
Subject: CINT -> g++

Posted by [Konstantin Antipin](#) on Wed, 20 Jun 2007 11:56:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi all,

I have a question concerning migration from CINT to g++.

Usually I am using root to load and execute macros with my tasks for analysing data. So, I am using cint interpreter.

But currently I am interested to make a stand-alone program and thus use g++ for compiling. So, I wrote a makefile (see attachment) and compiled my program. But when I am trying to execute it, it fails with segmentation error on loading libraries. I figured out, that the problem is in libParBase library in CbmRuntimeDb class. It fails when trying to access variable 'contFactories' (static TList) in function 'addContFactory'.

I can't find the reason why when I load this library under ROOT session all works perfectly.

So, if anybody has any suggestion how to make it work (if is it actually possible?), I would be very grateful.

Regards,
Konstantin

File Attachments

- 1) [Makefile](#), downloaded 891 times
 - 2) [test.C](#), downloaded 928 times
-

Subject: Re: CINT -> g++

Posted by [Konstantin Antipin](#) on Thu, 21 Jun 2007 12:17:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi all!

Due to Denis Bertini the problem is solved now.

The trouble was in the proper order of the loading library.

It is working now with the following order:

(-lSts -lGen -lRich -lTrd -lTof -lMvd -lGlobal -lKF -lDilept -lMuch -lBase -lField -lMCStack -lPassive -lGeoBase -lParBase)

Regards,
Konstantin

Subject: Why?

Posted by [Volker Friese](#) on Fri, 22 Jun 2007 17:37:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Nice to note it's working now, but is there an explanation why the order of the linked libraries

in the compile command matters at all?

Subject: Re: Why?

Posted by [Konstantin Antipin](#) on Fri, 22 Jun 2007 17:51:40 GMT

[View Forum Message](#) <> [Reply to Message](#)

Well, the problem probably is occurs due to the following:

There are some dependencies between this libraries. And when I am loading libraries not in the proper order - some of the static member are not initialized. And when on loading library constructor of this library is executed, segmentation error appears.

This is only guess - but it seems to be the true.
