Subject: Re: IDL calls C calls IDL?
Posted by Joost Aan de Brugh on Thu, 11 Sep 2008 12:42:17 GMT
View Forum Message <> Reply to Message

On Sep 11, 11:53 am, "hotplainr...@gmail.com" <hotplainr...@gmail.com>
wrote:
> Hey again,
>
> I've been thinking, I have code that does
>
> LINKIMAGE 'func' ,.....
>
> for t=0,tmax-1
>   func(data)
>   idl_function(data)
> endfor
>
> Is it possible to call idl_function from within C ?
>
> I want it to be like this
>
> In IDL:
> func(data)
>
> Then it passes onto C which does
> for (t=0; t<tmax;t++)
> {
>   do whatever it does with data
>   CALL IDL_FUNCTION(data)
>
> }
>
> This eliminates me calling func a few million times
>
> Can it be done?

Hello,

And in the above example, IDL takes the initiative, calls func (which
is, if I am right, a C function) and does its own IDL function. (IDL
calls C a million times).

Below, C takes the initiative, creates a forloop, and in each
iteration it does something with data and then has IDL do something
with data by calling IDL_FUNCTION. (C calls IDL a million times).

Which of them is less nasty is not easy to say. To figure out how to
call IDL from C is better known by C-people. So that could better be

asked in C-groups.

I do not know what your functions do with the data. But if the order is not so important, you could try to do somthing like.

In C:
(t=0; t<tmax;t++)
{
  C-code(data);
}

and in IDL:

Call_C_Code(data)

; Now it's IDL time.
for t=0,tmax-1 do IDL_FUNCTION,data

But here, C does its whole task before IDL starts. This may be a problem depending on the actions. It may be possible that C can generate the actions that IDL can perform. For example:

In C:

(t=0; t<tmax;t++)
{
  C-code(data,actions); // Actions is an output by C passed back to
IDL. Data is left unharmed.
}

in IDL:

Call_C_Code(data,actions)

; Now it's IDL time.
for t=0,tmax-1 do begin
  IDL_PERMORM_ACTION,actions,data
  IDL_FUNCTION,data
End

Best regards,
Joost