

---

Subject: Re: Passing keywords to DLM

Posted by [Karl Schultz](#) on Mon, 09 Aug 2004 19:35:12 GMT

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

---

"JGG" <jargoogle@explore4life.com> wrote in message  
news:4ed7753c.0408091034.5d054741@posting.google.com...

> Haje,

>

> "Haje Korth" <haje.korth@jhuapl.edu> wrote in message  
news:<cf7tm7\$k8a\$1@aplcore.jhuapl.edu>...

>

>> JG,

>

>>

>

>> I use the lines below to do keyword checking and it works great. I  
recommend

>

>> Ronn Kling's book, it will get you up to speed quickly.

>

>>

>

>> Haje

>

>

> Thanks. I'll have to look up Kling's book. I use nearly the same  
> syntax as you indicated in your email, except that I have 7 keywords  
> instead of 1. The code compiles without error messages and functions  
> correctly with no keywords. With keywords, the code accepts only 2 of  
> the 7 defined and reports that the other 5 keywords are "not allowed"  
> in the call to the function.

>

> If none of them worked, I'd at least look for a syntactical error.  
> But since 2 of the 7 are recognized and all are declared exactly the  
> same way, I'm at a loss.

>

> Here are my declaration statements (keywords medium and hi function  
> correctly while the others cause "keyword not allowed" errors to be  
> issued by IDL. See my previous posting.):

>

```
> typedef struct {  
>     IDL_KW_RESULT_FIRST_FIELD;  
>     IDL_LONG x;  
>     IDL_LONG y;  
>     IDL_LONG z;  
>     IDL_LONG medium;  
>     IDL_LONG hi;  
>     IDL_LONG overwrite;
```

```

> IDL_LONG cleanup;
> } KW_RESULT;
> static IDL_KW_PAR kw_pars[] = {
>     IDL_KW_FAST_SCAN,
>     {"X", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(x)},
>     {"Y", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(y)},
>     {"Z", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(z)},
>     {"HI", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(hi)},
>     {"OVERWRITE", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(overwrite) },
>     {"MEDIUM", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(medium) },
>     {"CLEANUP", IDL_TYP_LONG, 1, IDL_KW_ZERO|IDL_KW_VALUE|15, 0,
>      IDL_KW_OFFSETOF(cleanup) },
>     {NULL}
> };
> KW_RESULT kw;
>
> (void) IDL_KWProcessByOffset(argc, argv, argk, kw_pars,
>                             (IDL_VPTR *) 0, 1, &kw);
>
> Maybe I shouldn't use IDL_KW_VALUE? I adopted the same syntax here as
> in the example given in the 5.6 manual as well as Henry Chapman's
> similar DLM.
>
> The DLM is included in my previous post.
>
> Still stumped,
> JGG.

```

Perhaps you just need to sort your list of keywords. The IDL docs say:

The IDL\_KW\_PAR struct provides the basic specification for keyword processing.

The IDL\_KWProcessByOffset() function is passed a null-terminated array of these

structures. IDL\_KW\_PAR structures specify which keywords a routine accepts, the

attributes required of them, and the kinds of processing that should be done to them.

IDL\_KW\_PAR structures must be defined in lexical order according to the value of

the keyword field.

Karl

---