Subject: Re: Array concatenation
Posted by FL on Wed, 07 Jun 2006 16:12:56 GMT
View Forum Message <> Reply to Message

Hi,

concatenation is much faster in FL. I can't help showing a comparison.
Take this simple test program:

```
n=10l
for i=1,8 do begin
   a=[0l]
   t=systime(1)
   for j=1l,n do a=[a,j]
   print, n, systime(1)-t
   n *= 10l
endfor
end
```

On an Opteron 142 linux system FL prints:

```
        10   1.5974045e-05
       100   6.7949295e-05
      1000   0.00091814995
     10000     0.051065922
    100000      0.12665701
   1000000       1.2369611
  10000000       12.450622
 100000000       123.65290
```

In IDL 6.2 I have set i=1,5 (for obvious reasons :-), and the output is:

```
        10   1.4066696e-05
       100   7.3909760e-05
      1000   0.00089907646
     10000     0.045171022
    100000       22.684564
```

(FL has its own memory allocator and allocation strategy, this is the
reason of the big difference.)

regards,
lajos

On Wed, 7 Jun 2006, Paul Van Delst wrote:

> Let me second David's statement. For many points, array concatenation is s..l..o..w. In my
> early IDL days I improved the run time of a procedure (to read a 100,000's->millions of
> points) from minutes[*] to effectively instantaneous by switching from concatenation to
> allocate a block and then reallocate (or trim) extra space as required. Now it takes
> longer to output the number of points read rather than reading them.
>
> For a small number of points array concatenation is great, but it does not scale well.
>
> paulv
>
> [*] When I had to read many, many of the same type of files I could walk down the street
> to get a coffee before it had finished.
>
> --
> Paul van Delst Ride lots.
> CIMSS @ NOAA/NCEP/EMC Eddy Merckx
> Ph: (301)763-8000 x7748
> Fax:(301)763-8545
>
