Subject: Where vs Histogram vs ??
Posted by Andrew Cool on Wed, 16 Oct 2002 23:05:06 GMT
View Forum Message <> Reply to Message

Hello All,

I have a structure defined as :-

```
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
boundaries
           Parameter : FLTARR(5)}
```

Replicate that a few times :-

```
database = Replicate(data_st,15425228)
```

Data is plugged into this variable by reading from a file, and then converting
database to a system variable, !database, so that it survives intact just about
anything bar a .reset_session. Saves a lot of time recreating & reloading the database.

Roughly speaking, a third of the data is for any given WRF (waveform repetition frequency),
a quarter is at any given frequency, and a quarter is in each of the 4 possible beams.
Or, for any given day, the data is split over 4 beams, and cycled over 4 frequencies and
3 WRF's.

I need to be able to search this entire database and pull out a nominated parameter
value based on year,day, half_hr, range_idx, WRF, freq and beam and parameter.

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)


   This takes about 10-12 minutes on sizeable Alpha box running OpenVMS
(IDL v5.4)
   if working through the entire database for all 4 beams.

   To then plot each beam, there's a further loop of Where's to
subindex each
   particular beam out of index. The beam plots are either by UT or
range.


   Is there a quicker way than the above monsterous Where statement?

   I've browsed the Histogram tut on David Fanning's site, and rapidly
found
   my eyes glazing over. Can Histogram help here? Perhaps multiple
nested
   Histograms? David's SetUnion or SetIntersection, maybe?


   Any ideas appreciated,

   Andrew

 ------------------------------------------------------------ ----------
```

Andrew D. Cool
Electromagnetics & Propagation Group
Intelligence, Surveillance & Reconnaissance Division
Defence Science & Technology Organisation
PO Box 1500, Edinburgh
South Australia 5111
Phone : 061 8 8259 5740 Fax : 061 8 8259 6673
Email : andrew.cool@dsto.defence.gov.au

Subject: Re: Where vs Histogram vs ??
Posted by Pavel A. Romashkin on Thu, 17 Oct 2002 23:22:45 GMT
View Forum Message <> Reply to Message

I can definitely echo Bob's suggestion to use index for searching. Don't use structure fields. Using a database would like ly be better yet; I think MS Access Jet should be reasonably fast with 15 mln records.
Good luck,
Pavel

Andrew Cool wrote:
>
> Hello All,
>
> I have a structure defined as :-
>
> 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
> boundaries
> Parameter : FLTARR(5)}
>
>
> Replicate that a few times :-
>
> database = Replicate(data_st,15425228)

Subject: Re: Where vs Histogram vs ??
Posted by Andrew Cool on Fri, 18 Oct 2002 01:13:26 GMT
View Forum Message <> Reply to Message

"Pavel A. Romashkin" wrote:
>
> I can definitely echo Bob's suggestion to use index for searching. Don't
> use structure fields. Using a database would like ly be better yet; I
> think MS Access Jet should be reasonably fast with 15 mln records.
> Good luck,
> Pavel

Hi Pavel,

I'm confined to running this under OpenVMS, so MS Access probably
ain't the cure here. ;-)

Andrew

>
> Andrew Cool wrote:
>>
>> Hello All,
>>
>> I have a structure defined as :-
>>
>> 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
>> boundaries
>> Parameter : FLTARR(5)}
>>
>>
>> Replicate that a few times :-
>>
>> database = Replicate(data_st,15425228)

--

---------- ----------
Andrew D. Cool .->-.
Electromagnetics & Propagation Group '-<-'
Intelligence, Surveillance & Reconnaissance Division Transmitted on

Defence Science & Technology Organisation 100% recycled
PO Box 1500, Edinburgh electrons
South Australia 5111

Phone : 061 8 8259 5740 Fax : 061 8 8259 6673
Email : andrew.cool@no-spam.dsto.defence.gov.au
Please remove the no-spam from my email address to reply ;-)
---------- ----------

Subject: Re: Where vs Histogram vs ??
Posted by Craig Markwardt on Fri, 18 Oct 2002 03:21:44 GMT
View Forum Message <> Reply to Message

Andrew Cool <andrew.cool@dsto.defence.gov.au> writes:
> 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)

I'll be the broken record, and agree with everybody else that
structure access is slow.

I think this could be much faster to access as *gasp* a common block.
If each parameter were an array variable in a common, then you would
save the considerable time involved in extracting the fields from the

structures in each comparison.

You also definitely want to make a field which is Julian day, since
that reduces the number of comparisons for the date/time from three to
one, and I think it will save space. Or, are you *really* interested
in data from days 120-133 in years 2000, 2001 and 2002 combined?

Finally, if you can, try to thin the array first by applying the most
stringent selection. For example, if you are only looking in a narrow
date range, then first extract only those records fromt the date
range, then go back and apply the other criteria.

With 15 million samples, anything you do will take quite a bit of
time. However, I regularly do operations on 3 million sample arrays
and it isn't *too* bad.

Hope that helps!

Craig

--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------

Subject: Re: Where vs Histogram vs ??
Posted by Pavel A. Romashkin on Fri, 18 Oct 2002 20:49:53 GMT
View Forum Message <> Reply to Message

Allright, so we need a solution in IDL.
At this array size the slowest portion of the process is not the WHERE
function as far as I can tell. It is memory reallocation for the main
array and for the temporary index arrays that IDL creates. Therefore I
can suggest trying the following approach.
Allocate it all only once:

