Subject: Re: indexing over structure tags
Posted by Paul van Delst on Thu, 19 Apr 2001 17:00:20 GMT
View Forum Message <> Reply to Message

Randall Skelton wrote:
>
> Hello,
>
> Imagine someone has a structure of structures...
>
> ; define the basic structure for each
> sm_struc = {basic_struct, comment: ' ', values: fltarr(nlev)}
>
> ; define the large structure
> data = {big_struct, so4: sm_struc, co2: sm_struc, hcl: sm_struc}
>
> The IDL manual describes how to make an array of the tags in a structure
> using:
>
> ; get the names of the tags
> names = tag_names(data)
>
> so that names = [so4, co2, hcl].
>
> That is all fine. But is it possible to index over the tag names with a
> for-loop?
>
> i.e. for i=0, n_elements(names)-1 do data.names[i].values = i
>   ^^^^^^^^^^^^^^^^^^^^

Maybe

data.names[i].(j) where j=0->N_TAGS(data.names[i])-1 ??

This works on "single-level" structures but I don't see why it wouldn't work on nested
ones.

paulv

--
Paul van Delst   A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP   Drink deep, or taste not the Pierian spring;
Ph: (301)763-8000 x7274 There shallow draughts intoxicate the brain,
Fax:(301)763-8545   And drinking largely sobers us again.
paul.vandelst@noaa.gov   Alexander Pope.

Subject: Re: indexing over structure tags
Posted by Paul van Delst on Thu, 19 Apr 2001 17:05:43 GMT
View Forum Message <> Reply to Message

Paul van Delst wrote:
>
> Randall Skelton wrote:
>>
>> Hello,
>>
>> Imagine someone has a structure of structures...
>>
>> ; define the basic structure for each
>> sm_struc = {basic_struct, comment: ' ', values: fltarr(nlev)}
>>
>> ; define the large structure
>> data = {big_struct, so4: sm_struc, co2: sm_struc, hcl: sm_struc}
>>
>> The IDL manual describes how to make an array of the tags in a structure
>> using:
>>
>> ; get the names of the tags
>> names = tag_names(data)
>>
>> so that names = [so4, co2, hcl].
>>
>> That is all fine. But is it possible to index over the tag names with a
>> for-loop?
>>
>> i.e. for i=0, n_elements(names)-1 do data.names[i].values = i
>> ^^^^^^^^^^^^^^^^^^^^
>
> Maybe
>
> data.names[i].(j) where j=0->N_TAGS(data.names[i])-1 ??
>
> This works on "single-level" structures but I don't see why it wouldn't work on nested
> ones.

Oops, I guess you really wanted something like

data.(i).values

where i=0->number of molecular names

'Scusi

paulv

--
Paul van Delst A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP Drink deep, or taste not the Pierian spring;
Ph: (301)763-8000 x7274 There shallow draughts intoxicate the brain,
Fax:(301)763-8545 And drinking largely sobers us again.
paul.vandelst@noaa.gov Alexander Pope.

Subject: Re: indexing over structure tags
Posted by tam on Thu, 19 Apr 2001 17:16:38 GMT
View Forum Message <> Reply to Message

Just use the numeric tag syntax for structures.

```
  for i=0,n_tags(data)-1 do begin
     data.(i).values = .. some expresion ...
  endfor
```

where the (i) indicates use the i'th element of the structure.
So you don't need to use the names at all.

