hi, sorry about the confusion. I don't know why the same post got posted again. What I just typed was not lost.

Thanks Mike for your reply. Following is the postscript printing code I used. Since I am continuing on someone else's code, as far as I understood, the "owindow" variable is the IDL number of the graphic window the 3D image is drawing on. The 3D image could be rotated as wished in the window. The "oview" is an IDLgrView structure. From what I realized, the output is already in a format of an image ("im3d" variable) even before it was printed as a PS. please let me know if I need to post a more detailed documentation of the code. Thank you very much for your time. I appreciate your help.

partial code:
```
   sState.oVols[sState.iVrendVol]->SetProperty, $
     HIDE=1-vol_rend
   demo_draw, sState.oWindow, sState.oView, debug=sState.debug
   sState.oVols[sState.iVrendVol]->SetProperty,HIDE=1
   oImage = sState.oWindow->Read()
   oImage->GetProperty,DATA=data
   OBJ_DESTROY,oImage

   im3d=data

!P.FONT=0

; Set the plotting device to PostScript:
SET_PLOT, 'ps'

LOADCT, 0

; Write the image to the file:
;TV, [[[r]], [[g]], [[b]]], TRUE=3

TVSCL, im3d, TRUE=1
; Close the file:
DEVICE, /CLOSE
set_plot,'win'
```