Subject: make_rt change viewplane behaviour
Posted by LNpellen on Wed, 28 Jan 2015 12:43:24 GMT
View Forum Message <> Reply to Message

I have made an executable of my application by using make_rt.
I'm currently stuck with v 7.1 - sorry for asking questions that might be biased by such an old version, but I think I have had similar problems with newer version at my previous employer too.

In the Application I am opening images and by default the viewplane is set equal to the image size.

imgSz=SIZE(activeImg, /DIMENSIONS)

viewpl=[0. ,0. , imgSz[0:1]]
oView=OBJ_NEW('IDLgrView', VIEWPLANE_RECT =viewpl)

I also have a label showing the cursor position. DrawXY is the size of the (quadratic) WIDGET_DRAW.

xx=ev.X*imgSz(0)/drawXY
yy = ev.Y*imgSz(1)/drawXY

When I run this application in "debug mode" in the IDL Workbench on an XP machine (Remote) the cursor value (xx,yy) displays 510,510.
When I run the make_rt executable version (not remote) on an Win 7 machine the same image is open, but the image is all black, probably because it is only showing a part of the image up to 64,64 pixels as this is what the cursor display tells me. Also the position of the annotations on the image is stretched.

So what is happening with the viewplane between the workbench and the make_rt executable?

Any ideas? Will it automagically be fixed when I hopefulle proceed to version 8.x?

Best regards
Ellen

Subject: Re: make_rt change viewplane behaviour
Posted by LNpellen on Mon, 02 Feb 2015 08:37:28 GMT
View Forum Message <> Reply to Message

It turned out after some tedious testing debugging in the exe-version via DIALOG_MESSAGES that READ_DICOM works differently in the workbench than in the exe-version. I can't see why, but I realized how it behaved differently and found a trick to force the reading to be similar for both (by avoiding the READ_DICOM function).

So it had nothing to do with the VIEWPLANE_RECT (I realize I should have understood that from start, but I blamed this parameter for xenophobic reasons).

Changing
matrix=READ_DICOM(adr)

To
o=obj_new('idlffdicom')
t=o->read(adr)
test=o->GetReference('7FE0'x,'0010'x)
test_peker=o->GetValue(REFERENCE=test[N_ELEMENTS(test)-1],/NO_COPY)
matrix=FLOAT(*(test_peker[0]))

Problem is that READ_DICOM (in the exe-version, not in workbench) takes the first image it finds in the dicom file, which is a icon-image of 64x64. Now I force matrix to read the last instance of image data. Hope the guess that the real image always is the last one holds.