Subject: Re: Reading files with unknown amount of data
Posted by pendleton on Thu, 04 Nov 1993 17:46:56 GMT
View Forum Message <> Reply to Message

In article <ALANS.93Nov4110936@fallout.juliet.ll.mit.edu>, alans@ll.mit.edu (A.J.Stein) writes:
> I always thought the best approach to the "growing array" problem was to
> "cache" the data in an array of a KByte or so. When the "cache" is full,
> append to array. So, rewriting the previously posted example program this
> way yields:
>
> [example function deleted]
>
> Anyway, if there are better ways to do this, I'd sure love to hear about
> them.
> --
> Alan J.Stein MIT/Lincoln Laboratory alans@LL.mit.edu

"Better" depends a lot on what your needs are, of course. For the particular data sets we deal with, reading to EOF while counting records, then creating exact-size arrays and re-reading the file makes the most sense. (Actually, it's a little more complicated since we use indexed files, but you get the idea.)

The excess I/O time we incur is siginificantly less than the time it takes the pager to go out and find more continguous memory for each append. This, after all, is more I/O unless your data set is small enough to fit in physical memory.

This method also leaves around a lot more contiguous memory that will still be available later on.

You should try it both ways, but as your data set size increases, the two-pass method will, I predict, increase your efficiency in both speed and memory utilization.

For most of our analysis tasks, we've tried to avoid array appends completely, even with 400K block (VMS) pagefiles. Appends are easy to code, but they can become real hogs in just a few passes through a WHILE loop.

Jim Pendleton, Programmer Analyst/Technical Services Specialist
GRO/OSSE, Dept. Physics & Astronomy
Northwestern University
j-pendleton@nwu.edu (708) 491-2748 (708) 491-3135 [FAX]