

---

Subject: Re: IDLWAVE and call-external  
Posted by [henrygroo](#) on Fri, 29 Sep 2006 16:47:50 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

My immediate guess is you have a path issue. To be safe I usually explicitly list the full path to a \*.so file in the call\_external.  
(Yes, this makes my code less portable, but I generally don't distribute \*.so's, so my laziness doesn't often come back to bite me.)

cheers,  
-Henry

Nicola wrote:

```
> Hi all
> I'm trying to use IDLWAVE (vers. 6.0 on Emacs 21.3.1) in debug mode.
> Actually I've have an interactive program (no widgets, just reading
> from the keyboard some variables) which works fine using a normal bash
> shell. Besides I use some extra routine which uses call_external to
> link some libraries (not written by me). The problem is that using the
> same programs in the idlwave-shell mode (C-c C-d C-c to easily use the
> tools for debugging given by IDLWAVE) I obtain this error
> CALL_EXTERNAL: Error loading sharable executable.
>           Symbol: ....., File = ..... .so
>           .... .so: cannot open shared object file: No such
>           file or directory
>
> (the dots replace the name of the library called).....
> I cannot understand the reason.....
> just for completness this is my customization for IDLWAVE
>
> ;*-----
> (autoload 'idlwave-mode "idlwave" "IDLWAVE Mode" t)
> (autoload 'idlwave-shell "idlw-shell" "IDLWAVE Shell" t)
> (setq auto-mode-alist (cons '("\\.pro\\\"" . idlwave-mode)
> auto-mode-alist))
> (setq idlwave-indent-parens-nested t)
> (setq idlwave-reserved-word-upcase t)
> (setq idlwave-main-block-indent 4)
> (setq idlwave-block-indent 4)
> (setq idlwave-end-offset -4)
> (setq idlwave-expand-generic-end t)
> (setq idlwave-fill-comment-line-only nil)
> (setq idlwave-indent-parens-nested t)
> (setq idlwave-use-last-hang-indent t)
> (setq idlwave-libinfo-file "~/.idlwave/libinfo.el")
> (setq idlwave-max-extra-continuation-indent 100)
> (setq idlwave-header-to-beginning-of-file t)
> (setq font-lock-maximum-decoration 4)
```

```
> (setq idlwave-show-block t)
> (setq idlwave-completion-case '((routine . preserve)
>   (keyword . upcase)
>   (class . preserve)
>   (method . preserve)))
> (setq paragraph-separate "[ \\t\\f]*$\\|\\[ \\t]*;+[ \\t]*$\\|;+[+=-_*]+$")
> (setq idlwave-auto-fill-mode 1)
> (setq idlwave-init-rinfo-when-idle-after 2)
> (setq idlwave-do-actions t
>   idlwave-surround-by-blank t)
> (setq idlwave-use-library-catalog t)
> (setq idlwave-shell-automatic-start t)
> (setq idlwave-reindent-end t)
> (setq idlwave-abbrev-mode t)
> ; personal documentation header
> (add-hook 'idlwave-mode-hook
>   (function
>     (lambda ()
>       (setq
>         idlwave-file-header
>         '("/home/vianello/.xemacs/header_idl_template_classic.txt")
>         )))
> (add-hook 'idlwave-load-hook
>   (lambda () (require 'idlw-complete-structtag)))
> ;*-----
>
>
> any suggestion is welcomed
> regards
> Nicola
```

---

---

Subject: Re: IDLWAVE and call-external  
Posted by [JD Smith](#) on Fri, 29 Sep 2006 18:36:20 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Fri, 29 Sep 2006 08:53:14 -0700, Nicola wrote:

```
> Hi all
> I'm trying to use IDLWAVE (vers. 6.0 on Emacs 21.3.1) in debug mode.
> Actually I've have an interactive program (no widgets, just reading
> from the keyboard some variables) which works fine using a normal bash
> shell. Besides I use some extra routine which uses call_external to
> link some libraries (not written by me). The problem is that using the
> same programs in the idlwave-shell mode (C-c C-d C-c to easily use the
> tools for debugging given by IDLWAVE) I obtain this error
> CALL_EXTERNAL: Error loading sharable executable.
> Symbol: ....., File = ..... .so
```

> .... .so: cannot open shared object file: No such  
> file or directory  
>  
> (the dots replace the name of the library called).....  
> I cannot understand the reason.....  
> just for completeness this is my customization for IDLWAVE

I'd guess paths as well. Are you relying on the .so file to be in the current path? IDLWAVE doesn't do anything to affect this, so it must be path setup. Try starting Emacs from a shell where it is working, and attempt again. It *should* work.

JD

---

---

Subject: Re: IDLWAVE and call-external  
Posted by [Nicola](#) on Mon, 02 Oct 2006 08:54:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

>  
> I'd guess paths as well. Are you relying on the .so file to be in the  
> current path? IDLWAVE doesn't do anything to affect this, so it must be  
> path setup. Try starting Emacs from a shell where it is working, and  
> attempt again. It *should* work.  
>  
> JD

You are right.....

Actually starting emacs from a shell all works fine whereas starting from start bar (KDE environment) does not work properly. I can't understand the reason (the library is included in my LD\_LIBRARY\_PATH in my .bash\_profile file) but maybe this is not more a question for this newsgroup (but if someone has some suggestion is welcomed!)

Regards  
n

---

---

Subject: Re: IDLWAVE and call-external  
Posted by [JD Smith](#) on Mon, 02 Oct 2006 16:58:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Mon, 02 Oct 2006 01:54:25 -0700, Nicola wrote:

>  
>> I'd guess paths as well. Are you relying on the .so file to be in the  
>> current path? IDLWAVE doesn't do anything to affect this, so it must be  
>> path setup. Try starting Emacs from a shell where it is working, and

>> attempt again. It *\*should\** work.

>>

>> JD

>

> You are right.....

> Actually starting emacs from a shell all works fine whereas starting from

> start bar (KDE environment) does not work properly. I can't understand the

> reason (the library is included in my LD\_LIBRARY\_PATH in my .bash\_profile

> file) but maybe this is not more a question for this newsgroup (but if

> someone has some suggestion is welcomed!) Regards

Because starting from the command bar doesn't inherit the shell environment variable LD\_LIBRARY\_PATH... it never even runs the shell, just the Emacs executable directly.

Try putting LD\_LIBRARY\_PATH in your .login and logging out and in. The entire session should inherit the variable then. You can always check within IDL like:

IDL> print,getenv('LD\_LIBRARY\_PATH')

JD

---