Subject: Re: TRICKY TASK USING AWK
Posted by dg86 on Fri, 26 Feb 2016 12:03:57 GMT
View Forum Message <> Reply to Message

On Friday, February 26, 2016 at 4:20:20 AM UTC-5, Sapna Mishra wrote:
> Hello all,
>
> What I have is:
>
> IDL> i=1
> IDL> $ awk -v var="$i" '{print $var}' tmp1.dat> tmp3.dat
>
> Upto this it is working ok. now what I want:
>
> IDL> fname[i]='QB0332_CHIP1_GRIS_600V_SCIENCE_SKY_SPECTRUM_FORS1.
2007_11_11T06_43_02.032redwave.dat'
> IDL> $ awk -v var="$i" '{print $var}' fname[i]> fnameout[i]
>
> Can I do it some how??? I found it most tricky. Basically can we use variables defined inside idl into shell commands.
>
> Anyone know how to solve this tricky job???

Without delving too deeply into the details of your specific application, it sounds like you want to build up a string containing the desired command, and then to use SPAWN to execute that command. The general idea is

IDL> cmd = 'awk my_awk_command ' + fname[i] + ' > ' + fnameout[i]
IDL> spawn, cmd

Note that you'll have to be careful about the quotation marks in the IDL string that I've called 'cmd'. To get this right, you'll have to read the documentation on IDL strings, specifically how to escape special characters (such as quotation marks).

You can double-check that you've built up the command properly by printing the string:

IDL> print, cmd

This should print exactly the command that you'd like to execute.

Finally, you'll have to be careful to insert spaces between the substrings that make up your command so that the result is syntactically correct for execution by the shell. In my sketched solution, I've deliberately placed a space after the placeholder pseudocode 'my_awk_command'.

All the best,

David