```
ind = ptrarr(n_tags(data_St)
for i = 0, n_tags(data_st) do ind[i] = ptr_new(data_st.(i))
```

This does take a little time to execute.
Now you have a static index of all fields. Of course, you have used
twice the memory but given the relatively small data volume it seems ok.
On my machine the RAM used by both structure and pointer index barely
reaches 1010 Mb, so I have room for further calculations.
Now, you can search the pointer array elements using WHERE, and it is

fairly fast. I tested it on my machine; the same WHERE statement you show took 56 s for the structure array, but only 6 s using the index pointer array. Further speed increase will be achievable if you merged timestamps into one field, as others recommended; some flexibility in querying would be, however, lost. And of course you can use the resulting INDEX to subscript your original structire array.
Hope this helps,
Pavel

****

```
index = Where(* ind[0] GE 596   AND $
  * ind[0] LE 2000     AND $ ; yr
 * ind[1]  GE 15    AND $ ; day
  * ind[1]  LE 52      AND $
  * ind[6] GE 6   AND $ ;beam
  * ind[6] LE 5     AND $
  * ind[2] GE 15    AND $ ;half hr
 * ind[2] LE 5    AND $
 * ind[4] EQ 5    AND $ ;WRF
  * ind[5] EQ 5  AND $ ;Freq
  (* ind[8])[0] NE -555)
```

****

Andrew Cool wrote:
>
> "Pavel A. Romashkin" wrote:
>>
>> I can definitely echo Bob's suggestion to use index for searching. Don't
>> use structure fields. Using a database would like ly be better yet; I
>> think MS Access Jet should be reasonably fast with 15 mln records.
>> Good luck,
>> Pavel
>
> Hi Pavel,
>
> I'm confined to running this under OpenVMS, so MS Access probably
> ain't the cure here. ;-)
>
> Andrew
>

Subject: Re: Where vs Histogram vs ??
Posted by Andrew Cool on Tue, 22 Oct 2002 00:59:21 GMT
View Forum Message <> Reply to Message

Hi Pavel,

"Pavel A. Romashkin" wrote:
>
> Allright, so we need a solution in IDL.
> At this array size the slowest portion of the process is not the WHERE
> function as far as I can tell. It is memory reallocation for the main
> array and for the temporary index arrays that IDL creates. Therefore I
> can suggest trying the following approach.
> Allocate it all only once:
>
> ind = ptrarr(n_tags(data_St)

Should this be something like

ind = ptrarr(N_Tags(data_st) * 15425228L)

given that N_Tags(data_st) only returns a value of 9, which concurs
with Tag_Names(data_st), such that we effectively have

ind = ptrarr(9 * 15425228L)

Now that's a scary sized ptrarr.

Given that you say :-

> On my machine the RAM used by both structure and pointer index barely
> reaches 1010 Mb, so I have room for further calculations.

and assuming you've used the figure of 15425228, then I obviously
don't
understand your example... ;-)

Would you mind elaborating a bit, in words of one brain cell or less?

Thanks,

Andrew

---------- ----------
Andrew D. Cool
Electromagnetics & Propagation Group
Intelligence, Surveillance & Reconnaissance Division
Defence Science & Technology Organisation
PO Box 1500, Edinburgh
South Australia 5111

Phone : 061 8 8259 5740 Fax : 061 8 8259 6673

Email : andrew.cool@no-spam.dsto.defence.gov.au
Please remove the no-spam from my email address to reply.

Subject: Re: Where vs Histogram vs ??
Posted by Andrew Cool on Tue, 22 Oct 2002 00:59:45 GMT
View Forum Message <> Reply to Message

Hi Pavel,

