Subject: Re: removing objects from a container
Posted by Karl Schultz on Fri, 09 Jun 2006 23:38:43 GMT
View Forum Message <> Reply to Message

On Fri, 09 Jun 2006 14:08:01 -0400, Ben Tupper wrote:

> Hello,
>
> David - please look out the window at your roses for a few minutes...
>
> I can't quite figure the REMOVE method out.  IDLgrROIgroup inherits from
> (eventually) IDL_CONTAINER so the REMOVE method is supposed to work. I
> don't see in the docs that IDLgrROIgroup overrides the method - but
> something is different.
>
>
> ```
> IDL> group = OBJ_NEW('IDLgrROIgroup')
> IDL> roi = OBJ_NEW('IDLgrROI')
> IDL> group->Add, roi
> IDL> group->REMOVE, /ALL
> % IDLGRROIGROUP::REMOVE: Incorrect number of arguments.
> % Execution halted at: $MAIN$
> IDL> group->IDL_CONTAINER::REMOVE,/ALL
> ```
>
>
> Is this a bug or just a thorn?
>

It looks like a new bug was introduced here in 6.3 while trying to fix another problem.

The graphic objects have a PARENT property that is managed by the IDLgrContainer class as graphic objects are added or removed to/from containers like grModel.  IDLgrROIGroup can't subclass from IDLgrContainer because IDLgrROIGroup inherits from IDLanROIGroup, which inherits from IDL_Container (IDLanROIGroup does not store displayable graphic objects) So, IDLgrROIGroup inherits from IDL_Container for its container functionality, and IDL_Container doesn't manage a PARENT property.

So, prior to 6.3, the PARENT property of an IDLgrROI was never getting updated as it was added or removed from an IDLgrROIGroup.

We fixed this bug in 6.3 by updating the PARENT property in the already-overridden Add method and adding a new override for the Remove method that would clear the PARENT property when an IDLgrROI object was removed.

Unfortunately, the form IDLgrROIGroup->Remove, /ALL was not implemented correctly in the new Remove method. And it probably was not documented because it was supposed to work exactly like the superclass method.

I'll open a change request and fix this for next release.

The only workaround I can think of for now is:

```
group->Remove, group->Get(/ALL)
```

This would work correctly with or without the bug.

If you do

```
group->IDL_CONTAINER::REMOVE,/ALL
```

then the PARENT property of all the ROI's that were in the group will be left referring to the group object. If that doesn't bother you, then this would be another way to do it, but I don't think it is a good idea, as it leaves potential dangling object references.

Karl