Subject: Problems with Restore
Posted by vcarlos on Wed, 08 Mar 2006 19:07:16 GMT
View Forum Message <> Reply to Message

Hi all,

I am writing a class that I want add save and restore features to it.

My class is defined something like this:

<code>
```
pro OperatorLanguageMapping__define;

   objectClass = $
      {OperatorLanguageMapping,   $
         password : '',   $ ; holds the password
         name : '',   $ ; holds the operator name
         language : '' $; holds the language
      }

end

OperatorLanguageMapping::Restore, fileName

    restore, fileName, RESTORED_OBJECTS = temp
    self.name = temp->GetName()
    self.language = temp->GetLanguage()
    self.password = temp->GetPassword()

end

OperatorLanguageMapping::Save, fileName

   save, self, fileName

end
```
</code>

Obviously, the class has its getter and setter methods.

The problem is that when I do this:

<code>
```
o1 = obj_new("OperatorLanguageMapping")
o1->SetPassword, '123456'
o1->Save, '~/o1.sav'

o1->SetPassword, 'test'
```

o1->Restore, '~/o1.sav'
print, o1->GetPassword
</code>

What I got is: 'test' instead of '123456'

I tried to debug the software changing the method restore to look like this:

<code>
OperatorLanguageMapping::Restore, fileName

    restore, fileName, RESTORED_OBJECTS = temp
     self.password = 'password'
     print, self->GetPassword()

end
</code>

By now what I got is :

password
test

That is, after exiting the restore method of the object, the changes I have made inside are not available outside the method(!!).

I keep trying to find out what was happening, and when I comment out the line that call the restore procedure inside the Restore method of the Class OperatorLanguageMapping, then, what I got was:

password
password

That is, changing the attribute inside the method restore worked.

I don't know if I was clear, my english is not that good and the problem is not easy to explain too...

Does anyone have an idea of what could be wrong?

I have also tried to use IDL_SaveFile object, but the result was the same...

Thanks all

Vinicius

Subject: Re: Problems with Restore
Posted by Antonio Santiago on Thu, 09 Mar 2006 09:02:49 GMT
View Forum Message <> Reply to Message

Here is a partial response of your (our) problem, when you restoring the file, both 'self' and 'temp' references are the same, that is the 'self' reference changes.

```
------ Object code test --------
FUNCTION OperatorLanguageMapping::Init
    RETURN, 1
END


PRO OperatorLanguageMapping::Restore, fileName
print, 'Restoring'
print, 'Before:'
help, self  ;; <----- Here is a reference

     restore, fileName, RESTORED_OBJECTS = temp
print, 'After:'
help, self, temp[0] ;; <----- Here 'self' reference has changed
     self.password = temp[0]->GetPassword()
END

PRO OperatorLanguageMapping::Save, fileName
    save, self, FILENAME=fileName
END


PRO OperatorLanguageMapping::SetPassword, pass
    self.password = pass
END


FUNCTION OperatorLanguageMapping::GetPassword
    RETURN, self.password
END


PRO OperatorLanguageMapping__define;
    objectClass = { OperatorLanguageMapping,   $
                    password : ''   $ ; holds the password
                  }
END
------ Object code test --------

------ Test program --------
```

```
PRO optest
    o1 = obj_new("OperatorLanguageMapping")
    o1->SetPassword, '123456'
    o1->Save, '~/o1.sav'


    o1->SetPassword, 'test'

    o1->Restore, '~/o1.sav'

    print, o1->GetPassword()
END
------ Test program --------
```

A possible solution maybe store the 'self' reference into a variable and
after restoring the file reassign it:

```
PRO OperatorLanguageMapping::Restore, fileName
print, 'Restoring'
print, 'Before:'
help, self

a=self ;; <--- Store 'self' reference'
self=0 ;; <--- Changue it to a scaler, if not, a=self=temp.
     restore, fileName, RESTORED_OBJECTS = temp
print, 'After:'
help, self, temp[0]

self=a ;; <---- Re-assing the initial reference.
     self.password = temp[0]->GetPassword()
END
```

Bye.

--

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

GRAHI - Grup de Recerca Aplicada en Hidrometeorologia
Universitat Politï¿½cnica de Catalunya

Subject: Re: Problems with Restore
Posted by Antonio Santiago on Thu, 09 Mar 2006 09:05:52 GMT
View Forum Message <> Reply to Message

Remember to OBJ_DESTROY the 'temp' reference into the Restore method.

