Subject: Re: call_procedure with a dynamically created arguments list?
Posted by news.qwest.net on Sat, 07 Oct 2006 21:41:39 GMT
View Forum Message <> Reply to Message

"Dometz" <dometz@gmail.com> wrote in message
news:1160175290.967818.184890@m73g2000cwd.googlegroups.com.. .

> Basically, all I am doing is:
> I automatically parse someones code and replace certain calls (ex:
> read_png) with wrapper functions. Inside of these wrapper functions, I
> record then timestamp when they were called, with what arguments etc.
> Now all this is supposed to happen transparent to the user. Basically,
> the user shouldnt even know that this happening or at least it shouldnt
> affect him.
>
> dometz

Well, the user is going to know, because the user will have to call the
wrapper.

The following is a horrible suggestion, I forbid you to do it. :)

but, why not just modify the actual function (if it is indeed read_png.pro
or any of the other functions with pro files). They are in the lib folder
of the IDL installation.
Just make sure that you take care when upgrading IDL versions.

I would make a CREATE_LOG_ENTRY procedure that does everything you want,
and paste it into each of the functions you wanted to wrap.

Inside of read_pict.pro
"PRO READ_PICT, filename, resultimage, r, g, b, DEBUG = DEBUG"

add a line like:
CREATE_LOG_ENTRY, 'Read_pict', filename, resultimage, r, g, b, debug

where your routine is something like

```
pro CREATE_LOG_ENTRY,
 routinename,generic1,generic2,generic3,generic4,generic5,gen eric6,generic7,generic8
openw, lun,'logfile'
printf,lun, routinename
printf,lun, n_elements(generic1)
printf,lun, n_elements(generic2)
printf,lun, n_elements(generic3)
etc.
```

Cheers,
bob