
Subject: Importing nested structures?

Posted by [skerr](#) on Tue, 15 Jun 1999 07:00:00 GMT

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

Dear IDL gurus,

A colleague and I are very confused about the proper use of the IDL_MakeStruct and IDL_ImportNamedArray routines. Specifically, what do you do with the "type" field of an IDL_STRUCT_TAG_DEF when structures are nested? The IDL 4 Advanced Development Guide is a bit vague on this point. It says it is "either a pointer to another structure type definition, or a simple IDL type cast to void (e.g., (void *) IDL_TYP_BYTE)."

For a nested structure, this suggests it could either be another IDL_STRUCT_TAG_DEF defined previously, or (void *) IDL_TYP_STRUCT. The strange thing is, both work in IDL 4, at least for the sample program I have appended below. My colleague discovered that NULL also works in IDL 5.

Does anyone know what the correct syntax is that will work in both IDL 4 and IDL 5?

Regards,
S. Kerr

----- sample program appended -----

```
#include <stdio.h>
#include "/usr/local/rsi/idl/external/export.h"

int main(int argc, char **argv)
{
    static IDL_LONG one = 1;

    static IDL_STRUCT_TAG_DEF substruct_tags[] = {
        {"P", 0, (void *) IDL_TYP_INT},
        {"Q", 0, (void *) IDL_TYP_INT},
        {0}
    };

    static IDL_LONG matrix_dims[] = {2,2,2};
    static IDL_LONG substruct_dims[] = {1,1};
    static IDL_STRUCT_TAG_DEF s_tags[] = {
        {"MATRIX", matrix_dims, (void *) IDL_TYP_FLOAT},
        {"SUBSTRUCT", substruct_dims, substruct_tags},
        /* should we use (void *) IDL_TYP_STRUCT instead of substruct_tags? */
        {0}
    };
}
```

```

};

typedef struct substruct {
    short int p;
    short int q;
} SUBSTRUCT;

typedef struct data_struct {
    float mat [2] [2];
    SUBSTRUCT sub;
} DATA_STRUCT;

static DATA_STRUCT s_data, *s_new_data;
void *s;
IDL_VPTR v;
int i,j;

IDL_Init(0,&argc,argv);

s_data.mat[0][0] = s_data.mat[1][1] = 1.0;
s_data.mat[0][1] = s_data.mat[1][0] = 0.0;
s_data.sub.p = -999;
s_data.sub.q = +1001;
s = IDL_MakeStruct("FOO", s_tags);

printf("Almost all folks.\n");

v = IDL_ImportNamedArray("FOO",1,&one,IDL_TYP_STRUCT,
    (UCHAR *) &s_data, 0, s);

s_new_data = (DATA_STRUCT *) v->value.s.arr->data;
for (i=0;i<2;i++)
    for (j=0;j<2;j++)
        printf("%f\n",s_new_data->mat[i][j]);
printf("%i\n", s_new_data->sub.p);
printf("%i\n", s_new_data->sub.q);

printf("That's all folks.\n");
}

----- end of sample program -----

```
