Subject: How to allocate memory for an array of more than 2G
Posted by minchen on Tue, 13 Jul 2010 23:23:26 GMT

Hi,
I have a trouble to read a .h5 file which is about 3G.
I am using {x86_64 linux unix linux 7.1.1 Aug 21 2009 64 64} IDL. So
you see, my system is 64 bit and IDL is also 64 bit. The problem I
cannot allocate a memory for an array more than 2G. I do not know why?
Can someone help me?
Thanks a lot!

IDL> help,/memory
heap memory used: 818394, max: 1074560678, gets: 1069,
frees: 258
IDL> a=bytarr(1024,1024,1024)
IDL> help,/memory
heap memory used: 1074560268, max: 1074560367, gets: 1081,
frees: 269
IDL> a=bytarr(1024,1024,1024)
% Unable to allocate memory: to make array.
   Cannot allocate memory

Subject: Re: How to allocate memory for an array of more than 2G
Posted by Nigel Wade on Thu, 15 Jul 2010 08:18:27 GMT

On 14/07/10 23:14, NEW-IDL-USER wrote:
> Hi Bob,
>
> Thanks a lot!
>
> I cannot do this:
> IDL> a=bytarr(1024,1024,1024)
> IDL> b=bytarr(1024,1024,1024)
> IDL> c=bytarr(1024,1024,1024)
> IDL> d=bytarr(1024,1024,1024)
> IDL> help,/mem
> on my system. It is a 64 bit linux system and the memory is 64G.
> Is there a limitation of the virtual memory which is related to this
> one? If so, how can I do it?
>

What other processes is the system running? Just because there's 64GB of
system RAM doesn't mean that anything like that is available to you.
Other processes could have already used most of it.

Also, what per-process limits has the sys. admin. placed on you? Have a look at your ulimits to see.

```
-bash-3.00$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
file size               (blocks, -f) unlimited
pending signals                 (-i) 1024
max locked memory       (kbytes, -l) 32
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
stack size              (kbytes, -s) 10240
cpu time               (seconds, -t) unlimited
max user processes              (-u) 150
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

The "virtual memory" limit is the most likely one.

--
Nigel Wade