Subject: Re: IDL Runtime
Posted by David Fanning on Wed, 14 Nov 2001 23:31:44 GMT
View Forum Message <> Reply to Message

Andre Kyme (nak@imag.wsahs.nsw.gov.au) writes:

> Thanks for the replies re RUNTIME and RETALL.

Oh, oh. I was afraid of this. :-(

> Perhaps my problem is some confusion relating to how XMANAGER works. As
> far as I can understand if I create a bunch of buttons under a TL base,
> then call XMANAGER, the program is 'hung' in a loop checking whether
> events are being processed. So far OK?

Well, not exactly. There isn't any "loop" to get
hung up in, really. I prefer to think that
an officer of the court has been assigned to
to "pay attention", but it's probably a matter
of semantics, and it doesn't really matter which
metaphor you prefer to use.

> If one of the buttons is clicked,
> then we jump to the event handler for that button; but suppose that the
> event handler calls an event-processing routine that calls another, that
> calls another etc. - so we end up a few levels deep in the processing of
> the event. I have used modal prompts along the way in this event
> processing so that the user can decide "Yes" "No" "Maybe" and so on. One
> option I want to give them is "Cancel back to the bunch of buttons."
> Retall seemed a useful way of getting back to main program level, which am
> I right in saying will enter us back into the XMANAGER loop ready for the
> next event??

No, I don't think so. I really think it is a minor miracle
that your program works at all, but I believe it is because
widget programs don't really need to use XMANAGER. (And
I really don't want to get into any of this at all.) I think your
program works, but for the wrong reason, I guess is what I
am trying to say.

> David says, "Try a more conventional way than using retall" -
> what is the conventional way of returning from deep in event-processing
> programs back to the menu buttons?

The "conventional" way of writing widget programs is
to get into and out of event handlers as fast as
possible, not to spend the rest of the afternoon
in there trundling around. But I guess I can see

some sort of hierarchical data structure that might
require something like you describe. Maybe you want
to build a data structure based on the user's answers
to questions, or something like that.

If that is the case, then I think you don't have
to do anything special. It doesn't matter how deep you go
in an event handler, you don't get out of the event
handler until your return from all those levels. I guess
what you are trying to do is short-circuit all those
intervening levels.

I write most of my modal widgets with CANCEL
keywords attached to a Cancel button so I can
check to see if the user wanted out of a dialog
without being penalized:

```
   dataStruct = AskQuestion('Want a foo field?', Cancel=cancelled)
   IF cancelled THEN RETURN ELSE $
      dataStruct = Create_Struct(dataStruct, foo, Fltarr(12))
```

My dialogs are set up so that if the user kills the dialog
with the mouse, it is the same as hitting a cancel button.

Perhaps if these functions were a set of nested functions
something like this would be able to set the cancel flag
all the way back to the original event handler level.

(I don't know. It is hard for me to envision exactly
what you have. And I certainly don't want to write
an example program.)

Maybe you just need to re-think your program design,
although I'm sure you don't want to hear that after
writing what is, I am sure, a complicated program.

Good luck with this.

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting
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