Regards,
Tom McGlynn

Randall Skelton wrote:
>
> Hello,
>
> Imagine someone has a structure of structures...
>
> ; define the basic structure for each
> sm_struc = {basic_struct, comment: ' ', values: fltarr(nlev)}
>
> ; define the large structure
> data = {big_struct, so4: sm_struc, co2: sm_struc, hcl: sm_struc}
>
> The IDL manual describes how to make an array of the tags in a structure
> using:
>
> ; get the names of the tags
> names = tag_names(data)
>
> so that names = [so4, co2, hcl].
>
> That is all fine. But is it possible to index over the tag names with a
> for-loop?
>
> i.e. for i=0, n_elements(names)-1 do data.names[i].values = i

> ^^^^^^^^^^^^^^^^^^^^
>
> where IDL determines what the appropriate label 'data.name[i].values' is.
>
> Thanks,
> Randall

Subject: Re: indexing over structure tags
Posted by Randall Skelton on Thu, 19 Apr 2001 17:24:09 GMT
View Forum Message <> Reply to Message

Thanks Tom... I did think of that. However, in this particular case there is some merit in having the name of the structure be a useful and human readable tag. Nobody would be happy trying to remember yet another arbitrary numbering scheme for molecules when they'd rather just type the name ;) My suspicion is that there isn't an easy way to do what I want...

Randall

On Thu, 19 Apr 2001, tam wrote:

> Just use the numeric tag syntax for structures.
>
> for i=0,n_tags(data)-1 do begin
> data.(i).values = .. some expresion ...
> endfor
>
> where the (i) indicates use the i'th element of the structure.
> So you don't need to use the names at all.
>
> Regards,
> Tom McGlynn
>

Subject: Re: indexing over structure tags
Posted by Vapuser on Thu, 19 Apr 2001 17:36:59 GMT
View Forum Message <> Reply to Message

Randall Skelton <rhskelto@atm.ox.ac.uk> writes:

> Hello,
>
> Imagine someone has a structure of structures...
>
> ; define the basic structure for each

>   sm_struc = {basic_struct, comment: ' ', values: fltarr(nlev)}
>
>   ; define the large structure
>   data = {big_struct, so4: sm_struc, co2: sm_struc, hcl: sm_struc}
>
> The IDL manual describes how to make an array of the tags in a structure
> using:
>
>   ; get the names of the tags
>   names = tag_names(data)
>
> so that names = [so4, co2, hcl].
>
> That is all fine. But is it possible to index over the tag names with a
> for-loop?
>
> i.e.   for i=0, n_elements(names)-1 do data.names[i].values = i
>   ^^^^^^^^^^^^^^^^^^^^

>
> where IDL determines what the appropriate label 'data.name[i].values' is.
>
> Thanks,
> Randall
>

  data.(i) to iterate over that tags of data, and
  data.(i).(j) to iterate over the tags of data.(i)

If each tag is an array, rather than a scalar structure instance, you
might (probably?) have to pull it out, like this

IDL> junk={foo, a:0, b:0l, c:0.0}
IDL> bar={bar, a:replicate({foo},3),b:replicate({foo},4)}
IDL> help,bar,/struct
** Structure BAR, 2 tags, length=84:
   A   STRUCT   -> FOO Array[3]
   B   STRUCT   -> FOO Array[4]

IDL> .run
- FOR i=0,1 DO BEGIN
-   FOR j=0,2 DO BEGIN
-    tmp=bar.(i).(j) ;<--- I couldn't find a way to do this inline,
;   ; but there might be one that I'm missing
-    FOR k=0,n_elements(tmp)-1 DO tmp[k]=(2^(i+1))*(3^(j+1))+k
-    bar.(i).(j)=tmp
-   endfor

```
- endfor
- end
% Compiled module: $MAIN$.
IDL>  print,bar
{
  a = {       6         18      54.0000}
      {       7         19      55.0000}
      {       8         20      56.0000}

  b = {      12         36      108.000}
      {      13         37      109.000}
      {      14         38      110.000}
      {      15         39      111.000}

}
```

whd

--
William Daffer: 818-354-0161: William.Daffer@jpl.nasa.gov

Subject: Re: indexing over structure tags
Posted by Bernard Puc on Thu, 19 Apr 2001 17:50:15 GMT
View Forum Message <> Reply to Message

