Subject: Re: VM command line parameters
Posted by greg michael on Fri, 11 Aug 2006 11:33:09 GMT

Seems I didn't word that question so well...

I have a function like this:

 age_fit,crater_count_file,polynom_file,poly_index,chron_file ,chron_index,fit_range

which outputs text a file with some values for an external program. I'd
like the user to be able to call this routine through the VM to get the
output file.

So is there a way to type something like:

idl -vm=age_fit.sav
 crater_count_file,polynom_file,poly_index,chron_file,chron_i ndex,fit_range

?

Otherwise I suppose I can put them in a file and use David's
programrootdir to get them out.

thanks for any suggestions,
Greg

greg michael wrote:
> Is it possible to pass command line parameters to a virtual machine
> routine? What's the syntax for that?
>
> regards,
> Greg

Subject: Re: VM command line parameters
Posted by btt on Fri, 11 Aug 2006 12:47:16 GMT

greg michael wrote:
> Seems I didn't word that question so well...
>
> I have a function like this:
>
> age_fit,crater_count_file,polynom_file,poly_index,chron_file ,chron_index,fit_range
>
> which outputs text a file with some values for an external program. I'd

> like the user to be able to call this routine through the VM to get the
> output file.
>
> So is there a way to type something like:
>
> idl -vm=age_fit.sav
> crater_count_file,polynom_file,poly_index,chron_file,chron_i ndex,fit_range
>
> ?
>
> Otherwise I suppose I can put them in a file and use David's
> programrootdir to get them out.
>
Hi,

You might look into the newish command line switches ...

IDL> ?command line switches

I haven't tried these with a VM but it seems like a nice fit.

Ben

Subject: Re: VM command line parameters
Posted by greg michael on Fri, 11 Aug 2006 15:02:54 GMT
View Forum Message <> Reply to Message

Thanks very much, Ben - that works. You have to write an extra section
to decode the arguments from strings, but it's way better than using a
file. Works like this, where age_fit0 is the original routine:

```
idl -vm=age_fit.sav -args crater_count_file polynom_file poly_index
chron_file chron_index fit_range

pro age_fit
 cla=command_line_args(count=count)
 crater_count_file=cla[0]
 polynom_file=cla[1]
 poly_index=fix(cla[2])
 chron_file=cla[3]
 chron_index=fix(cla[4])
 fit_range=float([cla[5],cla[6]])

 age_fit0,crater_count_file,polynom_file,poly_index,chron_fil e,chron_index,fit_range
 end
```