Subject: IDL and C inter-language communication / call_external()
Posted by inkorporiert on Tue, 22 Jul 2014 19:34:44 GMT

Hey,
I have a plotting routine written in C that I call from IDL via call_external() command and this works pretty well. Now I want to plot the fit function with IDL (continously updated) while the external code is fitting.

I tried to call IDL_Execute() command from C, but this failed and then I read that this is only supported in "Callable IDL". But "Callable IDL" is not a solution because i need to pass data to the C routine from a IDL shell.

If there is no solution with call_external() to realize this: What would be the appropriate way to do this kind of bidirectional communication?

To sum it up:

- I want to pass data from a IDL shell to a C routine.
- Within the C routine I want to call IDL functions (plot) which make use of data generated by the C routine.

Thank you.

Subject: Re: IDL and C inter-language communication / call_external()
Posted by Craig Markwardt on Tue, 22 Jul 2014 21:35:23 GMT

On Tuesday, July 22, 2014 3:34:44 PM UTC-4, Frink3141 wrote:
> Hey,
>
> I have a plotting routine written in C that I call from IDL via call_external() command and this works pretty well. Now I want to plot the fit function with IDL (continously updated) while the external code is fitting.
>
>
>
> I tried to call IDL_Execute() command from C, but this failed and then I read that this is only supported in "Callable IDL". But "Callable IDL" is not a solution because i need to pass data to the C routine from a IDL shell.
>
>
>
> If there is no solution with call_external() to realize this: What would be the appropriate way to do this kind of bidirectional communication?
>
>

>
> To sum it up:
>
>
>
> - I want to pass data from a IDL shell to a C routine.
>
> - Within the C routine I want to call IDL functions (plot) which make use of data generated by the C routine.
>
>
>
> Thank you.

One way would be to have the C routine return occasionally with partial results and allow your IDL routine to plot them. That involves the least disruption and least potential for error.

Another possibility is to call your routine as a part of an external program. Then you could use some form of interprocess communication to communicate status and partial results. (In Unixy environments, you would just spit out some ASCII data to standard output.)

Craig

Subject: Re: IDL and C inter-language communication / call_external()
Posted by inkorporiert on Thu, 24 Jul 2014 09:08:53 GMT
View Forum Message <> Reply to Message

If I understand aright you mean that I call the routine via call_external() with initial value, then it does the first fitting step and ends returning the partial results, IDL plots and executes call_external() again with the last partial results and so on? Okay, this is a possibility, but would involve as far as I see a lot of changes in the C routine. Futhermore I think this would slow down everything a lot.

I didn't get what you mean by your second suggestion. The "external program" is a kind of deamon with the fitting routine waiting for input?

My primary idea was to allocate memory for the partial results with idl and pass that via call_external() to C. This way idl has access to the data and could plot it. But there are two problems: First I would have to create another thread for the plotting in IDL. But the major problem is the synchronisation: IDL should only plot after C has written the array and C should only write after IDL has read the array. This would require some kind of mutex-status variables that may be difficult to implement.

Thank you.

Subject: Re: IDL and C inter-language communication / call_external()
Posted by Craig Markwardt on Thu, 24 Jul 2014 22:46:01 GMT
View Forum Message <> Reply to Message

On Thursday, July 24, 2014 5:08:53 AM UTC-4, Frink3141 wrote:
> If I understand aright you mean that I call the routine via call_external() with initial value, then it does the first fitting step and ends returning the partial results, IDL plots and executes call_external() again with the last partial results and so on? Okay, this is a possibility, but would involve as far as I see a lot of changes in the C routine. Futhermore I think this would slow down everything a lot.
>
>
>
> I didn't get what you mean by your second suggestion. The "external program" is a kind of deamon with the fitting routine waiting for input?
>
>
>
> My primary idea was to allocate memory for the partial results with idl and pass that via call_external() to C. This way idl has access to the data and could plot it. But there are two problems: First I would have to create another thread for the plotting in IDL. But the major problem is the synchronisation: IDL should only plot after C has written the array and C should only write after IDL has read the array. This would require some kind of mutex-status variables that may be difficult to implement.

You really like to pour gasoline on the fire, don't you? :-) I wish you luck.

My comment about having an external program... At least on Unixy type computers, you can call
  SPAWN, 'myprogram', unit=unit
On the IDL side, you have a file UNIT you can read from using READF to monitor the progress of the task.

On the task side, you can print out some kind of progress indicator information that the IDL side can monitor to the standard output. (It is wise to use fflush() so that you don't run into buffering deadlocks.) It can be as simple as printing a "percent done" integer value. But it could be as sophisticated as printing out the numerical values of the fit after each iteration. It would be up to your IDL code to read the file UNIT to retrieve this information so in a timely way, and plot it or do whatever you want.

Criag