"Pavel A. Romashkin" wrote:
>
> Allright, so we need a solution in IDL.
> At this array size the slowest portion of the process is not the WHERE
> function as far as I can tell. It is memory reallocation for the main
> array and for the temporary index arrays that IDL creates. Therefore I
> can suggest trying the following approach.
> Allocate it all only once:
>
> ind = ptrarr(n_tags(data_St)

Should this be something like

ind = ptrarr(N_Tags(data_st) * 15425228L)

given that N_Tags(data_st) only returns a value of 9, which concurs
with Tag_Names(data_st), such that we effectively have

ind = ptrarr(9 * 15425228L)

Now that's a scary sized ptrarr.

Given that you say :-

> On my machine the RAM used by both structure and pointer index barely
> reaches 1010 Mb, so I have room for further calculations.

and assuming you've used the figure of 15425228, then I obviously
don't
understand your example... ;-)

Would you mind elaborating a bit, in words of one brain cell or less?

Thanks,

Andrew

---------- ----------
Andrew D. Cool
Electromagnetics & Propagation Group
Intelligence, Surveillance & Reconnaissance Division
Defence Science & Technology Organisation
PO Box 1500, Edinburgh
South Australia 5111

Phone : 061 8 8259 5740 Fax : 061 8 8259 6673
Email : andrew.cool@no-spam.dsto.defence.gov.au
Please remove the no-spam from my email address to reply.
---------- ----------

Subject: Re: Where vs Histogram vs ??
Posted by Stein Vidar Hagfors H[1] on Tue, 22 Oct 2002 15:58:41 GMT
View Forum Message <> Reply to Message

Andrew Cool <andrew.cool@dsto.defence.gov.au> writes:

> Hello All,
[...snip...]
> Roughly speaking, a third of the data is for any given WRF (waveform
> repetition frequency),
> a quarter is at any given frequency, and a quarter is in each of the 4
> possible beams.
> Or, for any given day, the data is split over 4 beams, and cycled over
> 4 frequencies and
> 3 WRF's.
[...snip...]
> 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)
[...]
```

Given the above, could you perhaps try a multi-stage selection, e.g.,

```
  wrf_ok = !database.WRF EQ WRF
  wrf_freq_ok = !database.FREQ EQ FREQ AND temporary(wrf_ok)
  ;; By now you should have 1/12th of the data left!
  ;; Don't know how many bad_data_values you expect, the next one might
  ;; not gain much:
  wrf_freq_good =!database.parameter(nominated_parameter) NE bad_data_value $
               AND temporary(wrf_freq_ok)

  index1 = where(wrf_freq_good)

  ;; Build a new database on this subset (smaller than 1/12th),
  ;; continue with the rest of your searches...
```

Otherwise, I'd say that going from year/day/half_hr to Julian Day (modified to fit into a smaller data type, perhaps, by multiplying JD with 48 half-hours & subtracting earliest possible epoch?) is good advice, as is the multiple-array (instead of structure) approach.

However, as with many other problems of this type, the "killer" approach would be staying with a structure, using a DLM that goes through the data once, producing a single byte array with 0B/1B given input start/end times, beams, WRF, FREQ and Nominated-parameter!

There's no way IDL can optimize these statements the way a C programmer would do. Depending of course on the number of times you expect to do these selections over your project lifetime, I'd say writing a DLM may be a good investment of time!

