

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [David Fanning](#) on Thu, 13 Sep 2007 18:56:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Jim Brown writes:

> Our R&D group uses IDL and has written .pro scripts for use in  
> testing. Our test engineers use a Microsoft Windows desktop app  
> (developed in-house) but have to get out of the app, start IDL, and  
> then run a script. This happens several times during a test. What I'd  
> like to do is to have our testing app call an IDL API and run the  
> script "in the background" while the test engineer did something else.  
> However, I don't know if this can even be done. Most of the  
> documentation that I've been reading addresses calling an external  
> program from within IDL.

There is a reason for this, but it won't do much good  
to go into it. But keep in mind, IDL was never designed  
for this purpose. :-)

> - I want to do the opposite.

You want to know about "Callable IDL". You can read all  
about it in the External Development Guide, found among  
the other Programmer's Guides in your on-line IDL help.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

---

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [codepod](#) on Thu, 13 Sep 2007 20:34:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 13, 12:56 pm, David Fanning <da...@dfanning.com> wrote:

> Jim Brown writes:

>> Our R&D group uses IDL and has written .pro scripts for use in  
>> testing. Our test engineers use a Microsoft Windows desktop app  
>> (developed in-house) but have to get out of the app, start IDL, and  
>> then run a script. This happens several times during a test. What I'd  
>> like to do is to have our testing app call an IDL API and run the  
>> script "in the background" while the test engineer did something else.  
>> However, I don't know if this can even be done. Most of the

>> documentation that I've been reading addresses calling an external  
>> program from within IDL.  
>  
> There is a reason for this, but it won't do much good  
> to go into it. But keep in mind, IDL was never designed  
> for this purpose. :-)  
>  
>> - I want to do the opposite.  
>  
> You want to know about "Callable IDL". You can read all  
> about it in the External Development Guide, found among  
> the other Programmer's Guides in your on-line IDL help.  
>  
> Cheers,  
>  
> David  
> --  
> David Fanning, Ph.D.  
> Fanning Software Consulting, Inc.  
> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>

What you'll want to use is the IDL Connector objects, which were build for this exact scenario. They are part of IDL 6.4 (no extra license required). Built off the IDL Export Bridge technology, these are basic COM and Java objects that provide simple access to IDL (Execute String, Set/Get variables). This is much easier that Callable IDL, which is difficult to use. So if you can access COM or Java from your application, you could use this technology.

-CP

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [mankoff](#) on Thu, 13 Sep 2007 21:30:26 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 13, 2:56 pm, David Fanning <[da...@dfanning.com](mailto:da...@dfanning.com)> wrote:

> Jim Brown writes:  
>> Our R&D group uses IDL and has written .pro scripts for use in  
>> testing. Our test engineers use a Microsoft Windows desktop app  
>> (developed in-house) but have to get out of the app, start IDL, and  
>> then run a script. This happens several times during a test. What I'd  
>> like to do is to have our testing app call an IDL API and run the  
>> script "in the background" while the test engineer did something else.  
>> However, I don't know if this can even be done. Most of the  
>> documentation that I've been reading addresses calling an external  
>> program from within IDL.  
>

> There is a reason for this, but it won't do much good  
> to go into it. But keep in mind, IDL was never designed  
> for this purpose. :-)  
>  
>> - I want to do the opposite.  
>  
> You want to know about "Callable IDL". You can read all  
> about it in the External Development Guide, found among  
> the other Programmer's Guides in your on-line IDL help.  
>  
> Cheers,  
>  
> David  
> --  
> David Fanning, Ph.D.  
> Fanning Software Consulting, Inc.  
> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>

Wait... can't this be like a hundred times easier? On \*nix you can just execute from the command line "idl foo.pro" and foo will be executed in IDL. Doesn't windows have some way to run startup commands? I know everything is a bit harder to do in Windows, but I don't think callable IDL is the necessary solution to this. Sorry, I don't have IDL in Windows to test this on.

-k.

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [Mike\[2\]](#) on Fri, 14 Sep 2007 11:23:54 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 13, 5:30 pm, mankoff <mank...@gmail.com> wrote:  
> Wait... can't this be like a hundred times easier? On \*nix you can  
> just execute from the command line "idl foo.pro" and foo will be  
> executed in IDL. Doesn't windows have some way to run startup  
> commands? I know everything is a bit harder to do in Windows, but I  
> don't think callable IDL is the necessary solution to this. Sorry, I  
> don't have IDL in Windows to test this on.

I don't think that the problem is with windows itself. I'm sure there is a way to spawn or otherwise start something from within their desktop application (details depending on how it was written). The problem is that there is not a command-line idl on windows, so the "idl foo.pro" idiom is not available. To start IDL, they have to start the IDL IDE. On windows, the "I" really does mean interactive! That's one reason (among others) why we run all of our "batch" IDL codes on \*nix boxen.

