Posted by btupper on Wed, 23 Oct 2002 13:53:10 GMT
View Forum Message <> Reply to Message

Hello all,

I enjoy working with objects graphics when I can... until somebody
wants a printout. Then I hate the dang stuff. It *always* seems
like such a tinker-with-it-until-it-works experience. Hmph!

I have been daydreaming (again) about a generic tool much like
FSC_PSCONFIG, that is, an interactive tool for placing/sizing object
graphics on a printed page.

I gather from the object printing examples in 'What's New in IDL 5.5'
the only things that need to be modified are

(1) the LOCATION property of the VIEW(s) to be drawn

and

(2) the SCALE properites of any of the MODELS contained in the
VIEW(s).

It appears that one only needs to ascertain and save the initial
properties of the view(s) and the models contained in the view(s).
Then, relocate the views and rescale the models for the printer.
Next, invoke the printer's draw method. Finally, restore the original
properties of the view(s) and models. Piece of cake, maybe.

I find the example a bit hard to fathom because information about the
object graphics window is used to scale/locate the object graphics
onto the printer page. It makes it seem complicated and a bit
adhoc... precisely my own experience.

My questions are...

(1) is it possible to properly size/place object graphics on a printer
page given *only* the VIEW(s) and the PRINTER object?

(2) If it is possible to size/locate object graphics in a general way,
can the concept be 'enlarged' to encompass any destination device
(like the clipboard or buffer)?

Thanks!
Ben

Subject: Re: Object Graphics Printing
Posted by David Fanning on Tue, 29 Oct 2002 02:57:15 GMT

Ben Tupper (btupper@bigelow.org) writes:

> The tricky part with this is dealing with the default VIEW dimensions.
> In this case, the programmer has allowed the VIEW dimensions to fit
> the destination device (usually a window). A call to the VIEW's
> getproperty method returns [0,0] for the dimensions of the view.
> Oops, what kind of aspect is that?   There's nothing to be done in
> that case, except match the VIEW's dimensions to the DESTINATION (in
> my case the printer - which doesn't have the same aspect ratio as the
> VIEW.)

Yes, that's a problem, but I guess you could always
argue that the user must have read the documentation
and decided this is what they had in mind. :-)

I haven't run your program yet (still fixing bad line
breaks caused by a 1960's era newsreader that stubbornly
insists on fitting everything for output to 80-column
punch cards), but have you taken into account that a
location of (0,0) puts part of the output in the
non-printable portion of the printer?

I've always found that a decidedly non-general printer
fudge factor is also required. :-(

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Object Graphics Printing

Ben Tupper (btupper@bigelow.org) writes:

> ;+
> ; NAME:
> ; FITASPECT

And then,

> FUNCTION Fit_Aspect, view, dest, $
>   margin = margin, Location = Location, $
>   Units = Units

I'm sure this is one of those "you say to-may-toe
and I say to-mah-toe" kinds of things, but we probably
ought to decide which it is.

I wouldn't even bring the subject up, but I hear that
a recent candidate for a job at Research Systems was
accused of having "a programming style like Fanning".
Let's just say it wasn't immediately obviously if this
was a complement or not. :-(

Apparently the distinction hinged on whether it is
better to use capital letters in a natural and eye-
pleasing way, or to sprinkle underscores all over
God's creation. We'll let Ben have it both ways,
since we don't want to confuse him. But what say
the rest of you?

Cheers,

David

P.S. Let's just say this is just the thing to stir
up the juices ahead of the big IEPA dinner on
the 31st! :-)

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

On Mon, 28 Oct 2002 20:22:47 -0700, David Fanning wrote:

> Ben Tupper (btupper@bigelow.org) writes:
>
>> ;+
>> ; NAME:
>> ; FITASPECT
>
> And then,
>
>> FUNCTION Fit_Aspect, view, dest, $
>>   margin = margin, Location = Location, $ Units = Units

> Apparently the distinction hinged on whether it is better to use capital
> letters in a natural and eye- pleasing way, or to sprinkle underscores
> all over God's creation. We'll let Ben have it both ways, since we don't
> want to confuse him. But what say the rest of you?

Interestingly, almost all contemporary coding style suggestions for
modern languages I could find specify visually segregating function
calls from variables by using MixedCasePascalStyle for one and
lower_case_underscore for the other. I've rarely seen an advocate of
both styles throughout. That said, I think the overuse of
capitalization is a grievance not to be taken lightly, one which most
net users have learned, often by harsh rebuke, is TANTAMOUNT TO
SHOUTING. Purveyors of this false truth can usually be excused as
having been damaged early on by over-exposure to FORTRAN (ditto for
GOTO). None of the coding style guidelines I've found recommend
significant use of ALL-CAPS.

I weigh in on the side of the quieter, less visually obstreperous use
of lower case variables and identifiers, saving upper case for spare
and special purposes like certain comparisons (AND, OR, EQ, NOT) and
keywords (device,/COLOR), which might otherwise get lost in the line
noise. I like MixedCase (but not uppercase) Methods, and lower-case
for most functions (except those associated with methods, or OO-like
bundles of code). And I use syntax coloring, which both IDLDE and
IDLWAVE do with aplomb, to provide additional visual differentiation.

Just the opinion of one,

JD

P.S. A comparison of the styles remains at:
 http://turtle.as.arizona.edu/idlstyle.html

JD Smith (jdsmith@as.arizona.edu) writes:

> I weigh in on the side of the quieter, less visually obstreperous use
> of lower case variables and identifiers, saving upper case for spare
> and special purposes like certain comparisons (AND, OR, EQ, NOT) and
> keywords (device,/COLOR), which might otherwise get lost in the line
> noise. I like MixedCase (but not uppercase) Methods, and lower-case
> for most functions (except those associated with methods, or OO-like
> bundles of code). And I use syntax coloring, which both IDLDE and
> IDLWAVE do with aplomb, to provide additional visual differentiation.
>
> Just the opinion of one,

Of course, those of us who have been around long enough
know that programming style changes almost as often
as the whims of fashion. Those of us without a
strong moral compass are often buffeted back and forth,
and over a career our code looks about as sensible as
everything else in our closet. :-(

Cheers,

David

P.S. Let's just say I hope wide ties come back into fashion
soon, or I'm going to look damn funny laid out for my funeral.

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155