bin to the already existing \$PATH. The PATH is not the environment variable which separates the entries with a ":", but an internal list in cmake

I see the problem. If you are in the GSI environment your path is defined like this:

```
IF(DEFINED ENV{PATH})
  CLEAN_PATH_LIST("${ENV{PATH}}" "${KEYWORDS}" PATH)
ENDIF(DEFINED ENV{PATH})
```

where PATH variable really contains the list of the all entries in the \$ENV{PATH}. But if you are outside of the GSI the PATH variable is defined slightly different, namely:

```
ELSE(GSI)
  MESSAGE("-- You're not using the GSI installation of the external packages.")
  MESSAGE("-- If any problems occur this can be due to the fact that the macros")
  MESSAGE("-- were mostly tested at GSI. If you find any problems please contact")
  MESSAGE("-- f.uhlig@gsi.de")
  SET(PATH $ENV{PATH})
  #MESSAGE("PATH: ${PATH}")
ENDIF(GSI)
```

As you see now the PATH variable just contains the content of the \$ENV{PATH}. Please put the both methods into correspondence.

By the way, the macro CLEAN\_PATH\_LIST already contains the cast of UNIQUE macro, so you shouldn't do this second time in the WriteConfigFile.cmake

Best regards,

Andrey

---