Mike

---

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [mankoff](#) on Fri, 14 Sep 2007 14:12:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 14, 7:23 am, Mike <Michael.Mill...@gmail.com> wrote:

> On Sep 13, 5:30 pm, mankoff <mank...@gmail.com> wrote:

>

>> Wait... can't this be like a hundred times easier? On \*nix you can  
>> just execute from the command line "idl foo.pro" and foo will be  
>> executed in IDL. Doesn't windows have some way to run startup  
>> commands? I know everything is a bit harder to do in Windows, but I  
>> don't think callable IDL is the necessary solution to this. Sorry, I  
>> don't have IDL in Windows to test this on.

>

> I don't think that the problem is with windows itself. I'm sure there  
> is a way to spawn or otherwise start something from within their  
> desktop application (details depending on how it was written). The  
> problem is that there is not a command-line idl on windows, so the  
> "idl foo.pro" idiom is not available. To start IDL, they have to  
> start the IDL IDE. On windows, the "I" really does mean interactive!  
> That's one reason (among others) why we run all of our "batch" IDL  
> codes on \*nix boxen.

>

> Mike

I've never once run IDLDE, but I just launched it on OS X (unix), and under the startup tab in Preferences you can set a procedure. I assume you can do this on Windows too. That procedure should run whatever code the original poster wants to run. All they then need to do is launch the IDLDE.

The last line of the startup procedure could be "exit" to shut things down too. Of course, it would now be hard to use IDL for anything other than this one task, so I would write the startup procedure like so:

1) widget with 5 second countdown timer that alerts that this 'auto processing for Jim Brown' will be run in 5 seconds unless you click "no"

2) If "no" then drop into regular IDL/IDLDE mode

3) IF time times out, then run auto processing, and exit (shutting down the IDLDE)

-k.

---

---

Subject: Re: Running IDL .pro script from another app, using the IDL API

Posted by [Jeff N.](#) on Fri, 14 Sep 2007 14:33:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 14, 10:12 am, mankoff <mank...@gmail.com> wrote:

> On Sep 14, 7:23 am, Mike <Michael.Mill...@gmail.com> wrote:

>

>

>

>> On Sep 13, 5:30 pm, mankoff <mank...@gmail.com> wrote:

>

>>> Wait... can't this be like a hundred times easier? On \*nix you can  
>>> just execute from the command line "idl foo.pro" and foo will be  
>>> executed in IDL. Doesn't windows have some way to run startup  
>>> commands? I know everything is a bit harder to do in Windows, but I  
>>> don't think callable IDL is the necessary solution to this. Sorry, I  
>>> don't have IDL in Windows to test this on.

>

>> I don't think that the problem is with windows itself. I'm sure there  
>> is a way to spawn or otherwise start something from within their  
>> desktop application (details depending on how it was written). The  
>> problem is that there is not a command-line idl on windows, so the  
>> "idl foo.pro" idiom is not available. To start IDL, they have to  
>> start the IDL IDE. On windows, the "I" really does mean interactive!  
>> That's one reason (among others) why we run all of our "batch" IDL  
>> codes on \*nix boxen.

>

>> Mike

>

> I've never once run IDLDE, but I just launched it on OS X (unix), and  
> under the startup tab in Preferences you can set a procedure. I assume  
> you can do this on Windows too. That procedure should run whatever  
> code the original poster wants to run. All they then need to do is  
> launch the IDLDE.

>

> The last line of the startup procedure could be "exit" to shut things  
> down too. Of course, it would now be hard to use IDL for anything  
> other than this one task, so I would write the startup procedure like  
> so:

>

> 1) widget with 5 second countdown timer that alerts that this 'auto  
> processing for Jim Brown' will be run in 5 seconds unless you click  
> "no"

>

> 2) If "no" then drop into regular IDL/IDLDE mode  
>  
> 3) IF time times out, then run auto processing, and exit (shutting  
> down the IDLDE)  
>  
> -k.

Well, two things come to mind: 1) Remember that the startup procedure can't be any old idl procedure: it has to be "batch jobs" like you would run with the @ symbol, or enter yourself at the command line (no multiple line statements). 2) What you *can* do is give the idlde some command line arguments using the -args switch. During my dissertation I used these, for example, to set my path preferences based on whether or not i had my external hard drive hooked up on my desktop, which had lots of data i'd need. You can parse whatever's passed via -args with the `COMMAND_LINE_ARGS` function.