Antonio Santiago wrote:
> Here is a partial response of your (our) problem, when you restoring the
> file, both 'self' and 'temp' references are the same, that is the 'self'
> reference changes.
>
>
> ------ Object code test --------
> FUNCTION OperatorLanguageMapping::Init
> RETURN, 1
> END
>
>
> PRO OperatorLanguageMapping::Restore, fileName
> print, 'Restoring'
> print, 'Before:'
> help, self ;; <----- Here is a reference
>
> restore, fileName, RESTORED_OBJECTS = temp
> print, 'After:'
> help, self, temp[0] ;; <----- Here 'self' reference has changed
> self.password = temp[0]->GetPassword()
> END
>
> PRO OperatorLanguageMapping::Save, fileName
> save, self, FILENAME=fileName
> END
>
>
> PRO OperatorLanguageMapping::SetPassword, pass
> self.password = pass
> END
>
>
> FUNCTION OperatorLanguageMapping::GetPassword
> RETURN, self.password
> END
>
>
> PRO OperatorLanguageMapping__define;
> objectClass = { OperatorLanguageMapping, $
> password : '' $; holds the password
> }
> END

```
>  ------ Object code test --------
>
>  ------ Test program --------
>  PRO optest
>     o1 = obj_new("OperatorLanguageMapping")
>     o1->SetPassword, '123456'
>     o1->Save, '~/o1.sav'
>
>
>     o1->SetPassword, 'test'
>
>     o1->Restore, '~/o1.sav'
>
>     print, o1->GetPassword()
>  END
>  ------ Test program --------
>
>
>  A possible solution maybe store the 'self' reference into a variable and
>  after restoring the file reassign it:
>
>  PRO OperatorLanguageMapping::Restore, fileName
>  print, 'Restoring'
>  print, 'Before:'
>  help, self
>
>  a=self ;; <--- Store 'self' reference'
>  self=0 ;; <--- Changue it to a scaler, if not, a=self=temp.
>       restore, fileName, RESTORED_OBJECTS = temp
>  print, 'After:'
>  help, self, temp[0]
>
>  self=a ;; <---- Re-assing the initial reference.
>       self.password = temp[0]->GetPassword()
>  END
>
>
>  Bye.
>


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

GRAHI - Grup de Recerca Aplicada en Hidrometeorologia
Universitat Politï¿½cnica de Catalunya

Subject: Re: Problems with Restore
Posted by Antonio Santiago on Thu, 09 Mar 2006 09:11:51 GMT
View Forum Message <> Reply to Message

Hello, other time. I think here is the final response to the problem:

The problem is in the SAVE and RESTORE methods:

(at method Save)
"save, self, FILENAME=filename"

saves the a variable with name 'self'

(at method restore)
"restore, fileName, RESTORED_OBJECTS = temp"

restores the 'self' named variable and also stores references of object
into the 'temp' array.
Here is the problem. Your 'self' object reference is replaced by the
previous 'self' variable stored with the save method.

At 'save' method change this:

```
PRO OperatorLanguageMapping::Save, fileName
    a=self
    save, a, FILENAME=fileName
END
```

Now when you make the restore, you are restoreing a named variable 'a'
that doesn't replace the 'self' reference.

Bye.

--

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

GRAHI - Grup de Recerca Aplicada en Hidrometeorologia
Universitat Politï¿½cnica de Catalunya

Subject: Re: Problems with Restore
Posted by vcarlos on Thu, 09 Mar 2006 12:16:39 GMT
View Forum Message <> Reply to Message

Hi Antonio,

Thanks for explanation. It really worked with the changes proposed in
the last post. And more than that, I understood what was happening :)

Thanks

Vinicius

Subject: Re: Problems with Restore
Posted by Antonio Santiago on Thu, 09 Mar 2006 12:38:02 GMT
View Forum Message <> Reply to Message

vcarlos wrote:
> Hi Antonio,
>
> Thanks for explanation. It really worked with the changes proposed in
> the last post. And more than that, I understood what was happening :)

Yep, I have already understand what was happening. Correct a problem
without understand why is like to have a granade in your hands :)

Bye.

>
> Thanks
>
> Vinicius
>

--

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

GRAHI - Grup de Recerca Aplicada en Hidrometeorologia
Universitat Politï¿½cnica de Catalunya

Subject: Re: Problems with Restore
Posted by btt on Thu, 09 Mar 2006 15:36:00 GMT
View Forum Message <> Reply to Message

vcarlos wrote:
> Hi all,
>
> I am writing a class that I want add save and restore features to it.

> OperatorLanguageMapping::Save, fileName
>
> save, self, fileName
>
> end
>

Hi,

It is that the self argument to SAVE that is the trickiest part. But
this modification should work.

```
OperatorLanguageMapping::Save, objRef, fileName

    if n_elements(objRef) EQ 0 then objRef = self
    save, objRef, fileName

end
```

Cheers,
Ben