Subject: How to find if an object contains another
Posted by Dave Greenwood on Thu, 19 Oct 2000 07:00:00 GMT

View Forum Message <> Reply to Message

How do I determine if an object "contains" another one?

For example, I can create a model object and add a surface object or a plot object. Given that model object, how would I find out whether or not a surface object has been added?

The best I've come up with is something like:

```
  xx = model->get(/all,isa='idlgrsurface')
  s = size(xx)
  if s[1] eq 11 then $
      print,'Has one or more surface' $
  else $
      print,'Has no surfaces'
```

Surely there's a simpler way?

A related question is: how do I test if an object pointer (returned by the GetByName method, for example) is null?

Thanks,
Dave

Dave Greenwood   Email: Greenwoodde@ORNL.GOV
Oak Ridge National Lab   %STD-W-DISCLAIMER, I only speak for myself

Subject: Re: How to find if an object contains another
Posted by davidf on Sun, 03 Dec 2000 08:00:00 GMT

View Forum Message <> Reply to Message

Dave Greenwood (greenwoodde@ornl.gov) writes:

> How do I determine if an object "contains" another one?
>
> For example, I can create a model object and add a surface object or
> a plot object. Given that model object, how would I find out whether
> or not a surface object has been added?
>
> The best I've come up with is something like:
>
> ```
>   xx = model->get(/all,isa='idlgrsurface')
>   s = size(xx)
>   if s[1] eq 11 then $
> ```

>   print,'Has one or more surface' $
>   else $
>   print,'Has no surfaces'
>
> Surely there's a simpler way?

How about something like this:

```
   xx = model->get(/all,isa='idlgrsurface', Count=numFound)
   Print, numFound
```

> A related question is: how do I test if an object pointer (returned by
> the GetByName method, for example) is null?

If an object is NULL it is an invalid object:

```
   IF Obj_Valid(theObject) EQ 0 THEN Print, 'Object is invalid.'
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: How to find if an object contains another
Posted by davidf on Wed, 17 Jan 2001 14:01:20 GMT

View Forum Message <> Reply to Message

Dave Greenwood (greenwoodde@ornl.gov) writes:

> How do I determine if an object "contains" another one?
>
> For example, I can create a model object and add a surface object or
> a plot object. Given that model object, how would I find out whether
> or not a surface object has been added?
>
> The best I've come up with is something like:
>
>   xx = model->get(/all,isa='idlgrsurface')
>   s = size(xx)
>   if s[1] eq 11 then $
>   print,'Has one or more surface' $

>   else $
>   print,'Has no surfaces'
>
> Surely there's a simpler way?

That's one way. But, and I think you have stumbled
upon this, it is much easier (and produces more
readable code) if you give the objects you are
looking for "names" and fish them out with the
GetByName method.

> A related question is: how do I test if an object pointer (returned by
> the GetByName method, for example) is null?

You use the Obj_Valid function to find a null object:

```
   IF Obj_Valid(theObject->GetByName('MYSURFACE') THEN $
      Print, "Object Found!!!" ELSE Print, "Object Not Here."
```

Cheers,

David

P.S. I think Martin is on vacation at the moment, but
he has a wonderful modified IDL_CONTAINER object named
MGS_CONTAINTER that can find any object by name. You
can use wildcard searches, etc. It's very nice for this
kind of thing. You can find it in his extensive program
library.

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155