Subject: Re: C Alignment/IDL structures
Posted by joey on Mon, 21 Mar 2005 16:56:20 GMT
View Forum Message <> Reply to Message

Randall Skelton < randall.skelton@gmail.com > wrote:

- > I'm not sure I completely understand what you are doing... can you post
- > a snipit of code? I trust that you are passing back an unnamed

Here's my code that does the IDL/C interaction. I have a vector (_dataIDL) which has all the values I want into IDL within it as a malloc'ed sets of memory.

```
// Copy the real data
  unsigned long pos = 0;
  unsigned char *myStructureThatLooksLikeTags = malloc (_totalSpaceNeeded
                                   * dataIDL.size ());
  for (unsigned int i = 0; i < _dataIDL.size (); i++) {
memcpy (&(myStructureThatLooksLikeTags [pos]), _dataIDL [i],
         totalSpaceNeeded);
pos += totalSpaceNeeded;
  char idl_struct_name [100];
  sprintf (idl_struct_name, "%s_K%ld_V%d", dbVirtualName (data_key),
         data key, version);
  void *idl struct = IDL MakeStruct (idl struct name, tags);
  IDL LONG dims;
  dims = _dataIDL.size (); // number of elment in the array of structures
  IDL VPTR ivReturn = IDL ImportArray (1, &dims, IDL TYP STRUCT,
                         myStructureThatLooksLikeTags,
                         cleanUpIDL, idl_struct);
```

- > In general, when I have to pass C structures from existing code back to
- > IDL I do it by creating a shadow structure (in C) that uses all the
- > defined IDL types and copying the data. You really cannot rely on
- > generic C variables having the same size as thier IDL counterparts
- > (take a look at the definition of IDL ALLTYPES in idl export.h).

Ok, this probably answers my question. I was hoping I could create an array of structs, but this is maybe not so memory efficient so I might try to create one structure with multiple arrays.

Joey