Subject: Re: Size of variables in bytes?
Posted by davidf on Tue, 02 Sep 1997 07:00:00 GMT
View Forum Message <> Reply to Message

Christian Salow writes:

> is it true that there is no function in IDL 5.0.2 to get the size of a
> variable in bytes (except structures)?

True.

> An integer variable seams to be 2 bytes indpendent from the system
> architecture but a string seams to be 1 byte per character on a Win32
> machine and 2 bytes on a SunOS-machine. I wrote a string to an
> unformatted output file and looked at the filepointer to get the size in
> bytes.

I think it would be less trouble to use the StrLen function to
find the length of the string. This is its size in bytes. Even
on a SunOS machine, I should think.

> Is the following true and is it machine independent?
> byte 1b
> integer 2b
> long int 4b
> floating 4b
> double 8b
> string ???
> structure n_tags(var, /length)

Yes, this is true and it is machine independent.
Strings are always ??? (variable length) in IDL.

> What about:
> complex floating

8 bytes.

> double-precision complex

16 bytes.

> pointer

4 bytes

> object reference

Don't know.

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
Customizable IDL Programming Courses
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com

Subject: Re: Size of variables in bytes?
Posted by pete on Wed, 03 Sep 1997 07:00:00 GMT
View Forum Message <> Reply to Message

Christian Soeller wrote:
>
> Christian Salow <csalow@chaos.bwl.uni-mainz.de> writes:
>
>>
>> Is the following true and is it machine independent?
>>
>> byte 1b
>> integer 2b
>> long int 4b
>> floating 4b
>> double 8b
>> string ???
>> structure n_tags(var, /length)
>>
>
> A related question would be why one has to know? If all you
You may need/want to know what the sizes are because you are giving a
data set generated outside of pvwave/idl. On Dec Alpha workstations the
long int maps to 8 bytes leaving no 4 byte integer. I am interfacing
with a c++ routine that writes out 4 byte integers, to force the c++
code to reengineer and waste 4*1465*15500 (this size is one of the
smaller file size) bytes of disk space per file (appx 40-50 files
generated daily) is absurd --- beside does it hurt to understand how
something is defined --- don't want to start an argument here *I*
> are interested in is cross platform data exchange you are better
> off using an appropriate (IDL supported) data format. Otherwise
> you have to take care of endianess youself, etc. And C programmers
> can use sizeof with the appropriate IDL C-type.
>

> Christian Soeller

--
 -- ------
"Madness takes it's toll. Please have exact change."
 -- ------

KYSOTI :P
