
Subject: EDG: Using IDL_MakeTempStruct() vs. IDL_ImportArray()

Posted by [Peter Rodriguez](#) on Tue, 10 Feb 2015 22:11:51 GMT

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

Hi,

My objective is to create within C an IDL_TYP_STRUCT to be returned to IDL.

The EDG and Kling's book cover IDL_ImportArray(), but that costs me coding both for the C & IDL_STRUCT_TAG_DEF structures, so that data may be mapped.

I've been successful just using IDL_MakeTempStruct(strIDLdef, 1, &one, &vIDL, IDL_TRUE) having defined my strIDLdef and populating the structure manually via,

```
char *pIDL=vIDL->value.s.arr->data +
IDL_StructTagInfoByName(strIDLdef,(char*)sTAG,IDL_MSG_LONGJMP,&vTAG);
Then employing CASE(vTAG->type) to stride and store within the data space appropriately.
```

However, upon return to IDL the structure fields are only accessible by index not tagname.

Although the in-built tag_names() works!

```
IDL> o=my_make_struct("blablabla")
```

```
IDL> help,o
```

```
** Structure <743af8>, 7 tags, length=832, data length=830, refs=1:
```

```
  sSCAN      STRING   'PPI'
  iANT_HGT_m  INT      320
  nSWEEPs     LONG     3
  fRAYres_deg FLOAT    0.500000
  fBINres_km  FLOAT    0.250000
  fSWEEP_angle_deg_arr
                FLOAT  Array[40]
  sSWEEP_ISO8601_arr
                STRING  Array[40]
```

```
IDL> help,o.sSCAN
```

```
% Tag name sSCAN is undefined for structure <Anonymous>.
```

```
% Execution halted at: $MAIN$
```

```
IDL> help,o.(0)
```

```
<Expression>  STRING   = 'PPI'
```

```
IDL> print,tag_names(o)
```

```
sSCAN iANT_HGT_m nSWEEPs fRAYres_deg fBINres_km fSWEEP_angle_deg_arr
sSWEEP_ISO8601_arr
```

So as a work-around, I've written an IDL wrapper to rebuild/copy the structure based on determining the SIZE(/TYPE) of each field.

Does anyone have insight, how in C may I manually set each structure tagname string? The magic that IDL_ImportArray() apparently does... Something along the lines of tweaking vIDL->value.s.sdef I surmise because from the EDG "the implementation of structure definitions is not public information".

Thanks, Peter Rodriguez

Subject: Re: EDG: Using IDL_MakeTempStruct() vs. IDL_ImportArray()
Posted by chris_torrence@NOSPAM on Tue, 10 Feb 2015 22:30:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Peter,

Here's some sample code from the TIFF module, where it is creating the "info" structure:

```
static      IDL_STRUCT_TAG_DEF s_tags[] = {
    { "CHANNELS", 0, (void *) IDL_TYP_LONG },
    { "DIMENSIONS", dim_dims, (void *) IDL_TYP_LONG },
    { "HAS_PALETTE", 0, (void *) IDL_TYP_INT },
    { "IMAGE_INDEX", 0, (void *) IDL_TYP_LONG },
    { "NUM_IMAGES", 0, (void *) IDL_TYP_LONG },
    { "PIXEL_TYPE", 0, (void *) IDL_TYP_INT },
    { "TYPE", 0, (void *) IDL_TYP_STRING },
    { "BITS_PER_SAMPLE", 0, (void *) IDL_TYP_LONG },
    { "ORIENTATION", 0, (void *) IDL_TYP_LONG },
    { "PLANAR_CONFIG", 0, (void *) IDL_TYP_LONG },
    { "PHOTOMETRIC", 0, (void *) IDL_TYP_LONG },
    { "POSITION", dim_dims, (void *) IDL_TYP_FLOAT },
    { "RESOLUTION", dim_dims, (void *) IDL_TYP_FLOAT },
    { "UNITS", 0, (void *) IDL_TYP_LONG },
    { "TILE_SIZE", dim_dims, (void *) IDL_TYP_LONG },
    { "DESCRIPTION", 0, (void *) IDL_TYP_STRING },
    { "DOCUMENT_NAME", 0, (void *) IDL_TYP_STRING },
    { "DATE_TIME", 0, (void *) IDL_TYP_STRING },
    { "VERSION", 0, (void *) IDL_TYP_LONG },
    { 0 }
};

typedef struct {
    IDL_LONG  channels;
    IDL_LONG  dimensions[2];
    IDL_INT   has_palette;
    IDL_LONG  image_index;
    IDL_LONG  num_images;
    IDL_INT   pixel_type;
    IDL_STRING type_str;
    IDL_LONG  bits_per_sample;
    IDL_LONG  orientation;
    IDL_LONG  planar_config;
    IDL_LONG  photometric;
    float     position[2];
    float     resolution[2];
};
```

```

IDL_LONG  units;
IDL_LONG  tile_size[2];
IDL_STRING description;
IDL_STRING document_name;
IDL_STRING date_time;
IDL_LONG  version;
} ret_struct;

s_def = IDL_MakeStruct(0, s_tags); /* create the struct */
ret = (ret_struct *)
    IDL_MakeTempStructVector(s_def, 1, &vpTmp,IDL_FALSE);

```

I'm not sure how much of this API is actually included in "idl_export.h", so you might be unsuccessful at getting this to compile. But hopefully it will give you a start!

