

## Re: Linking a C++ library to a C program.

---

*Source:* [http://coding.derkeiler.com/Archive/C\\_CPP/comp.lang.c/2005-07/msg00938.html](http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2005-07/msg00938.html)

---

- *From:* "Default User" <[defaultuserbr@xxxxxxxxxx](mailto:defaultuserbr@xxxxxxxxxx)>
  - *Date:* 11 Jul 2005 16:03:48 -0700
- 

CBFalconer wrote:

> Malcolm wrote:

>> "a2x" <[add\\_aaron\\_2\\_x@xxxxxxxxxxxxx](mailto:add_aaron_2_x@xxxxxxxxxxxxx)> wrote

>>>

>>> I am writing a C program which interfaces with a C++ library. The C++ library works fine with C. I was wondering if I need to use a C++ compiler (g++ in my case) to compile the C program or can I use a C compiler (gcc).

>>

>> If you want to call a function written in C++ from a C file your best bet is to write the C code in the common subset of the two languages and run it through a C++ compiler. The alternative is to mess about with C++ name mangling and linkage conventions.

>

> For which alternative, AFAICT, there exists nothing even remotely resembling a standard, so that everything is totally non-portable.

Correct, but there is a C++ standard way to specify C-style linkage (extern "C"). That prevents you from calling overloaded functions from the C module, of course.

Whether a C object file can actually be linked with a C++ object file or library is still implementation dependent. In practice it works pretty well.

All of this should be discussed on comp.lang.c++, not here.

Brian

.

---

- *Follow-Ups:*

Re: Linking a C++ library to a C program.

◆ **Re: Linking a C++ library to a C program.**

◇ From: CBFalconer

• **References:**

◆ **Linking a C++ library to a C program.**

◇ From: a2x

◆ **Re: Linking a C++ library to a C program.**

◇ From: Malcolm

◆ **Re: Linking a C++ library to a C program.**

◇ From: CBFalconer

• Prev by Date: **Re: lint**

• Next by Date: **Re: Eigenvector routine....**

• Previous by thread: **Re: Linking a C++ library to a C program.**

• Next by thread: **Re: Linking a C++ library to a C program.**

• Index(es):

◆ **Date**

◆ **Thread**