Subject: Re: CALL_EXTERNAL and structures
Posted by Jeff Kommers on Mon, 23 Jun 1997 07:00:00 GMT
View Forum Message <> Reply to Message

> Hubert Rehrauer <rehrauer@vision.ee.ethz.ch>
> In the following example the values of the tag 'second' can not be
> modified and I can't see why!

I think the problem is in the way you defined the structure in
'idltest.c':

> typedef struct {
>   long first;
>   double * second;
> } data;

If your IDL structure is defined like this

IDL> data = {first:0L, second:dblarr(4)}

then in your C program you would want to define the same structure.
That means the second tag should be a double, *not* a pointer to a
double. In fact, if data.second always has 4 or fewer elements, then
in C the second tag should be a "double second[4];".

But assuming that IDL is responsible for allocating memory for the
data structure, and that your C source code does not know how many
elements data.second is going to have, you could do something like
this:

```
/* content of the file 'idltestfix.c' */
typedef struct {
  long first;
  double second; /* Actually the first element in an array of doubles */
                 /* See IDL code */
} data;

long idltestfix( int argc, void * argv[])
{
  data * d;
  double * psecond;

  d = (data * ) argv[0];

  d->first = 25;
  psecond = &d->second;
  *psecond = 1.4;
  *(psecond+2) =3.8;
```

```
  return(0L);
}
```

On SunOS 4.1.3 with IDL 5.0 I did the following

```
IDL> $ acc -c -pic idltestfix.c
IDL> $ ld -o idltestfix.so -assert pure-text idltestfix.o
IDL> a = 5l
IDL> b = dblarr(4)
IDL> data = {first:a,second:b}
IDL> print, data
{           5       0.0000000       0.0000000       0.0000000       0.0000000
}
IDL> check = call_external('idltestfix.so','idltestfix',data)
IDL> print, data
{          25       1.4000000       0.0000000       3.8000000       0.0000000
}
```

Good luck

Jeff
