Subject: Re: selecting model objects
Posted by Rick Towler on Tue, 25 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Thanks David and Mark for your responses.

David Fanning wrote:

> To make a model object "selectable" you must first set
> its SELECT_TARGET keyword. The Select method should then
> return an array of selected objects located under the
> selection point.

I have done this (I was listening in class!) but as I see it the problem is more subtle than this. Let's strip this down all the way.

I require the ability to select individual models within a view and to translate them AS A GROUP.

If I have model "A" which is a child (added to) model "B" within a view that is drawn to a window, will the select method return ALL of the models or just the parent model B? As my program is working now only model B is being returned even if I click on it's child model A.

If in theory the select method can return an array of models (all of the children models plus the parent model) any ideas why the child model information isn't trickling down? The /SELECT_TARGET keyword is set on both models.

Adding model "A" and model "B" directly to the view does allow for selection of individual models. But, this approach requires that I keep track of all of the models in the view and translate them individually when one of them is selected to move.

I guess I have a solution but I am still curious. Has anybody tried using the select method to select child models within a parent model? Is this just impossible?

Thanks all!

-Rick Towler

> Rick Towler (rtowler@u.washington.edu) writes:
>

>> I have a base model (a map) that contains other models (geographic
>> data). I need to select data within the map using the IDLgrWindow
>> Select method. Right now I am only getting the base model returned.
>>
>> object heirarchy is: [data_model]->[map_model]->[oView]->[oWindow]
>>
>> ; my select code
>> item = oWindow->Select(oView, [event.x, event.y])
>>
>> item contains the reference to map_model only.
>>
>> A key point is that I need to anchor the geographic data to the map and
>> I need to be able to translate and scale the map. Because of this I
>> haven't been able to add the data objects directly to the view since
>> each of the objects would translate/scale independently and the
>> geographic data would be dereferenced from the underlying map.
>>
>> is it possible to select a model (or atom) that is contained in another
>> model that is contained in a view?
>>
>> If not, is there a way to "join" 2 or more models so when translating or
>> scaling they behave as 1?
>
> To make a model object "selectable" you must first set
> its SELECT_TARGET keyword. The Select method should then
> return an array of selected objects located under the
> selection point.
>
> 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: selecting model objects
Posted by davidf on Tue, 25 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Rick Towler (rtowler@u.washington.edu) writes:

> I have a base model (a map) that contains other models (geographic
> data). I need to select data within the map using the IDLgrWindow

> Select method. Right now I am only getting the base model returned.
>
> object heirarchy is: [data_model]->[map_model]->[oView]->[oWindow]
>
> ; my select code
> item = oWindow->Select(oView, [event.x, event.y])
>
> item contains the reference to map_model only.
>
> A key point is that I need to anchor the geographic data to the map and
> I need to be able to translate and scale the map. Because of this I
> haven't been able to add the data objects directly to the view since
> each of the objects would translate/scale independently and the
> geographic data would be dereferenced from the underlying map.
>
> is it possible to select a model (or atom) that is contained in another
> model that is contained in a view?
>
> If not, is there a way to "join" 2 or more models so when translating or
> scaling they behave as 1?

To make a model object "selectable" you must first set
its SELECT_TARGET keyword. The Select method should then
return an array of selected objects located under the
selection point.

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: selecting model objects
Posted by Mark Hadfield on Wed, 26 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Rick Towler" <rtowler@u.washington.edu> wrote in message
news:397E213D.4C565D8@u.washington.edu...
> ...
> I require the ability to select individual models within a view and to
> translate them AS A GROUP.
>

> If I have model "A" which is a child (added to) model "B" within a view
> that is drawn to a window, will the select method return ALL of the
> models or just the parent model B? As my program is working now only
> model B is being returned even if I click on it's child model A.

Yes, if model B has its SELECT_TARGET property set then it "traps" selection
for all its children.

You shouldn't be thinking about "clicking on models". Models are containers
with a transformation matrix. They don't have any spatial extent. Sort of
like invisible elbow joints (for rotation) or zoom lenses (for scaling). I
like the zoom lens analogy better: you look *through* a model at its
children. The things you can click on are atoms. Models with the
SELECT_TARGET property set are a special case: they let you gather several
atoms together and make them all act like a single atom. But unless you're
building such a composite atom, I don't think you should be setting a
model's SELECT_TARGET property.

Perhaps what you need to do is work upwards through the graphics tree.
Having caught a selection on one of the atoms (or composite atoms) find the
model it is attached to via the atom's PARENT property. If you
scale/translate/rotate the parent then you affect the selected atom and all
its siblings. (Is that what you want to do? Or do you need to go further up
the tree?)

Mark Hadfield
m.hadfield@niwa.cri.nz http://katipo.niwa.cri.nz/~hadfield/
National Institute for Water and Atmospheric Research
PO Box 14-901, Wellington, New Zealand

> If in theory the select method can return an array of models (all of the
> children models plus the parent model) any ideas why the child model
> information isn't trickling down? The /SELECT_TARGET keyword is set on
> both models.
>
>
> Adding model "A" and model "B" directly to the view does allow for
> selection of individual models. But, this approach requires that I keep
> track of all of the models in the view and translate them individually
> when one of them is selected to move.
>
> I guess I have a solution but I am still curious. Has anybody tried
> using the select method to select child models within a parent model?
> Is this just impossible?
>
>
> Thanks all!

