Subject: Re: Running IDL from cron
Posted by Vapuser on Fri, 23 Feb 2001 21:06:59 GMT
View Forum Message <> Reply to Message

Gwyn Fireman <Gwyn.Fireman@gsfc.nasa.gov> writes:

> Hi, everyone -
>
> With Deja gone and Google barely functional I have to ask here what I'm
> sure has been addressed before:
>
> How do you run IDL routines from cron - while maintaining IDL_PATH and
> other environment variables?
>
> I have had no trouble running my idl routines using batch or at, when
> starting in the same directory as the main-level routine. In that case
> my environment is copied to the spawned process.
>
> Now I have the need to run these from cron. I have been able to write
> scripts to cd to the appropriate directory and to run my .cshrc (where
> my IDL_PATH and other environment variables are defined), but keep
> running into problems. The latest is than .cshrc bombs on a
> system-level env that apparently isn't defined when running from cron.
>
> So can anyone point me in the right direction?
> I'm running { alpha OSF unix 5.3 Nov 11 1999}.
>
> Many thanks in advance,
>   Gwyn Fireman

Cron uses a crippled environment in the Bourne shell, it defines
only a limited number of environmental variables.

from the crontab manpage

   The shell is invoked from your $HOME directory with an arg0 of sh. Users
   who desire to have their .profile executed must explicitly do so in the
   crontab file. Cron supplies a default environment for every shell,
   defining HOME, LOGNAME, USER, SHELL(=/bin/sh),
   PATH(=/usr/sbin:/usr/bsd/:/usr/bin:/bin/:/etc:/usr/etc), and TZ.

You have to explicitly source whatever files you need to define the
correct env. Usually, one does this by sourcing their .profile as
the first part of a multi-part command, but you can source any file
if you want to restrict the cronjob environment in some way. To
execute two or more commands in the same shell, you put them inside

parantheses separated by a semi-colon. When run the first command ensures the correct environment for the second.

Like this.

0 * * * * (. /path/to/rc/file; /path/to/script) > some-file 2>&1;

(Note the `.' there!)

If your .profile sets up your environment and you don't think you need to restrict the cron job in anyway, simply source it as the first part of each crontab entry. e.g.

0 * * * * (. $HOME/.profile ; /path/to/script) > some-file 2>&1;

(`some-file' can be /dev/null if you desire. If nothing is there and there is any output to stdout/stderr, crontab mails it to you)

If you're using the csh, you can convert all the commands to their Bourne shell equivalents. I think you'll find this easier in the long run than trying to figure out how to correctly configure a Bourne shell using Csh configuration files.

This does *not* mean, by the way, that you have to convert *all* the scripts you call to any particular shell! All my cron jobs call perl scripts as the second command but they could just as easily run csh scripts. All that's being done with the first script is to configure the environment.

If you need help with the differences between Bourne and CSH configuration syntax, I'm sure there's someone around you who can help.

All of my automated processing for my web page (http://haifung.jpl.nasa.gov) is done by cronjobs with a combination of perl scripts calling IDL using temporary files constructed on the fly by the perl scripts. I just source $HOME/.profile, which then sources other .rc files, as the first thing in each crontab entry. By the way, as my native shell, I run the tcsh, so I'm sympathetic to csh users.

The only other problem is the matter of the display, which you can get around, as Liam(?) pointed out, by using the Z buffer. No problem if you're doing gifs or grayscale jpegs. But it's an inconvenient but not insurmountable difficulty if you're doing true-color graphics. The solution is to operate on the indivudual color planes individually in the Z buffer and then assemble the image at the end

and write it out. It's harder to do, but it isn't impossible. This
is precisely how I make the truecolor jpegs on haifung.

In fact, were it not for a problem caused by an SGI utility, which
is necessary to my processing, I wouldn't even need to be logged
into the console for it to run. As it is, I have to be or the
animation processing will timeout waiting for a connection to the X
server when this SGI utility (dmconvert) goes to convert the 60 .gif
files into a quicktime movie. The words I spoke when I found out
that despite spending a week ridding my IDL code of all connections
to the X server only to discover that it was the blackbox of the SGI
utility that was the fatal stumbling block to my goal of complete
'X' independence are not reproducible in polite company.

whd
--
William Daffer: 818-354-0161: William.Daffer@jpl.nasa.gov