
Subject: Re: Storing variables in multiple widgets applications
Posted by [James Kuyper](#) on Wed, 03 Nov 2004 17:26:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Jesús Domínguez Colino wrote:

- > Hi all,
- >
- > i have read many things about user values and common blocks but still
- > don't find the best solution.
- >
- > Imagine you have a top-level base with many children and many
- > parameters each. Which is the best way to store these parameters? I am
- > storing all the parameters in the user value of the top-level base,
- > but i don't think this is the best solution when you have many
- > parameters. Defining common blocks is something that has also many
- > disadvantages. I have thought about defining external objects or
- > saving the variables to .sav files. Which is the optimal solutions?
- > What do you think?

When you have many parameters, I think you'll find that many of those parameters are specific to the component parts of your widget, rather than being specific to the top-level base. Therefore, store those parameters in the user values of the corresponding components.

Subject: Re: Storing variables in multiple widgets applications
Posted by [jesusin](#) on Thu, 04 Nov 2004 07:50:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

James Kuyper <kuyper@saicmodis.com> wrote in message
news:<418914D8.70008@saicmodis.com>...

- >
- > When you have many parameters, I think you'll find that many of those
- > parameters are specific to the component parts of your widget, rather
- > than being specific to the top-level base. Therefore, store those
- > parameters in the user values of the corresponding components.

But when you kill these components you have to pass the parameters to the top-level base or store them somehow. Just imagine you need all the parameters for a "process button" in the top-level base.

Greetings,
Jesús

Subject: Re: Storing variables in multiple widgets applications
Posted by [Antonio Santiago](#) on Thu, 04 Nov 2004 10:51:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Perhaps you can encapsulate the information of every component in an object and associate the UVALUE to that object.

"component" <-----> "Object with component information"

The top-level widget UVALUE can have a set of references to the object information on every component.

```
top-level <-----> "Object component info 1"
      |
      |-----> "Object component info 2"
      |
      |-----> "Object component info 3"
```

You can remove the component and the last information persists in the object.

It is only an idea.

Bye :)

```
> Jes?s Dom?nguez Colino wrote:
> Hi all,
>
> i have read many things about user values and common blocks but still
> don't find the best solution.
>
> Imagine you have a top-level base with many children and many
> parameters each. Which is the best way to store these parameters? I am
> storing all the parameters in the user value of the top-level base,
> but i don't think this is the best solution when you have many
> parameters. Defining common blocks is something that has also many
> disadvantages. I have thought about defining external objects or
> saving the variables to .sav files. Which is the optimal solution?
> What do you think?
>
> Thanks in advance,
> Jes?s Dom?nguez Colino
```

Subject: Re: Storing variables in multiple widgets applications
Posted by [kuyper](#) on Sat, 06 Nov 2004 03:16:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

jesusin@gmail.com (Jes?s Dom?nguez Colino) wrote in message
news:<fd04930c.0411032350.4f49322@posting.google.com>...
> James Kuyper <kuyper@saicmodis.com> wrote in message
news:<418914D8.70008@saicmodis.com>...

>
>> When you have many parameters, I think you'll find that many of those
>> parameters are specific to the component parts of your widget, rather than
>> being specific to the top-level base. Therefore, store those parameters in
>> the user values of the corresponding components.
>
> But when you kill these components you have to pass the parameters to
> the top-level base or store them somehow. Just imagine you need all
> the parameters for a "process button" in the top-level base.
> Greetings,
> Jes?s

If you still need the parameters after killing the component, then the parameter isn't really related to that component. You'll have to make it a part of the top-level base's user value, or at least store it in a pointer or a file that are referenced from the top-level base's user value. Sorry.

Subject: Re: Storing variables in multiple widgets applications
Posted by [R.Bauer](#) on Sun, 07 Nov 2004 10:39:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Jes?s Dom?nguez Colino wrote:

> James Kuyper <kuyper@saicmodis.com> wrote in message
news:<418914D8.70008@saicmodis.com>...
>
>> When you have many parameters, I think you'll find that many of those
>> parameters are specific to the component parts of your widget, rather
>> than being specific to the top-level base. Therefore, store those
>> parameters in the user values of the corresponding components.
>
>
> But when you kill these components you have to pass the parameters to
> the top-level base or store them somehow. Just imagine you need all
> the parameters for a "process button" in the top-level base.
> Greetings,
> Jes?s

Dear Jes?s

this is a small example from our exercises. It shows how to use a

pointer as uvalue. The pointer stays as long as it is destroyed.

http://www.fz-juelich.de/vislab/software/idl_samples/Widgets/Oberflaechen/wid5.pro

more examples you'll find here.

http://www.fz-juelich.de/vislab/software/idl_samples/IDL-Bei spielsammlung.html

cheers

Reimar

--

Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-I)

Forschungszentrum Juelich

email: R.Bauer@fz-juelich.de

a IDL library at ForschungsZentrum Juelich

http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html

=====