What i think i would do is this: when i want idl to get started and do some automatic processing, i'd start it with an -args switch. I'd set up my startup file to look for that switch, and if it finds it, do the automatic processing and then quit. If that switch isn't found, start normally.

Jeff

---

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [Maarten\[1\]](#) on Fri, 14 Sep 2007 15:20:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 14, 11:23 am, Mike <Michael.Mill...@gmail.com> wrote:  
> I don't think that the problem is with windows itself. I'm sure there  
> is a way to spawn or otherwise start something from within their  
> desktop application (details depending on how it was written). The  
> problem is that there is not a command-line idl on windows, so the  
> "idl foo.pro" idiom is not available. To start IDL, they have to  
> start the IDL IDE. On windows, the "I" really does mean interactive!  
> That's one reason (among others) why we run all of our "batch" IDL  
> codes on \*nix boxen.

I don't use Windows myself, but a tool I wrote to analyse OMI level 2 data runs on both Linux and Windows (and presumably Mac OS X) from the command-line, in batch mode. My coworker wrote part of the manual of this tool with instructions on how to run this tool in batch mode:

(quote from manual)

For processing jobs involving many different operations on an OMI data set one can write an IDL program to instruct IDL to call the CAMA

routine a number of times with each time a different "description file". However, to avoid running into the before mentioned caveat, IDL should be reset in between these calls and unfortunately this can only be performed manually in IDL on Windows. Advantageously, IDL on Windows allows for calling the IDLDE.EXE program directly from a DOS prompt and from a DOS batch file. For example, the file "mass\_run.bat" contains lines of the following form:

```
"C:\RSIIDL\IDL61\bin\bin.x86\idlde.exe" @E:\Calculations\IDL\CAMA
\WholeGlobe\WG_mass_run_verigen_12.pro
```

where "C:\RSIIDL\IDL61\bin\bin.x86\idlde.exe" specifies the location of the local installation of the idlde.exe program and the text directly following the "@" sign specifies the location of a .pro file. In this .pro file regular CAMA calling routines can be written such a specified above. To achieve resetting IDLDE between parts of the processing job, the last line of code of the .pro file should read "EXIT" forcing IDL to exit. Subsequently, the next line of command in the batch file restarts IDLDE with the default setting and all counters reset to zero. In this way a large number of processing jobs can be programmed to conveniently run outside office hours. Please do make absolutely sure that the option "Confirm Exit" in section "General" of "Preferences" is turned off, otherwise IDL on Windows requires a manual confirmation from the user to exit.  
(end quote)

To me this means that batch processing is possible, but the commands need to be specified in a batch file (no multi-line commands, just like IDL on \*nix). To fully run without user interaction some setting must be changed ("Confirm exit"). The issue here is that processing more than 600 hdf files sequentially causes some unresolved buffer overruns (not present on Linux), hence the exit in between different batches.

HTH,

Maarten

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [mankoff](#) on Fri, 14 Sep 2007 17:06:47 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Sep 14, 10:33 am, "Jeff N." <jnett...@utk.edu> wrote:  
> On Sep 14, 10:12 am, mankoff <mank...@gmail.com> wrote:  
>  
>  
>



>> On Sep 14, 7:23 am, Mike <Michael.Mill...@gmail.com> wrote:  
>  
>>> On Sep 13, 5:30 pm, mankoff <mank...@gmail.com> wrote:  
>  
>>>> Wait... can't this be like a hundred times easier? On \*nix you can  
>>>> just execute from the command line "idl foo.pro" and foo will be  
>>>> executed in IDL. Doesn't windows have some way to run startup  
>>>> commands? I know everything is a bit harder to do in Windows, but I  
>>>> don't think callable IDL is the necessary solution to this. Sorry, I  
>>>> don't have IDL in Windows to test this on.  
>  
>>> I don't think that the problem is with windows itself. I'm sure there  
>>> is a way to spawn or otherwise start something from within their  
>>> desktop application (details depending on how it was written). The  
>>> problem is that there is not a command-line idl on windows, so the  
>>> "idl foo.pro" idiom is not available. To start IDL, they have to  
>>> start the IDL IDE. On windows, the "I" really does mean interactive!  
>>> That's one reason (among others) why we run all of our "batch" IDL  
>>> codes on \*nix boxen.  
>  
>>> Mike  
>  
>> I've never once run IDLDE, but I just launched it on OS X (unix), and  
>> under the startup tab in Preferences you can set a procedure. I assume  
>> you can do this on Windows too. That procedure should run whatever  
>> code the original poster wants to run. All they then need to do is  
>> launch the IDLDE.  
>  
>> The last line of the startup procedure could be "exit" to shut things  
>> down too. Of course, it would now be hard to use IDL for anything  
>> other than this one task, so I would write the startup procedure like  
>> so:  
>  
>> 1) widget with 5 second countdown timer that alerts that this 'auto  
>> processing for Jim Brown' will be run in 5 seconds unless you click  
>> "no"  
>  
>> 2) If "no" then drop into regular IDL/IDLDE mode  
>  
>> 3) IF time times out, then run auto processing, and exit (shutting  
>> down the IDLDE)  
>  
>> -k.  
>  
> Well, two things come to mind: 1) Remember that the startup procedure  
> can't be any old idl procedure: it has to be "batch jobs" like you  
> would run with the @ symbol, or enter yourself at the command line (no  
> multiple line statements). 2) What you \*can\* do is give the idlde



