In article <1993Jan6.175820.21889@news2.cis.umn.edu>, patel@sparky.drad.umn.edu (8-))
writes...
> I have seen postings on this net concerning IDL widgets
> but none concerning the PV_WAVE widgets. I was wondering if
> anyone out there has tried out WAVE widgets. I am running
> PV-WAVE CL 4.01 with widgets. I like the ease of programming
> these widgets. Moreover, we can use any of the Motif (or OL)
> widgets also apart from the Wave widgets and also mix them.
> I donot know if same things hold true for IDL widgets. IDL user?

Not being familiar with PV-Wave, I'm not sure I follow you. IDL widgets work
the same on both Motif and OpenLook, and take on the native look and feel of
either system, depending on which machine you're running on.

There are a small number of basic widget types available. Off the top of my
head, I can think of base, button, slider, list (inclusive and exclusive),
menu, draw, text and label. There are keyword parameters that extend the
capabilities of each widget type, such as allowing slide bars on draw or list
widgets, for example.

The latest version of IDL introduces the concept of combined widgets, which are
routines that combine several widgets, which can then be called as if it were a
single widget.

> Only thing I dislike is passing the parametes by common block.
> The program look like FORTRAN programs and I hate the common blocks.
> I was wondering if PV-WAVE can come up with something like a global
> declarations. This might ease a bit the parameter passing and
> make it easy to add/delete parameters to be passed.

It sounds like widget procedures in IDL and PV-WAVE work very similarly in this
case.

Of course, system variables are global and could be defined by the user in
either IDL or PVWAVE. I have stored parameters used in a number of routines in
a system variable that was defined as a structure.

> I was wondering if someone out there can summarize the pros and cons
> of IDL and WAVE widgets.

How about discussing the pros and cons of IDL versus PV-WAVE just in general?
Anybody out there have much experience in both programs?

Bill Thompson

Subject: Re: PV-WAVE widgets
Posted by lindstrom on Thu, 07 Jan 1993 14:07:38 GMT
View Forum Message <> Reply to Message

In article <1993Jan6.175820.21889@new[Cs2.cis.umn.edu>,
  patel@sparky.drad.umn.edu (8-)) writes:

> Only thing I dislike is passing the parametes by common block.
> The program look like FORTRAN programs and I hate the common blocks.
> I was wondering if PV-WAVE can come up with something like a global
> declarations. This might ease a bit the parameter passing and
> make it easy to add/delete parameters to be passed.

Amen, brother. We are writing data analysis code which is
currently ~5000 lines with LOTS os subroutines. There are
some routines that contain 10 lines or more of common blocks!
Global variables would greatly reduce at least this aspect of
the code.

Also, it would be nice if one could call subroutines from
other directories. For example, if a branch of the program
deals with data retreival (off the 8mm tape), I would like
to place all of the routines dealing with this in a sub-directory
named ./Get_Data. This would allow me to clean up the one
directory holding everything now without fiddling with the
search paths.

Having said this, we are not about to switch over to Solaris 2.1
until PV-Wave runs on it.. we would be lost without PV-Wave!

> Maqbool Patel

Greg Lindstrom
Harding University

Subject: Re: PV-WAVE widgets
Posted by pendleton on Thu, 07 Jan 1993 16:40:45 GMT
View Forum Message <> Reply to Message

In article <1993Jan7.090739.826@ualr.edu>, lindstrom@acs.harding.edu writes:
>
> Amen, brother. We are writing data analysis code which is
> currently ~5000 lines with LOTS os subroutines. There are
> some routines that contain 10 lines or more of common blocks!
> Global variables would greatly reduce at least this aspect of
> the code.
>

> Greg Lindstrom
> Harding University

I get around this problem by simply declaring the common block in an external
text file and adding an include ("@file-name") in each of the calling procedures
Saves a lot of editing if you make a change in one of the commons.

Jim Pendleton
GRO/OSSE
Northwestern U.

Subject: Re: PV-WAVE widgets
Posted by joslyn on Thu, 07 Jan 1993 18:35:38 GMT
View Forum Message <> Reply to Message

In article <6JAN199316245880@stars.gsfc.nasa.gov> thompson@stars.gsfc.nasa.gov (William
Thompson, code 682.1, x2040) writes:
> In article <1993Jan6.175820.21889@news2.cis.umn.edu>, patel@sparky.drad.umn.edu (8-))
writes...

>> I was wondering if someone out there can summarize the pros and cons
>> of IDL and WAVE widgets.

Sure, IMHO...

I have used both the IDL widgets and the new (version 4.01) PV-WAVE widgets
for creating sizable applications, and think that WAVE widgets are much
better for creating big and involved applications because of its similarity
to X programming. Below is a biased (because I like WAVE widgets) summary
of both. It has been probably close to a year since I've used IDL widgets,
so I welcome any corrections.

Widgets

The new PV-WAVE widgets gives the user access to all the motif and openlook
widgets (this includes things like the file open dialogs, selection lists,
and all the convience routines given to any "normal" motif or openlook
programmer), whereas IDL widgets only supply a limited set of widgets (as
Bill pointed out). The IDL widgets include base windows, buttons, radio
boxes, drawing windows (which is the IDL 'window'), sliders, text boxes,
labels, and lists.

