Subject: Re: SAV file generation with command line arguments
Posted by justspam03 on Wed, 19 Nov 2003 10:05:24 GMT
View Forum Message <> Reply to Message

Another possibility, which might be an option for batch
processing, is to use environment variables.
a la

```
exe='/x/y/z/idlrt.exe studyeval.sav'
export study="study1"
eval $exe
export study="study2"
eval $exe
...
```

and in studyeval.pro

```
; setenv, 'study=study1'    ; for testing
study = getenv('study')
```

For user interaction this approach might be a bit cumbersome, though.
Cheers
Oliver

David  Fanning <david@dfanning.com> wrote in message
news:<MPG.1a24aa6a6984db23989744@news.frii.com>...
> Sangwoo writes:
>
>> I have a simple question. Is it possible to create a .sav application
>> from a procedure including command line arguments? I know that making
>> a .sav file from a procedure like below is absolutely possible :
>>
>> pro example
>> .....
>> end
>>
>> But, isn't it possible from a procedure like below? :
>>
>> pro example, a, b

>> .....
>> end
>>
>> I know that IDL VM doesn't allow access to command line. Does it mean
>> the answer to my question is negative?
>
> No, it means the answer to your question is complicated. :-)
>
> It is true that you can't pass command line arguments
> to IDL save files. (There is no, uh, command line.)
> But that is not to say you can't write procedures
> with arguments and save them as save files. It's just
> that those arguments can't be there when the file is
> run. If you need the arguments, you have to work out
> how to get them. In other words, you have to treat them
> as optional arguments.
>
> One option is to open a dialog and allow the
> user to enter the arguments. Another option is
> to read the arguments out of a data file (which
> some people create with a shell script so that
> their program *appears* to accept command line
> arguments.
>
> Rob Dimeo brought a nifty little program to my
> attention this week that was written by Jim
> Pendleton, an RSI programmer who writes awfully
> clever IDL code, that may help with the latter.
> The program is named SOURCEROOT. It allows you
> to store data files in the same directory as
> your pro files, and then distribute the files
> to any location. The SOURCEROOT program figures
> out which directory the pro files are being
> run from, so it can locate the data files
> without having to know ahead of time where
> they are. Neat. It works like this:
>
>  file = filepath('data.dat',root = sourceroot())
>
> You can find the program here:
>
>  http://www.rsinc.com/codebank/search.asp?FID=35
>
> There are probably other solutions as well. But you
> have to be pretty creative to discover some of them. :-)
>
> Cheers,
>

> David