--

Stein Vidar Hagfors Haugan
ESA SOHO SOC/European Space Agency Science Operations Coordinator for SOHO

NASA Goddard Space Flight Center, Email: shaugan@esa.nascom.nasa.gov
Mail Code 682.3, Bld. 26, Room G-1, Tel.: 1-301-286-9028/240-354-6066
Greenbelt, Maryland 20771, USA. Fax: 1-301-286-0264

Subject: Re: Where vs Histogram vs ??
Posted by Andrew Cool on Wed, 23 Oct 2002 00:26:23 GMT
View Forum Message <> Reply to Message

Stein Vidar Hagfors Haugan wrote:
>
> Andrew Cool <andrew.cool@dsto.defence.gov.au> writes:
>
>> Hello All,
>> 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)
> [...]
>
> Given the above, could you perhaps try a multi-stage selection, e.g.,
>
> wrf_ok = !database.WRF EQ WRF
> wrf_freq_ok = !database.FREQ EQ FREQ AND temporary(wrf_ok)
> ;; By now you should have 1/12th of the data left!
> ;; Don't know how many bad_data_values you expect, the next one might
> ;; not gain much:
> wrf_freq_good =!database.parameter(nominated_parameter) NE bad_data_value $
> AND temporary(wrf_freq_ok)
>
> index1 = where(wrf_freq_good)
>
> ;; Build a new database on this subset (smaller than 1/12th),
> ;; continue with the rest of your searches...
>

Hello Stein,

I think your multistage selection using "AND Temporary(prev_stage)"
is the way for me. I'm rather enamoured with the use of the structure
in this database, and reluctant to give it up without a fight. It just
makes it so easy to query the database from the command line as well as

programatically.

Although arrays would probably be faster, I'll settle for a V2 rather
than
a Saturn V if it means I can keep the structures.

Thanks to everyone for their suggestions!

Andrew

---------- ----------
Andrew D. Cool
Electromagnetics & Propagation Group
Intelligence, Surveillance & Reconnaissance Division
Defence Science & Technology Organisation
PO Box 1500, Edinburgh
South Australia 5111

Phone : 061 8 8259 5740 Fax : 061 8 8259 6673
Email : andrew.cool@dsto.defence.gov.au
---------- ----------

Subject: Re: Where vs Histogram vs ??
Posted by Andrew Cool on Wed, 23 Oct 2002 03:53:03 GMT
View Forum Message <> Reply to Message

Hi Stein,

The multistage method with the structure gives about a 13-15%
improvement, or from about 10 minutes down to about 8.5 minutes
for a run through the entire 15,425,228 records.

Cheers,

Andrew

Andrew Cool wrote:
>
>> Given the above, could you perhaps try a multi-stage selection, e.g.,
>>
>> wrf_ok = !database.WRF EQ WRF
>> wrf_freq_ok = !database.FREQ EQ FREQ AND temporary(wrf_ok)

>>   ;; By now you should have 1/12th of the data left!
>>   ;; Don't know how many bad_data_values you expect, the next one might
>>   ;; not gain much:
>>   wrf_freq_good =!database.parameter(nominated_parameter) NE bad_data_value $
>>   AND temporary(wrf_freq_ok)
>>
>>   index1 = where(wrf_freq_good)
>>
>>   ;; Build a new database on this subset (smaller than 1/12th),
>>   ;; continue with the rest of your searches...
>>
>
> Hello Stein,
>
>   I think your multistage selection using "AND Temporary(prev_stage)"
>   is the way for me. I'm rather enamoured with the use of the structure
>   in this database, and reluctant to give it up without a fight. It just
>   makes it so easy to query the database from the command line as well as
>   programatically.
>
>   Although arrays would probably be faster, I'll settle for a V2 rather
> than
>   a Saturn V if it means I can keep the structures.
>
>   Thanks to everyone for their suggestions!
>
>   Andrew
>
> -- -----------------
> Andrew D. Cool
> Electromagnetics & Propagation Group
> Intelligence, Surveillance & Reconnaissance Division
> Defence Science & Technology Organisation
> PO Box 1500, Edinburgh
> South Australia 5111
>
> Phone : 061 8 8259 5740   Fax : 061 8 8259 6673
> Email : andrew.cool@dsto.defence.gov.au
> -- -----------------

--

 -- -----------------
 Andrew D. Cool   .->-.
 Electromagnetics & Propagation Group   '-<-'
 Intelligence, Surveillance & Reconnaissance Division Transmitted on
 Defence Science & Technology Organisation 100% recycled

PO Box 1500, Edinburgh electrons
South Australia 5111

Phone : 061 8 8259 5740 Fax : 061 8 8259 6673
Email : andrew.cool@no-spam.dsto.defence.gov.au
Please remove the no-spam from my email address to reply ;-)
---------- ----------

