
Subject: Re: Subclassing from IDLitTool ??
Posted by [Chris\[2\]](#) on Wed, 08 Sep 2004 16:35:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Antonio & Chris,

There is a bug in framework/idlitcommandbuffer__define.pro, around line 232, where it assumes that you have the Undo/Redo operations. You can work around this by making sure the Undo/Redo operations are registered in your subclass:

```
self->RegisterOperation, 'Undo', $  
    'IDLitopUndo', $  
    ACCELERATOR='Ctrl+Z', $  
    IDENTIFIER='Edit/Undo', ICON='undo', /disable, $  
    /IGNORE_AVAILABILITY
```

```
self->RegisterOperation, 'Redo', $  
    'IDLitopRedo', $  
    ACCELERATOR='Ctrl+Y', $  
    IDENTIFIER='Edit/Redo', ICON='redo', /disable, $  
    /IGNORE_AVAILABILITY
```

Or, if you are brave or don't want to have these operations, you can modify idlitcommandbuffer__define.pro by adding a check after line 232:

change this:

```
oUndo = self._oEnv->GetByIdentifier('OPERATIONS/EDIT/UNDO')  
idUndo = oUndo->GetFullIdentifier()
```

to the following:

```
oUndo = self._oEnv->GetByIdentifier('OPERATIONS/EDIT/UNDO')  
if (~OBJ_VALID(oUndo)) then $  
    return  
idUndo = oUndo->GetFullIdentifier()
```

-Chris
Research Systems, Inc.

"Christopher Lee" <cl@127.0.0.1> wrote in message
news:20040901.182404.1960709859.9943@buckley.atm.ox.ac.uk...
> In article <4135CD89.4050300@est.fib.upc.es>, "Antonio Santiago"
> <d6522117@est.fib.upc.es> wrote:
>
>
>> What do i need to do to subclassing from IDLitTool? Thanks.
>
> IDLitToolbase comes preloaded with all of the menus in the IDL

```
> standard iTools. IDLitTool comes with zero, nadda,  
> zip. You have to supply all of the menu items (for example). Then  
> IDLitTool contains the necessary functions to return the object  
> references.  
>  
> e.g.  
>  
> ;Init method  
> FUNCTION iprobe::Init  
>  
>   if( self->IDLitTool::Init(_extra=_extra) eq 0)then $  
>     return, 0  
> ;file  
>   self->RegisterOperation, 'Exit', 'IDLitopFileExit', $  
>     IDENTIFIER='File/Exit', /SEPARATOR  
>  
> ;button  
>   self->RegisterManipulator, 'Arrow', 'IDLitManipArrow', $  
>     ICON='arrow', /DEFAULT, IDENTIFIER="ARROW", $  
>     DESCRIPTION='Click to select items, click and drag to select  
multiple items'  
>  
>   return, 1  
>  
> END  
>  
> That gets you onto the first step...  
>  
> Chris.
```
