Subject: Re: Where vs Histogram vs ??
Posted by R.G. Stockwell on Thu, 17 Oct 2002 13:18:35 GMT
View Forum Message <> Reply to Message

Andrew Cool wrote:
> Hello All,
> At the moment I'm doing something like this :-
>
> start_year = 2000
> end_year = 2002
> start_day = 120
> end_day = 133
> start_half_hr = 0
> end_half_hr = 47
> WRF = 1
> FREQ = 2
> start_beam = 0
> end_beam = 3
> nominated_parameter = 2
>
> index = Where(!database.year GE start_year AND $
> !database.year LE end_year AND $
> !database.day GE start_day AND $
> !database.day LE end_day AND $
> !database.beam GE start_beam AND $
> !database.beam LE end_beam AND $
> !database.half_hr GE start_half_hr AND $
> !database.half_hr LE end_half_hr AND $
> !database.WRF EQ WRF AND $
> !database.FREQ EQ FREQ AND $
> !database.parameter(nominated_parameter) NE
> bad_data_value)
>

A quick suggestion, change year, day, and half_hr into
julian day number, and thus reduce three searches into one.

index = Where((!database.time GE start_time) and (!database.time LT end_time),count)

BUt, if you REALLY want eye crossing speed, don't search on the array of structures.
Create an array of julian days (of the same size as database) and search that
for the index with which to use on the database array.
Here are the time differences, the code snippet that produced it is below.

IDL> .GO
time elapsed for where function array of struct 0.14732301
time elapsed for where function array of struct 0.036754966

DATABASE   STRUCT   = -> <Anonymous> Array[999999]

Another suggestion, you could always make your data base
a "real" database and use SQL style queries. This would
require some programming outside IDL (or getting Dataminer, etc.)

Cheers,
bob stockwell

PS
A small note, you use "GE start" and "LE end", so you may possibly
include the "end" in two subsets of the data (i.e. as the highest
in one subset,and as the lowest in the next subset). You might
want to look at using "GE start" and "LT end" (note LT rather than
LE). This would only be a problem if you are counting through
starttimes in a loop or something like that.

; START CODE SNIPPET THAT COMPARES A WHERE OF an array of
; structures, vs a simple array

```
data_st = {YEAR      : 0          ,$
                DAY       : 0          ,$ ; 136 days over 12 years
                HALF_HR   : 0          ,$ ; 0..47
                RANGE_IDX : 0          ,$ ; 0..267
                WRF       : 0B         ,$ ; 3 possible values
                FREQ      : 0B         ,$ ; 4 possible values
                BEAM      : 0B         ,$ ; 4 possible values
                PAD       : 0B         ,$ ; Padding to align byte
                Parameter : FLTARR(5)}
len = 999999L

database = Replicate(data_st,len)
array = dblarr(len)

for i = 0L,len-1 do begin
 database[i].year = (randomn(seed,1))[0]
 array[i] =  (randomn(seed,1))[0]
endfor
```

```
t0 = systime(1)
w = where(database.year gt 0.5,count)
print,'time elapsed for where function array of struct',-t0 + systime(1)

t0 = systime(1)
w = where(array gt 0.5,count)
print,'time elapsed for where function array of struct',-t0 + systime(1)
```