>
> -Rick Towler
>
>
>> Rick Towler (rtowler@u.washington.edu) writes:
>>
>>> I have a base model (a map) that contains other models (geographic
>>> data). I need to select data within the map using the IDLgrWindow
>>> Select method. Right now I am only getting the base model returned.
>>>
>>> object heirarchy is: [data_model]->[map_model]->[oView]->[oWindow]
>>>
>>> ; my select code
>>> item = oWindow->Select(oView, [event.x, event.y])
>>>
>>> item contains the reference to map_model only.
>>>
>>> A key point is that I need to anchor the geographic data to the map
and
>>> I need to be able to translate and scale the map. Because of this I
>>> haven't been able to add the data objects directly to the view since
>>> each of the objects would translate/scale independently and the
>>> geographic data would be dereferenced from the underlying map.
>>>
>>> is it possible to select a model (or atom) that is contained in
another
>>> model that is contained in a view?
>>>
>>> If not, is there a way to "join" 2 or more models so when translating
or
>>> scaling they behave as 1?
>>
>> To make a model object "selectable" you must first set
>> its SELECT_TARGET keyword. The Select method should then
>> return an array of selected objects located under the
>> selection point.
>>
>> 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: selecting model objects
Posted by Rick Towler on Wed, 26 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

O.K. Mystery is solved.

When you set the /select_target keyword on a model the window->select method will only return that models obj reference. If the model (the parent) contains other models with their /select_target keywords set selecting >anything< will return only the parent models obj reference.

This makes sense, too bad it isn't thoroughly documented.

My problem will be solved by adding another model that doesn't have the /select_target keyword set and sticking everything in that.

```
mapModel=obj_new('idlgrmodel', /select_target)
drifterModel1=obj_new('idlgrmodel', /select_target)
drifterModeln=obj_new('idlgrmodel', /select_target)
topModel=obj_new('idlgrmodel')

topmodel->add, [mapModel,drifterModel1,drifterModeln]
```

now I can translate/scale topModel and still select the child models within it.

-Rick

Rick Towler wrote:
>
> Hi all,
>
> I have a base model (a map) that contains other models (geographic
> data). I need to select data within the map using the IDLgrWindow
> Select method. Right now I am only getting the base model returned.
>
> object heirarchy is: [data_model]->[map_model]->[oView]->[oWindow]
>
> ; my select code
> item = oWindow->Select(oView, [event.x, event.y])
>
> item contains the reference to map_model only.
>
> A key point is that I need to anchor the geographic data to the map and
> I need to be able to translate and scale the map. Because of this I
> haven't been able to add the data objects directly to the view since
> each of the objects would translate/scale independently and the
> geographic data would be dereferenced from the underlying map.

>
> is it possible to select a model (or atom) that is contained in another
> model that is contained in a view?
>
> If not, is there a way to "join" 2 or more models so when translating or
> scaling they behave as 1?
>
> thanks!
>
> -Rick Towler

Subject: Re: selecting model objects
Posted by Mark Hadfield on Wed, 26 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Rick Towler" <rtowler@u.washington.edu> wrote in message
news:397CE43B.78DEAA48@u.washington.edu...
>
> I have a base model (a map) that contains other models (geographic
> data). I need to select data within the map using the IDLgrWindow
> Select method. Right now I am only getting the base model returned.
>
> object hierarchy is: [data_model]->[map_model]->[oView]->[oWindow]
>
> ; my select code
> item = oWindow->Select(oView, [event.x, event.y])
>
> item contains the reference to map_model only.
>
> A key point is that I need to anchor the geographic data to the map and
> I need to be able to translate and scale the map. Because of this I
> haven't been able to add the data objects directly to the view since
> each of the objects would translate/scale independently and the
> geographic data would be dereferenced from the underlying map.
>
> is it possible to select a model (or atom) that is contained in another
> model that is contained in a view?
>
> If not, is there a way to "join" 2 or more models so when translating or
> scaling they behave as 1?
>
> thanks!
>
> -Rick Towler

Hmmm. I'm not *sure* I understand what you're trying to do or what the
problem is, but here goes...

You want to translate/scale/rotate map_model in response to mouse events?
Bear in mind that if you know which model you want to move, then you don't
need to call Select at all. You can keep track of the relevant model's
object reference in the object structure, or maybe you know its position in
the view container so you can track it down with one or more Get operations.
Then just translate/scale/rotate that model based on event.x & event.y.

I have example code that works this way in:

http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/mghgrw indow__define.pr
o

though I don't know if looking at it will help--I find it hard to figure it
out myself! An MGHgrWindow doesn't know much at all about its GRAPHICS_TREE,
but it does assume that translation, scaling and rotation always work on the
first model in each view. This isn't really too restrictive.

As always, if you want to use one of the routines from my library, you'd
better get the lot, in:

http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/MARKS_ ROUTINES.tar.gz
http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/MARKS_ ROUTINES.zip

---
Mark Hadfield
m.hadfield@niwa.cri.nz http://katipo.niwa.cri.nz/~hadfield/
National Institute for Water and Atmospheric Research
PO Box 14-901, Wellington, New Zealand