Interfaces

IDL's interface to their widgets consists of a number of WIDGET_* (BASE,
BUTTON, LIST, TEXT, INFO, etc) where the different options are passed
in as parameters and keywords. This type of simplified interface is well

suited for people who are not familiar with X (or some other kind of GUI interface) programming. The learning curve for those folks may be a bit faster, but I can't be positive since I wasn't new to GUI programming when I used either of them. On the other hand, another reason I liked WAVE widgets, is because the interfacing routines were very, VERY similar to the C calls needed to create the same thing. Being very similar means that most routines like XtCreate() are WtCreate() and take basically the same parameters. However, this doesn't mean that it is like programming in X. A number of convience routines, Ww*() routines, provide methods for initializing the X server, creating pop-up message boxes, etc.

Event Handling (& callbacks)

Probably the best thing about the WAVE widgets over the IDL widgets is the event handling capabilities. IDL provides the user with a single routine where all events (and callbacks) are sent. The user must then parse the event through a huge CASE statement. This can generate some pretty ugly looking code if the application gets big enough. However, WAVE widgets provides methods to add any number of real X callbacks and event handlers to any widget and associate it to any any user written routine. This, along with having access to all the widgets, is probably the biggest winning point for WAVE.

 - Christopher (joslyn@cs.Colorado.EDU)
--
----- Purgamentum Init, Exit Purgamentum ------------------------------------
Christopher M. Joslyn   Voice - (303) 449-5497
University of Colorado - Boulder   joslyn@spot.Colorado.EDU
Computer Science//CB 430//80310

In article <1993Jan7.090739.826@ualr.edu>, lindstrom@acs.harding.edu writes:
> In article <1993Jan6.175820.21889@new[Cs2.cis.umn.edu>,
>   patel@sparky.drad.umn.edu (8-)) writes:
>
>> Only thing I dislike is passing the parametes by common block.
>> The program look like FORTRAN programs and I hate the common blocks.
>> I was wondering if PV-WAVE can come up with something like a global
>> declarations. This might ease a bit the parameter passing and
>> make it easy to add/delete parameters to be passed.
>
> Amen, brother. We are writing data analysis code which is

> currently ~5000 lines with LOTS os subroutines. There are
> some routines that contain 10 lines or more of common blocks!
> Global variables would greatly reduce at least this aspect of
> the code.

You could keep all your 'global' variables in a structure or two and declare
them in the common block. Using structures to group related variables together
is good practice anyway. I tend to keep all the varaibles dealing with a set
of widgets and their contents etc in one structure.

Someone suggested another method in this newsgroup a month or two ago, whereby
you can use the UVALUE for the widgets to pass information around. This was
for IDL, but I imagine it would work for PV-Wave.

Nathaniel Livesey
Dept. of Atmospheric Physics
Oxford.

Subject: Re: PV-WAVE widgets
Posted by thompson on Fri, 15 Jan 1993 21:34:00 GMT
View Forum Message <> Reply to Message

In article <1993Jan7.183538.23237@ucsu.Colorado.EDU>, joslyn@spot.Colorado.EDU
(Christopher Joslyn) writes...
> In article <6JAN199316245880@stars.gsfc.nasa.gov> thompson@stars.gsfc.nasa.gov (William
Thompson, code 682.1, x2040) writes:
>> In article <1993Jan6.175820.21889@news2.cis.umn.edu>, patel@sparky.drad.umn.edu (8-))
writes...
>
>>> I was wondering if someone out there can summarize the pros and cons
>>> of IDL and WAVE widgets.
>
> Sure, IMHO...

 (stuff deleted)

> Interfaces
> ----------
> IDL's interface to their widgets consists of a number of WIDGET_* (BASE,
> BUTTON, LIST, TEXT, INFO, etc) where the different options are passed
> in as parameters and keywords. This type of simplified interface is well
> suited for people who are not familiar with X (or some other kind of GUI
> interface) programming. The learning curve for those folks may be a bit
> faster, but I can't be positive since I wasn't new to GUI programming
> when I used either of them. On the other hand, another reason I liked
> WAVE widgets, is because the interfacing routines were very, VERY similar
> to the C calls needed to create the same thing. Being very similar means

> that most routines like XtCreate() are WtCreate() and take basically the
> same parameters. However, this doesn't mean that it is like programming
> in X. A number of convience routines, Ww*() routines, provide methods
> for initializing the X server, creating pop-up message boxes, etc.

 (rest deleted)

Thank you for your discussion of the differences between IDL and PV-WAVE. I
thought it was very clear and informative.

I've always thought of IDL (and by inference PV-WAVE) as a utility for people
who *aren't* sophisticated programmers. I know that I myself hate the idea of
trying to program in C. Having no previous experience with widgets programming
(but being an experienced IDL user), I was able to grasp the rudiments of IDL
widgets programming and be able to write simple routines in about an hour. I
can't comment on the learning curve for PV-WAVE widgets, never having been
exposed to them.

One thing I was unclear on about PV-WAVE widgets: widget routines written in
IDL are completely portable between Motif and OpenLook. Routines written on a
Motif system can be run on an OpenLook system without any software changes.
The application takes on the look and feel of the native system. Supposedly
this will be extended to Microsoft Windows as well. Is the same true about
PV-WAVE, or is one making explicit calls to either the Motif or OpenLook
development libraries?

Bill Thompson

Subject: Re: PV-WAVE widgets
Posted by mberkley on Sat, 16 Jan 1993 21:53:35 GMT
View Forum Message <> Reply to Message

> On Thu, 7 Jan 1993 18:35:38 GMT, joslyn@spot.Colorado.EDU (Christopher Joslyn) said:

CJ> This type of simplified interface is well suited for people who
CJ> are not familiar with X (or some other kind of GUI interface)
CJ> programming.

IDL's simplified widget interface also leads to faster GUI programming
for experienced GUI programmers. Sort of like the difference between
programming in XLib and Xaw.

Mike Berkley
mberkley@sirius.UVic.CA