
Subject: Re: Returning values within _EXTRA.
Posted by [davidf](#) on Tue, 11 Aug 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Imanol Echave (ccaeccai@sc.ehu.es) writes:

> When you use a keyword parameter on a program, it's possible to change the
> value of the keyword (if it's a named value), OK. But, I want to do the same
> using the _EXTRA keyword, and it doesn't work (because a tag of a structure is
> passed by value, I think). Anybody knows how to do it?

Perfect place for the new _REF_EXTRA keyword, which
passes keywords by reference and not by value. :-)

Cheers,

David

--

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

Subject: Re: Returning values within _EXTRA.
Posted by [Vap User](#) on Thu, 13 Aug 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Imanol Echave <ccaeccai@sc.ehu.es> writes:

I'm going to answer this one again to spare myself a little egg on the
face occasioned by my previous answer. I just read the help for
Obj_destroy and find that you can specify both arguments and keywords.
So you should pass keywords to the getproperty method using
_ref_extra.

Consider the following

```
FUNCTION junk::Init, prop1, prop2
  IF n_params() NE 2 THEN return,0
  self.prop1 = prop1
  self.prop2 = prop2
  return,1
END
```

```
PRO junk::cleanup, _ref_extra = extra
```

```
self-> get,_extra = extra
END
```

```
PRO junk::get, prop1 = prop1, prop2=prop2
  IF arg_present( prop1 ) THEN prop1 = self.prop1
  IF arg_present( prop2 ) THEN prop2 = self.prop2
END
```

```
PRO junk::set, prop1 = prop1, prop2=prop2
  IF n_elements( prop1 ) NE 0 THEN self.prop1 = prop1
  IF n_elements( prop2 ) NE 0 THEN self.prop2 = prop2
END
```

```
PRO junk__define
  junk = {JUNK,$
          prop1:0I,$
          prop2:0I }
```

```
END
```

```
IDL> .run /tmp/junk__define.pro
% Compiled module: JUNK::INIT.
% Compiled module: JUNK::CLEANUP.
% Compiled module: JUNK::GET.
% Compiled module: JUNK::SET.
% Compiled module: JUNK__DEFINE.
IDL> j=obj_new('junk',1,2)
IDL> j->get,prop1=p1,prop2=p2 & print,p1,p2
      1      2
IDL> j->set,prop1=3,prop2=10
IDL> j->get,prop1=p1,prop2=p2 & print,p1,p2
      3      10
IDL> obj_destroy,j,prop1=pp1,prop2=pp2
IDL> print,pp1,pp2
      3      10
IDL>
```

```
>
> Hi people:
>
> When you use a keyword parameter on a program, it's possible to change the
> value of the keyword (if it's a named value), OK. But, I want to do the same
> using the _EXTRA keyword, and it doesn't work (because a tag of a structure is
> passed by value, I think). Anybody knows how to do it?
>
> An example: I'm creating a class, and on the CLEANUP method I want to return
```

> the object properties values on keywords, so I call the GETPROPERTY method
> within the CLEANUP method. I tried this:
>
> PRO ...::Cleanup,_EXTRA=extra
>
> self->GetProperty,_EXTRA=extra
> ...
> END
>
> But it doesn't work. Any advice?

--

I don't speak for JPL, it doesn't speak for me.

Well, not all the time, at least.

William Daffer <vapuser@haifung.jpl.nasa.gov>

Subject: Re: Returning values within _EXTRA.

Posted by [Vap User](#) on Thu, 13 Aug 1998 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

Imanol Echave <ccaeccai@sc.ehu.es> writes:

Dave Fanning may already have answered this question. If so, forgive.

Look at the _REF_EXTRA keyword. See 'Keyword Inheritance' in the online help. Not having used it, I'm not completely sure it will fit the bill. The online help makes mention of the unavailability of the *values* of the keyword parameters to the called routine when using _ref_extra.

How can one return anything from the cleanup routine of an object?

>

> Hi people:

> When you use a keyword parameter on a program, it's possible to change the
> value of the keyword (if it's a named value), OK. But, I want to do the same
> using the _EXTRA keyword, and it doesn't work (because a tag of a structure is
> passed by value, I think). Anybody knows how to do it?

>

> An example: I'm creating a class, and on the CLEANUP method I want to return
> the object properties values on keywords, so I call the GETPROPERTY method
> within the CLEANUP method. I tried this:

>

> PRO ...::Cleanup,_EXTRA=extra

>

```
> self->GetProperty,_EXTRA=extra  
> ...  
> END  
>  
> But it doesn't work. Any advice?
```

--

I don't speak for JPL, it doesn't speak for me.

Well, not all the time, at least.

William Daffer <vapuser@haifung.jpl.nasa.gov>
