Subject: Re: array concatenation and optimization
Posted by alt on Thu, 27 Sep 2001 05:05:17 GMT
View Forum Message <> Reply to Message

Our coryphaei are seem to keep silence so I will try to help you
...:-)

> Is there a reason this is so slow?
Because in each concatenation step IDL initializes new variable with memory freeing, allocation, coping, etc. And it becomes slower and slower with increasing of array size.
> Is there a better way to do this?
Using temporary() will not help as it seems to.

Very obvious solution is to allocate array of maximum possible size for data and clip it to filled size at the end.

```
arr = bytarr(10000,/nozero)
i = 0L
while ... do begin
arr[i] = item
i = i + 1
endwhile
arr = temporary(arr[0:i-1])
```

This way is not always good because one have to think of maximum number of data and allocate huge superfluous memory. The next idea is to lengthen array by the lump only on demand. I am using for this purpose my simple "AS IS" procedure.

```
pro AddItem, arr, Q, item, step, clear = clear
  ; arr - array of data
  ; Q - number of valid elements in array
  ; item - item to be added
  ; step - size of superfluous elements of array
  ; /clear - initialize arr and Q (if they exist before)
  if keyword_set(clear)  then begin
    if n_elements(arr) NE 0 then tmp = temporary(arr)
    Q = 0L
    return
  endif
  Qitem = n_elements(item)
  Qarr = n_elements(arr)
  if Qarr EQ 0 then begin
    arr = [item,replicate(item[0],step)]                ; first
add to arr
    Q = Qitem
  endif else begin
```

```
    if Qarr GE (Q+Qitem) then $
      arr[Q:Q+Qitem-1] = item $                    ; item fit
in arr
    else $
      arr = [arr[0:Q-1],item,replicate(item[0],step)] ; item do not
fit in arr, extending
    Q = Q + Qitem
  endelse
end
```

AddItem usage:

```
  AddItem, arr, Qarr, /clear

  while ... do begin
    AddItem, arr, Qarr, item, 10000L
  endwhile

  arr = temporary(arr[0:Qarr-1])
```

You can rewrite AddItem as object if you will. I guess it is done already by someone.

What is interesting for me is file version of this procedure. Sometimes I need file that keeps different data arrays. I need to add, insert, and delete items from arrays in this file. And it should be relatively fast. I understand that this task is solved by database management systems but often using DBMS seems very excessive. And it would be nice to have save/restore style of parameters setting. Does anyone have written something like that?

Regards,
Altyntsev Dmitriy
