Subject: resizing an array of structures (uugh)
Posted by Randall Skelton on Mon, 04 Jun 2001 11:57:48 GMT
View Forum Message <> Reply to Message

Hi all,

I have an ascii file containing a few thousand lines with each individual dataset comprising 10 lines of floats, ints, strings, etc. It seems logical to read this in as an array of structures as each dataset contains the same information, just different numbers.

My problem is that I don't know what the dimension of the array should be before I start. Initially I just defined a large array and counted the number of datasets for subsequent processing. However, as time progresses and this code gets more use, I have to say that I really hate all the excess array elements... I figured there would be an easy way to resize the array of structures, but the best I can come up with is a double for loop that is rather slow.

```
; loop over the number of array elements
for i, n_elements(array) do begin
   ; loop over the number of tags
   for j, n_tags(structure) do begin
     resized_array[i].(j) = array[i].(j)
   endfor
endfor
```

Is there a *faster* or more elegant way to do this? Does IDL have a *fast* resize command that can handle any type of array to simply adjust the number of elements in the array without rebinning, or otherwise changing the numbers?

Cheers,
Randall

Subject: Re: resizing an array of structures (uugh)
Posted by Pavel A. Romashkin on Wed, 06 Jun 2001 17:01:48 GMT
View Forum Message <> Reply to Message

Hi Randall,

The experts answered already. I thought I might provide another way that I found much faster than anything else for files up to 100,000 rows. Read the file in a loop but use a large buffer, I use 10,000 rows. If the size of the buffer exceeds the length of the file, use Transfer_Count toobtain the right size for the buffer. Then, read the leftover data. Below is an excerpt from my own routine. It is not

structure oriented but is easy to alter to fit your case:

```
; Read data from the file. Try to read by 10000 rows.
Buffer = fltarr(N_PrimVars+1) ; Because column 0 is always time.
readf, SourceFile, Buffer
; Create ResultArray with only 1 record, to initialize.
ResultArray = Transpose(Buffer)
; Create BIG buffer, in case file is long.
Buffer = fltarr(N_PrimVars+1, 10000)

; If Buffer was longer than file, reading will fail. In such case,
reduce buffer.
; Just to skip buffer reduction the first time,
FileTooShort = 0b
on_ioError, reduce_buffer
reduce_buffer : if FileTooShort eq 1b then begin
 unread_rows = (fstat(SourceFile)).Transfer_Count / (n_primVars+1)
 Buffer=fltarr(N_PrimVars+1, unread_rows)
 point_lun, SourceFile, LastPos
endif
; In case of ioError, this will be set to 1b
FileTooShort = 1b

while not EOF(SourceFile) do begin
; Memorize start in case reading fails.
 point_lun, (-SourceFile), LastPos
 readf, SourceFile, Buffer
 ResultArray = [ResultArray, Transpose(Buffer)]
endwhile
```

Subject: Re: resizing an array of structures (uugh)
Posted by davidf on Wed, 06 Jun 2001 17:23:55 GMT

View Forum Message <> Reply to Message

Pavel A. Romashkin (pavel.romashkin@noaa.gov) writes:
>
> The experts answered already.
>

Pavel!! Where are you posting from? We had given you up
for dead. It is nice to hear your voice again, even if there
weren't any jokes. :-)

Cheers,

David
--

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

Subject: Re: resizing an array of structures (uugh)
Posted by Pavel A. Romashkin on Wed, 06 Jun 2001 17:34:49 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Pavel A. Romashkin (pavel.romashkin@noaa.gov) writes:
>>
>> The experts answered already.
>>
>
> Pavel!! Where are you posting from? We had given you up
> for dead. It is nice to hear your voice again, even if there
> weren't any jokes. :-)

Since I haven't been using IDL much this last month, you can expect my
next code snippet to be the funniest joke of the month :-)

Cheers,
Pavel

P.S. I am back home, and that fresh, cold Sawtooth is soo-o easily
available ... where are you?