Cheers,
Chris

Subject: Re: EDG: Using IDL_MakeTempStruct() vs. IDL_ImportArray()

Posted by [Jim Pendleton](#) on Wed, 11 Feb 2015 00:30:09 GMT

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

On Tuesday, February 10, 2015 at 3:30:50 PM UTC-7, Chris Torrence wrote:

```

> Hi Peter,
>
> Here's some sample code from the TIFF module, where it is creating the "info" structure:
>
> static      IDL_STRUCT_TAG_DEF s_tags[] = {
>   { "CHANNELS", 0, (void *) IDL_TYP_LONG },
>   { "DIMENSIONS", dim_dims, (void *) IDL_TYP_LONG },
>   { "HAS_PALETTE", 0, (void *) IDL_TYP_INT },
>   { "IMAGE_INDEX", 0, (void *) IDL_TYP_LONG },
>   { "NUM_IMAGES", 0, (void *) IDL_TYP_LONG },
>   { "PIXEL_TYPE", 0, (void *) IDL_TYP_INT },
>   { "TYPE", 0, (void *) IDL_TYP_STRING },
>   { "BITS_PER_SAMPLE", 0, (void *) IDL_TYP_LONG },
>   { "ORIENTATION", 0, (void *) IDL_TYP_LONG },
>   { "PLANAR_CONFIG", 0, (void *) IDL_TYP_LONG },
>   { "PHOTOMETRIC", 0, (void *) IDL_TYP_LONG },
>   { "POSITION", dim_dims, (void *) IDL_TYP_FLOAT },
>   { "RESOLUTION", dim_dims, (void *) IDL_TYP_FLOAT },
>   { "UNITS", 0, (void *) IDL_TYP_LONG },
>   { "TILE_SIZE", dim_dims, (void *) IDL_TYP_LONG },
>   { "DESCRIPTION", 0, (void *) IDL_TYP_STRING },
>   { "DOCUMENT_NAME", 0, (void *) IDL_TYP_STRING },
>   { "DATE_TIME", 0, (void *) IDL_TYP_STRING },

```

```

> { "VERSION", 0, (void *) IDL_TYP_LONG },
> { 0 }
> };
>
> typedef struct {
> IDL_LONG channels;
> IDL_LONG dimensions[2];
> IDL_INT has_palette;
> IDL_LONG image_index;
> IDL_LONG num_images;
> IDL_INT pixel_type;
> IDL_STRING type_str;
> IDL_LONG bits_per_sample;
> IDL_LONG orientation;
> IDL_LONG planar_config;
> IDL_LONG photometric;
> float position[2];
> float resolution[2];
> IDL_LONG units;
> IDL_LONG tile_size[2];
> IDL_STRING description;
> IDL_STRING document_name;
> IDL_STRING date_time;
> IDL_LONG version;
> } ret_struct;
>
> s_def = IDL_MakeStruct(0, s_tags); /* create the struct */
> ret = (ret_struct *)
> IDL_MakeTempStructVector(s_def, 1, &vpTmp, IDL_FALSE);
>
>
> I'm not sure how much of this API is actually included in "idl_export.h", so you might be
> unsuccessful at getting this to compile. But hopefully it will give you a start!
>
> Cheers,
> Chris

```

Peter,

It looks like you just need to ensure that the tag names in your C IDL_STRUCT_TAG_DEF are uppercase only. The interpreter isn't going to handle the mixed case that you're using presently.

Jim P.

Subject: Re: EDG: Using IDL_MakeTempStruct() vs. IDL_ImportArray()
 Posted by [Peter Rodriguez](#) on Thu, 12 Feb 2015 17:05:22 GMT

Yes, thank you, thank you Jim!

Funny how a nuance like that can derail things downstream.

In hindsight I do read in the EDG: char *name must be "Null-terminated uppercase name of the tag."

Hmm, it's an immutable string, so can't patch my code with toupper().
A simple vim command on my *.h file and a uppercase cast/check before
IDL_StructTagInfoByName and I'm good.

Cheers, Peter
