
Subject: Calling a Python script from IDL
Posted by [lwelser](#) on Wed, 07 Jul 2004 16:45:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Is there a way to call a Python script from within IDL and have it return information to the IDL program? (Python is an object-oriented scripting language similar to perl and java). I'm working on visualizing huge amounts of data in an IDL widget, and a python script to parse through this data is already in existence.

Thanks,
Leslie Welser

Subject: Re: Calling a Python script from IDL
Posted by [cedricl](#) on Sun, 11 Jul 2004 18:03:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

It depends on how involved your script is. If you're on Unix, you could maybe use a Unix pipe with the IDL command "spawn". Something like "spawn, 'python myscript.py'". Look at the external development guide.

I have never used a Unix pipe myself, so I'm not sure if they can support "huge amounts of data", or how a Python program can manage this data coming from the standard input...

Cedric

lwelser@physics.unr.edu (Leslie Welser) wrote in message news:<c870f808.0407070845.74d8e68c@posting.google.com>...

> Is there a way to call a Python script from within IDL and have it
> return information to the IDL program? (Python is an object-oriented
> scripting language similar to perl and java). I'm working on
> visualizing huge amounts of data in an IDL widget, and a python script
> to parse through this data is already in existence.
> Thanks,
> Leslie Welser

Subject: Re: Calling a Python script from IDL
Posted by [Michael Wallace](#) on Sun, 11 Jul 2004 22:54:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

>> Is there a way to call a Python script from within IDL and have it
>> return information to the IDL program? (Python is an object-oriented
>> scripting language similar to perl and java). I'm working on
>> visualizing huge amounts of data in an IDL widget, and a python script

>> to parse through this data is already in existence.
>> Thanks,
>> Leslie Welser

You could open a python process with SPAWN and then use shared memory to send data between the processes. Since you're working on "huge amounts of data," shared memory would probably be the easiest way to go. The gotcha to this is that you will probably have to be careful how you read/write data to the shared memory to make sure that values are converted between IDL and Python correctly. Don't know if that helps or not. Happy trails,

-Mike
