Subject: Re: iTools Layout - Some help for once :-)
Posted by Antonio Santiago on Mon, 06 Mar 2006 08:14:02 GMT
View Forum Message <> Reply to Message

Thanks for the tip James.

I just take a look at DoSetProperty, and I saw it uses the SET_PROPERTY service that is nothing mora than an operation (IDLitopSetProperty) and , also it uses de CommandBuffer to bring redo/undo options.

I have a problem with this (the undo/redo) method because in my program I don't want to offer this possibility, and also I dont want to use extra space storing the previous value of a property or an image. If you know some tips to avoid the command buffer I will appreciate to you a lot.

Bye.

> Hello everybody,
>
> I'm not sure whether any of you have been subjected to any of my
> previous posts/cries for help, but when my questions started getting
> out of the boundaries of this forum, I had to do break off for a while
> and now I'm back with some knowledge. :-)
>
> I had been creating an iTool that, aside from having some panel UI's
> with thumbnail images, needed to switch back and forth between
> different layouts. In some cases, I wanted to have a single view with
> one image, but then in other cases I would want to switch to a 2x2 grid
> layout and have four different images. I'm sure a lot of you have been
> through the iTools tutorials and know how to do this using the menus,
> but when it came time to actually programming a way to do this, I had
> gotten very stuck. Luckily, I found a great thing called
> ITPROPERTYREPORT that helps a lot when working with iTools.
>
> If you perform a search using IDLitTool::FindIdentifiers for the string
> '*layout*', you'll be able to find the full identifier /TOOLS/IMAGE
> TOOL/OPERATIONS/WINDOW/LAYOUT (which we'll call layoutID). When you
> have an object reference to the tool (let's call it oTool), you can run
> the line:
>> ITPROPERTYREPORT, oTool, layoutID, /value
> and this will show you all the identifiers associated with the layout,
> along with their names, types, and values. You'll see something along
> these lines (this is a shortened list of what actually appears):
>
>
>> Properties of /TOOLS/IMAGE TOOL/OPERATIONS/WINDOW/LAYOUT
>>

```
>> Identifier                Name                  Type          Value
>> ----------                 ----                  ----          -----
>> NAME                      Name                  STRING        Layout...
>> DESCRIPTION               Description            STRING        Layout...
>> SHOW_EXECUTION_UI    Show dialog              BOOLEAN     True
>> VIEW_COLUMNS           Grid columns            INTEGER      1
>> VIEW_ROWS              Grid rows               INTEGER      1
>
>
>  Now, in my case, I wanted to switch to a 2x2 grid layout. First, you
>  need to get an object reference to the layout by using the line:
>>  layoutOBJ = oTool->GetByIdentifier ( layoutID )
>> From here, you can use the IDLitTool::DoSetProperty() function to
>  modify the values you want to change (you'll sometimes see USERDEF
>  types, which I unfortunately haven't found a way to change yet). When I
>  want to change to a 2x2 grid layout, I run these lines:
>
>
>> result = oTool->DoSetProperty(layout_ID, 'SHOW_EXECUTION_UI', 0)
>> result = oTool->DoSetProperty(layout_ID, 'VIEW_COLUMNS', 2)
>> result = oTool->DoSetProperty(layout_ID, 'VIEW_ROWS', 2)
>> result = oTool->DoAction(layout_ID)
>
>
>  You need to start off by setting the SHOW_EXECUTION_UI to 0 because
>  when you're dealing with the layout, modifying any values will bring up
>  the Layout menu. Also, you need to finish off with the
>  IDLitTool::DoAction() method in order to complete the operation.
>
>  I'm sure that this method of using DoSetProperty can be used to modify
>  a lot of things in iTools, but i haven't yet tested them out.
>
>  I hope that some of you are pleased that I'm able to post some help,
>  finally. I know I sure am :-)
>
>  Sincerely,
>
>  - James
>



--
-----------------------------------------------------
Antonio Santiago Pï¿½rez
( email: santiago<<at>>grahi.upc.edu       )
(   www: http://www.grahi.upc.edu/santiago )
(   www: http://asantiago.blogsite.org     )
-----------------------------------------------------
```

Subject: Re: iTools Layout - Some help for once :-)
Posted by Michael Galloy on Fri, 31 Mar 2006 01:47:54 GMT
View Forum Message <> Reply to Message

Antonio Santiago wrote:
> Thanks for the tip James.
>
> I just take a look at DoSetProperty, and I saw it uses the SET_PROPERTY
> service that is nothing mora than an operation (IDLitopSetProperty) and
> , also it uses de CommandBuffer to bring redo/undo options.
>
> I have a problem with this (the undo/redo) method because in my program
> I don't want to offer this possibility, and also I dont want to use
> extra space storing the previous value of a property or an image. If you
> know some tips to avoid the command buffer I will appreciate to you a lot.

You can bypass doSetProperty by using setProperty directly on the object
whose properties you want to modify. Remember that doSetProperty is a
method of the tool object which takes an argument that is the identifier
of the component whose properties you are changing. This would be like:

```
status = oTool->doSetProperty(id, 'COLOR', [255, 0, 0])
```

If you want to use setProperty directly, then use getByIdentifier method
of the tool like:

```
oComponent = otool->getByIdentifier(id)
oComponent->setProperty, COLOR=[255, 0, 0]
```

to get the object reference of the component. This bypasses the
undo/redo system.

-Mike