Subject: Re: Keyword discrimination
Posted by davidf on Wed, 12 Aug 1998 07:00:00 GMT
View Forum Message <> Reply to Message

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

> The question is: How to discriminate between a keyword not set and a keyword
> set to an undefined variable? N_ELEMENTS returns 0 in the two cases.

Don't even get me started about IDL routines with misleading
names... :-)

You are correct. It is IMPOSSIBLE, using N_ELEMENTS (or
KEYWORD_SET, for that matter), to distinguish between
keywords that are unused and keywords that are set to
an undefined variable.

But sometimes you clearly want to. For example, you
would like an output keyword to contain the results of
some time-consuming calculation, but you don't want to
DO the calculation unless the caller of the routine requests
the result. You would like to know if the caller of the
function USED the keyword.

Neither N_ELEMENTS or KEYWORD_SET can furnish this information
reliably. To address this issue, RSI introduced the ARG_PRESENT
function, which *almost* does what you want it to do. It can
tell you if a "passed by reference" variable exists in a program.

To understand this, suppose I have a procedure defined like this:

```
PRO JUNK, DO_CALC=doit
IF ARG_PRESENT(doit) THEN doit = Big_Calculation()
END
```

This will work reliably if I call the function like this
(where XX is an undefined variable):

```
IDL> JUNK, DO_CALC=xx
```

It will fail if I call the function like this:

```
IDL> JUNK, /DO_IT
```

Here the variable argument "doit" is certainly *present*
in my procedure (it has the value of 1), but the ARG_PRESENT
function returns 0 because the variable exists in a "passed
by value" form, not a "passed by reference" form.

Given the limitations of each of these functions, it is usually
possible to work something out in your program, often by
using plenty of IF statements. :-)

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: Keyword discrimination
Posted by Vap User on Wed, 12 Aug 1998 07:00:00 GMT
View Forum Message <> Reply to Message

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

I take it that you are trying to determine whether a keyword is set in
order to know whether to calculate an output variable.

There is no way to discriminate these two cases for versions < 5.1
that I am aware of. However, in IDL 5.1 there is the function
ARG_PRESENT() which will tell you that a keyword or positional
argument is present and is a named variable into which a value may be
passed.

If you are working with a version < 5.1 and the data intended for
output is small, there isn't much penalty in creating it whether the
return variable is there or not. Otherwise, you might need a two
variable scheme, one, a flag to tell the routine to calculate the
output, and the other the actual output area.

Best bet, get IDL 5.1 and use arg_present.

FYI: keyword_set(arg) = 0 if arg is undefined, or arg is not present
in call, or arg=0,

> Hi people:
>
> he question is: How to discriminate between a keyword not set and a
> keyword set to an undefined variable? N_ELEMENTS returns 0 in the

> two cases.

--
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>
