Subject: Memory de-allocation and "restore"
Posted by rutledge on Wed, 10 May 1995 07:00:00 GMT
View Forum Message <> Reply to Message

I apologize to those who are boring of this subject, but I
have not received any answers at all, and I cannot believe that
the difficulty I am having MUST exist -- there must be SOME way
around it.

I save a variabele "specdata" (a common variable) using a "save" command.
"specdata" contains, at different times, different information, but all of the
same structure, to various files.

I ran the following test:

```
for i=0L, n_ffiles-1 do begin   ; CYCLE THROUGH THE FILES
   specdata = 0
   restore, filename=ffiles(i), /verbose
endfor
```

which cycles through files and restores to the variable "specdata". On the
first iteration, the loop takes about 10 seconds, and no time is spent
at the "specdata=0" line. On the second iteration, approximately 2minutes
are required by the "specdata=0" line, and >10 minutes are required on
that "restore". The second "restore" is half the memory size of the first
"restore", so it is not a problem of the memory requirements.

Is this REALLY the way IDL is? Is there REALLY no way to improve this
(either for me to improve this, or RSI to improve this). If this cannot
be improved, I cannot use IDL in the future, because it is simply not
fast enough on silly 'ol I/O.

Bob

Subject: Re: Memory de-allocation and "restore"
Posted by soc on Fri, 12 May 1995 07:00:00 GMT
View Forum Message <> Reply to Message

Bob Rutledge (rutledge@zvezda.mit.edu) wrote:

: I ran the following test:

: for i=0L, n_ffiles-1 do begin   ; CYCLE THROUGH THE FILES

: specdata = 0
: restore, filename=ffiles(i), /verbose
: endfor

: which cycles through files and restores to the variable "specdata". On the
: first iteration, the loop takes about 10 seconds, and no time is spent
: at the "specdata=0" line. On the second iteration, approximately 2minutes
: are required by the "specdata=0" line, and >10 minutes are required on
: that "restore". The second "restore" is half the memory size of the first
: "restore", so it is not a problem of the memory requirements.

: Is this REALLY the way IDL is? Is there REALLY no way to improve this
: (either for me to improve this, or RSI to improve this). If this cannot
: be improved, I cannot use IDL in the future, because it is simply not
: fast enough on silly 'ol I/O.

: Bob

Hi, well when I tried to do something very similar I find no delay at
all. I just tried saving the data 5 times in a row with a loop, then
read it back five times in row nd it was instantaneous. The structure
was quite large too.

I'd say the problem lies elsewhere. I run on a DEC Alpha running Open
VMS 1.5-1, and idl 3.6.

Rob

Subject: Re: Memory de-allocation and "restore"
Posted by thompson on Tue, 16 May 1995 07:00:00 GMT
View Forum Message <> Reply to Message

rutledge@zvezda.mit.edu (Bob Rutledge) writes:

> I apologize to those who are boring of this subject, but I
> have not received any answers at all, and I cannot believe that
> the difficulty I am having MUST exist -- there must be SOME way
> around it.

> I save a variabele "specdata" (a common variable) using a "save" command.
> "specdata" contains, at different times, different information, but all of the
> same structure, to various files.

> I ran the following test:

> for i=0L, n_ffiles-1 do begin ; CYCLE THROUGH THE FILES
>   specdata = 0
>   restore, filename=ffiles(i), /verbose
> endfor

> which cycles through files and restores to the variable "specdata". On the
> first iteration, the loop takes about 10 seconds, and no time is spent
> at the "specdata=0" line. On the second iteration, approximately 2minutes
> are required by the "specdata=0" line, and >10 minutes are required on
> that "restore". The second "restore" is half the memory size of the first
> "restore", so it is not a problem of the memory requirements.

> Is this REALLY the way IDL is? Is there REALLY no way to improve this
> (either for me to improve this, or RSI to improve this). If this cannot
> be improved, I cannot use IDL in the future, because it is simply not
> fast enough on silly 'ol I/O.

> Bob

I think that the reason why you didn't get any response is because such behavior is not normal. Please give more information about your configuration: What platform? What operating system? How much RAM memory do you have? How large are the files you're trying to restore? If applicable, what quotas are you allowed? How loaded down is your system with other processes and other users?

It sounds like you're having a problem with memory fragmentation. This can happen if the amount of data you're trying to read in is significant compared to your total available memory (e.g. 30%). It's just like disk fragmentation. The problem could be especially acute if your RAM memory is small and you're mainly using virtual memory on disk. If your operating system is Microsoft Windows, I've heard (but can't say for certain) that it is especially prone to memory fragmentation problems.

Bill Thompson