Subject: IDL memory question
Posted by Eugenio Sansosti on Mon, 07 Sep 1998 07:00:00 GMT
View Forum Message <> Reply to Message

Hi everyone,

maybe it is a trivial question, but solving the following will help me a lot!

Suppose I define a very big array, for instance a=fltarr(8000,8000). When I finish using it, I can make memory free by assigning to it a scalar value (for example a=0.), as also suggested in the IDL documentation.

Even if this operation make the required memory free for my IDL application, it does not make memory free for other applicstions running onmy machine. That is, other machine users cannot use the memory I have allocated untill I exit IDL.

Does any of you have a solution for that? Is it an IDL configuration problem or is there any IDL command I can use?

Thanks in advance,
Eugenio

Subject: Re: IDL memory question
Posted by davidf on Tue, 08 Sep 1998 07:00:00 GMT
View Forum Message <> Reply to Message

Mark Hadfield (m.hadfield@niwa.cri.nz) writes:

> For what it's worth, the win32 version does not have this problem. That is,
> the Task Manager shows the memory usage increasing when a large array is
> created and dropping straight back down when it is destroyed.

Apparently the Windows version of IDL is being compiled with a special compiler (whose name completely escapes me at the moment) that has the ability to give freed memory back to the operating system. As many people have pointed out, this is NOT a normal feature of most standard operating system libraries that use Malloc and Free to allocate and free memory.

Yet another reason why I buy more Microsoft stock every time the market dips. :-)

Cheers,

David

---
David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: http://www.dfanning.com/

---

Subject: Re: IDL memory question
Posted by Mark Hadfield on Wed, 09 Sep 1998 07:00:00 GMT
View Forum Message <> Reply to Message

David Kastrup wrote in message ...
>
> Alex Schuster <alex@rosa.mpin-koeln.mpg.de> writes:
>>
>> Eugenio Sansosti wrote:
>>> Even if this operation make the required memory free for my IDL
>>> application, it does not make memory free for other applicstions
running
>>> onmy machine. That is, other machine users cannot use the memory I have
>>> allocated untill I exit IDL.
>>
>> Sorry, there is no way. It seems this is a general problem for any C
>> progam which uses malloc() and free() to access heap memory.
>>
>> The FAQ has an entry about this:
>> http://la.znet.com/~mgs/idl_faq.html#T27
>
> It's a general problem for any C program compiled with a stupid C
> library or working on a braindead system. If your system is not
> braindead and your C library is the GNU C library glibc, then large
> allocations will be done in a way that allow reclaiming space
> immediately by the operating system as soon as it gets released, even
> if the chain of allocations would leave holes in the available memory
> space.

For what it's worth, the win32 version does not have this problem. That is,
the Task Manager shows the memory usage increasing when a large array is
created and dropping straight back down when it is destroyed.

--
Mark Hadfield, m.hadfield@niwa.cri.nz http://www.niwa.cri.nz/~hadfield/
National Institute for Water and Atmospheric Research
PO Box 14-901, Wellington, New Zealand

---

Subject: Re: IDL memory question
Posted by Michael Werger on Thu, 10 Sep 1998 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
....
> Mark Hadfield (m.hadfield@niwa.cri.nz) writes:
....
>> For what it's worth, the win32 version does not have this problem. That is,
>> the Task Manager shows the memory usage increasing when a large array is
>> created and dropping straight back down when it is destroyed.
>
> Apparently the Windows version of IDL is being compiled with
> a special compiler (whose name completely escapes me at the
> moment) that has the ability to give freed memory back to
> the operating system. As many people have pointed out, this
> is NOT a normal feature of most standard operating system
> libraries that use Malloc and Free to allocate and free
> memory.

It is not only the compiler but some cool stuff RSI bought from
another small company especially for this little machines... ;-)

(just a lot of blank lines because my newsgroup mailer stops
me from replying with a shorter reply than the question :-( )
[just opposite to the memory shrink method used in this special
package - they rewrote malloc and free]

--
Michael Werger ----------o
ESA ESTEC & Praesepe B.V. |
Astrophysics Division mwerger@astro.estec.esa.nl|
| Postbus 299 http://astro.estec.esa.nl |
| 2200 AG Noordwijk +31 71 565 3783 (Voice)
o---------- The Netherlands +31 71 565 4690 (FAX)