Subject: Re: Where vs Histogram vs ??
Posted by Pavel A. Romashkin on Wed, 23 Oct 2002 17:29:45 GMT
View Forum Message <> Reply to Message

Hi Andrew,
Sorry for delaying the answer.
No, no, no. No. It needs to be just what it is. It will be an array of just 9 pointers. Each of them points to a vector (well, except for the last one which is a matrix), and as such is searchable quite quickly using WHERE.
You may notice that for an array of structures:

```
a = {a: 0, b: 0.0, c: fltarr(5)}
a = replicate(a, 1000)
help, a.(0)
;<Expression>    INT       = Array[1000]
help, a.(2)
;<Expression>    FLOAT     = Array[5, 1000]
```

Therefore, when you loop over just *fields* of a structure array, you get the contents of the entire array. In your case, this is perfect for indexing the data. I use this a lot - it allows to shift arrays throughout the entire structure array just as if it were a plain matrix or vector, and is just as fast.
As I said, you can basically do away with the sreucture array becasue now your 9-element pointer array contains everything the old structure array contained. In fact, yopu can dump the old array do free up some RAM, but that is not critical. Also, in a general case, you want only to include those fields in the ptr array that you use for searching, and then use the resulting index to extract the data from the original structure array.
Regarding memory use:

```
; Here, A is an array of structures of exactly your type of size 16 mln.
; I have nothing else in the IDL session.
IDL> help, /mem
heap memory used:  512482366, max:  512483544, gets:     1719, frees:
 1167
IDL> ind = ptrarr(n_tags(a)
```

IDL> for i = 0, n_tags(a)-1 do ind[i] = ptr_new(a.(i))
; The above takes less than a minute
IDL> help, /mem
heap memory used: 1024484012, max: 1024484732, gets: 3656, frees:
 3093

As expected, the memory use doubles; if that's a problem, discard the
original array.

Hope this helps.
Pavel

Andrew Cool wrote:
>   Should this be something like
>
>   ind = ptrarr(N_Tags(data_st) * 15425228L)
>
>   given that N_Tags(data_st) only returns a value of 9, which concurs
>   with Tag_Names(data_st), such that we effectively have
>
>   ind = ptrarr(9 * 15425228L)
>
>
>   Now that's a scary sized ptrarr.
>
>   Given that you say :-
>
>> On my machine the RAM used by both structure and pointer index barely
>> reaches 1010 Mb, so I have room for further calculations.
>
>   and assuming you've used the figure of 15425228, then I obviously
> don't
>   understand your example... ;-)
>
>   Would you mind elaborating a bit, in words of one brain cell or less?
>
>   Thanks,
>
>   Andrew

Subject: Re: Where vs Histogram vs ??
Posted by Andrew Cool on Wed, 23 Oct 2002 22:31:22 GMT
View Forum Message <> Reply to Message

"Pavel A. Romashkin" wrote:
>
> Hi Andrew,

> Sorry for delaying the answer.
> No, no, no. No. It needs to be just what it is. It will be an array of
> just 9 pointers. Each of them points to a vector (well, except for the
> last one which is a matrix), and as such is searchable quite quickly
> using WHERE.
> You may notice that for an array of structures:
>
> a = {a: 0, b: 0.0, c: fltarr(5)}
> a = replicate(a, 1000)
> help, a.(0)
> ;<Expression>   INT   = Array[1000]
> help, a.(2)
> ;<Expression>   FLOAT   = Array[5, 1000]
>
> Therefore, when you loop over just *fields* of a structure array, you
> get the contents of the entire array. In your case, this is perfect for
> indexing the data. I use this a lot - it allows to shift arrays
> throughout the entire structure array just as if it were a plain matrix
> or vector, and is just as fast.
> As I said, you can basically do away with the sreucture array becasue
> now your 9-element pointer array contains everything the old structure
> array contained. In fact, yopu can dump the old array do free up some
> RAM, but that is not critical. Also, in a general case, you want only to
> include those fields in the ptr array that you use for searching, and
> then use the resulting index to extract the data from the original
> structure array.
> Regarding memory use:
>
> ; Here, A is an array of structures of exactly your type of size 16 mln.
> ; I have nothing else in the IDL session.
> IDL> help, /mem
> heap memory used: 512482366, max: 512483544, gets: 1719, frees:
> 1167
> IDL> ind = ptrarr(n_tags(a)
> IDL> for i = 0, n_tags(a)-1 do ind[i] = ptr_new(a.(i))
> ; The above takes less than a minute
> IDL> help, /mem
> heap memory used: 1024484012, max: 1024484732, gets: 3656, frees:
> 3093
>
> As expected, the memory use doubles; if that's a problem, discard the
> original array.
>
> Hope this helps.
> Pavel

Hi Pavel,

I doubt that I'd be able to hold both the structure and ptrarr in memory
at any one time - our VMS SYSMAN1 would have conniptions if I asked to
increase my user quotas anymore - as it is I totally hog one Alpha server
when this code runs...