> some command line arguments using the -args switch. During my  
> dissertation I used these, for example, to set my path preferences  
> based on whether or not i had my external hard drive hooked up on my  
> desktop, which had lots of data i'd need. You can parse whatever's  
> passed via -args with the `COMMAND_LINE_ARGS` function.  
>  
> What i think i would do is this: when i want idl to get started and  
> do some automatic processing, i'd start it with an -args switch. I'd  
> set up my startup file to look for that switch, and if it finds it, do  
> the automatic processing and then quit. If that switch isn't found,  
> start normally.  
>  
> Jeff

Yes all those caveats apply. But presumably the startup.pro (which is limited to batch) would be something like:

```
startup_real_procedure_abilities_go_here.pro  
exit
```

and then you can put whatever you want into the startup\_real....pro procedure.

Anyway, I think it is pretty clear from this thread that windows and batch jobs can do what was requested without resorting to advanced programming techniques. Its up to Jim Brown which approach to take now...

-k.

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [Jim Brown](#) on Fri, 14 Sep 2007 20:22:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello and thank you for your time and effort and for your replies.

First, the fact that I have never used IDL before limits me somewhat, but I am a pretty competent software engineer, so I've made some good progress.

I don't have access to IDL 6.4, and am stuck with 6.1 for now. However, I was able to implement a connection using Callable IDL from a Visual Studios VC++ project. The connection uses a callback function registered with `IDL_ToutPush` to display messages from IDL onto the GUI's text box. After initializing IDL with `IDL_Win32Init`, I pass the name of the .pro script via a call to `IDL_ExecuteStr`. IDL parses the

script correctly, but I immediately run into two problems.

First, the script calls an IDL function "read\_dicom" but Callable IDL fails and sends me a message "Variable is undefined: READ\_DICOM." The script runs just fine from the IDL IDE, so I'm thinking that something is not set right. I found the file idl\_dicom.dlm in the IDL bin directory, and set the environment variable IDL\_DLM\_PATH to the location of the .dlm files, but this made no difference. Does anyone have any insight into this?

The second problem is worse. The original scripts use read functions, and Callable IDL tells me that it can't open stdin. Is there any way to interact with the running script, or pass parameters to the script in my call to IDL\_ExecuteStr?

Thank you for your help.

Sincerely,  
Jim Brown

---

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [Jim Brown](#) on Fri, 14 Sep 2007 20:27:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I solved the first problem - I had to add C:\RS\IDL61\lib to IDL\_PATH. It works fine.

Now, if I could just solve the I/O problem...

---

---

Subject: Re: Running IDL .pro script from another app, using the IDL API  
Posted by [R.Bauer](#) on Wed, 19 Sep 2007 17:42:36 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Jim Brown wrote:

> Hi,  
>  
> Our R&D group uses IDL and has written .pro scripts for use in  
> testing. Our test engineers use a Microsoft Windows desktop app  
> (developed in-house) but have to get out of the app, start IDL, and  
> then run a script. This happens several times during a test. What I'd  
> like to do is to have our testing app call an IDL API and run the  
> script "in the background" while the test engineer did something else.  
> However, I don't know if this can even be done. Most of the  
> documentation that I've been reading addresses calling an external  
> program from within IDL - I want to do the opposite.

>  
> Does anyone have any experience doing this? Just point me in the right  
> direction if you can.  
>  
> Thank you.  
> - Jim Brown  
>

Hi Jim

this can be done you can have a look at the

compile or resolve\_dependencies routine

from our lib at

[http://www.fz-juelich.de/icg/icg-i/idl\\_icglib/idl\\_lib\\_intro.html](http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html)

which do call another idl session.

cheers

Reimar

--

Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-1)

Forschungszentrum Juelich

email: R.Bauer@fz-juelich.de

-----

a IDL library at ForschungsZentrum Juelich

[http://www.fz-juelich.de/icg/icg-i/idl\\_icglib/idl\\_lib\\_intro.html](http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html)

=====

---