
Subject: Re: Using zlib and IDL 6.1 in a DLM
Posted by [Karl\[1\]](#) on Fri, 23 Dec 2005 18:17:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Maurizio Tomasi wrote:

> Craig Markwardt wrote:
>> Maurizio Tomasi <tomasi@lambrate.inaf.it> writes:
>>> My question is: how can I tell my DLM to use my zlib library instead
>>> of the one provided by IDL?
>>
>> Static link instead of dynamic?
>>
>> Good luck!
>> Craig
>
> I tried this also (that is, adding "-static -lz" to the end of the line
> invoking the compiler), but the result is still the same:
>
> -----
> \$ gcc -c -I/opt/idl/idl/external/include test.c -fPIC -DPIC -o test.o
> \$ gcc -o test.so -shared test.o -L/opt/idl/idl/bin/bin.linux.x86/ -lidl\
> -static -lz
> \$ idl
> IDL Version 6.1 (linux x86 m32). (c) 2004, Research Systems, Inc.
> Trial version expires on 31-dec-2005.
> Licensed for personal use by INAF-Catania only.
> All other use is strictly prohibited.
> IDL> linkimage, 'PRINT_ZLIB_VERSION', 'test.so'
> IDL> print_zlib_version
> 1.1.4
> -----
>
> I ever tried to manually include every source file of the zlib into the
> source code of my app and then linking together all the object files,
> but (that *is* incredible) nothing changes!
>
> It begins to be a bit frustrating...
>
> Maurizio.

IDL is overriding the definitions in the shared library. This is normal linux linker/loader behavior. To stop this from happening, use -Bsymbolic.

I also found that I had to link the non-shared libz.

So, the linker command that worked for me is:

```
gcc -shared -o test.so test.o /usr/lib/libz.a -Wl,-Bsymbolic  
-L/tmp/idl/bin/bin.linux.x86 -lidl
```