But you seem pretty sure of your onions on this. I'll give it a whirl and
to see if Pavel > Stein Vidar!

Thanks,

Andrew

Andrew D. Cool .->-.
Electromagnetics & Propagation Group '-<-'
Intelligence, Surveillance & Reconnaissance Division Transmitted on
Defence Science & Technology Organisation 100% recycled
PO Box 1500, Edinburgh electrons
South Australia 5111

Phone : 061 8 8259 5740 Fax : 061 8 8259 6673
Email : andrew.cool@dsto.defence.gov.au

Subject: Re: Where vs Histogram vs ??
Posted by Stein Vidar Hagfors H[2] on Thu, 24 Oct 2002 13:33:23 GMT
View Forum Message <> Reply to Message

Andrew Cool <andrew.cool@dsto.defence.gov.au> writes:

> Hi Pavel,
>
>
> I doubt that I'd be able to hold both the structure and ptrarr in
> memory at any one time - our VMS SYSMAN1 would have conniptions if I asked
> to increase my user quotas anymore - as it is I totally hog one Alpha server
> when this code runs...
>
> But you seem pretty sure of your onions on this. I'll give it a whirl
> and to see if Pavel > Stein Vidar!

[...]

If you're making it into a competition, I won't concede defeat until you've gained a full order of magnitude in speed *or* tried a DLM! (From your earlier statements, you have a pretty fixed structure definition, so handling the structure needn't be fully general).

--
---------- ----------
Stein Vidar Hagfors Haugan
ESA SOHO SOC/European Space Agency Science Operations Coordinator for SOHO

NASA Goddard Space Flight Center, Tel.: 1-301-286-9028
Mail Code 682.3, Bld. 26, Room G-1, Cell: 1-240-354-6066
Greenbelt, Maryland 20771, USA. Fax: 1-301-286-0264
---------- ----------

Subject: Re: Where vs Histogram vs ??
Posted by Pavel A. Romashkin on Thu, 24 Oct 2002 20:36:53 GMT
View Forum Message <> Reply to Message

Andrew,
Well, that is another issue. Its too bad you can't just move the data over to a PC or a Mac. Either platform would handle this data size with ease. Doing searches as you specified on my Mac (which is not top of the line anymore) is slow (6-7 s) but not unbearable. I do have 1.5 Gb or RAM in it but this is nothing unusual these days. I bet David has 10 Gb in his screamer - how else he could make his nested, sprawling, self aware and self reproducing objects to survive? :-) I wonder, did David write an object by now that actually writes optimized code for him?...
Hope you can make it work for you!
Pavel

Andrew Cool wrote:
>
> Hi Pavel,
>
> I doubt that I'd be able to hold both the structure and ptrarr in
> memory
> at any one time - our VMS SYSMAN1 would have conniptions if I asked to
> increase my user quotas anymore - as it is I totally hog one Alpha
> server
> when this code runs...
>
> But you seem pretty sure of your onions on this. I'll give it a whirl
> and
> to see if Pavel > Stein Vidar!

Subject: Re: Where vs Histogram vs ??
Posted by Pavel A. Romashkin on Mon, 28 Oct 2002 19:01:22 GMT
View Forum Message <> Reply to Message

Stein Vidar Hagfors Haugan wrote:

> If you're making it into a competition, I won't concede defeat until you've
> gained a full order of magnitude in speed *or* tried a DLM! (From your earlier
> statements, you have a pretty fixed structure definition, so handling the
> structure needn't be fully general).

Aha! Stein Vidar is already feeling nervous :-)
Order of magnitude! That wouldn't be *defeat*, that would be a complete leveling with the ground! Which of course would not happen.
I never meant to get into a competition. Just wanted something that would work and take the least amount of code. Preferably less than 5 lines :-)
Cheers,
Pavel