Randall Skelton wrote:
>
> Thanks Tom... I did think of that. However, in this particular case there
> is some merit in having the name of the structure be a useful and human
> readable tag. Nobody would be happy trying to remember yet another
> arbitrary numbering scheme for molecules when they'd rather just type the
> name ;) My suspicion is that there isn't an easy way to do what I want...
>
> Randall
>
> On Thu, 19 Apr 2001, tam wrote:
>
>> Just use the numeric tag syntax for structures.
>>
>>   for i=0,n_tags(data)-1 do begin
>>   data.(i).values = .. some expresion ...
>>   endfor
>>
>> where the (i) indicates use the i'th element of the structure.
>> So you don't need to use the names at all.
>>

>> Regards,
>> Tom McGlynn
>>

Perhaps you misunderstood...there are two ways of accessing a tag within
a structure.

data = {lat:fltarr(10), lng:fltarr(10), sig:fltarr(10)}

Method 1:
data.lat
data.lng
data.sig

Method 2:
data.(0)
data.(1)
data.(2)

Either method accesses the same variables.

--
Bernard Puc AETC, INC.
bpuc@va.aetc.com 1225 Jefferson Davis Highway #800
(703) 413-0500 Arlington, VA 22202

Subject: Re: indexing over structure tags
Posted by Paul van Delst on Thu, 19 Apr 2001 18:08:29 GMT
View Forum Message <> Reply to Message

Randall Skelton wrote:
>
> Thanks Tom... I did think of that. However, in this particular case there
> is some merit in having the name of the structure be a useful and human
> readable tag. Nobody would be happy trying to remember yet another
> arbitrary numbering scheme for molecules when they'd rather just type the
> name ;) My suspicion is that there isn't an easy way to do what I want...

Maybe more context is needed to solve your problem .... the user of the code *shouldn't*
have to remember the (not so) arbitrary numbering scheme - the user would type in a
molecule name (or names). How your code deals with searching the human readable tagnames
is a different matter, no?

e.g.

; define the basic structure for each

```
  sm_struc = {basic_struct, comment: ' ', values: fltarr(nlev)}

  ; define the large structure
  data = {big_struct, so4: sm_struc, co2: sm_struc, hcl: sm_struc}

  ; get the names of the tags
  names = tag_names(data)
```

so that names = [so4, co2, hcl].

Say the user requests data for 'so4' and 'hcl' so how about

```
  user_request = ['so4', 'hcl']
  n_requests = N_ELEMENTS( user_request )

  FOR i = 0, n_requests - 1 DO BEGIN
    tag_number = (WHERE( user_request[i] EQ names ))[0]  ; <-- assume this always succeeds
    data_to_get = data.(tag_number).values
    IF ( i EQ 0 ) THEN $
      data_to_return = data_to_get $
    ELSE $
      data_to_return = [ [ data_to_return ], [ data_to_get ] ]
  ENDFOR

  RETURN, data_to_return
```

or something like that? As it is above might not work for plucking out structures, but that's a detail. So is the concatenation build of the data_to_return. Should be o.k. for small arrays tho'.

paulv

--
Paul van Delst A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP Drink deep, or taste not the Pierian spring;
Ph: (301)763-8000 x7274 There shallow draughts intoxicate the brain,
Fax:(301)763-8545 And drinking largely sobers us again.
paul.vandelst@noaa.gov Alexander Pope.

Subject: Re: indexing over structure tags
Posted by tam on Thu, 19 Apr 2001 19:23:50 GMT

View Forum Message <> Reply to Message

Hmmm.... Well assuming that you want to iterate (for whatever reason) over a set of names rather than a set of indices how about.

```
names = tagnames(data)
for i=0,n_tags(data)-1 do begin  ; Or any other kind of loop
   name = ... entered somehow by the user perhaps ...
   w=where(name eq names[i]);
   if (w(0) ne -1) then begin
      ind = w[0]
      data.(ind).values = ...
....
```

It's easy enough to translate from the variable name to the variable index -- and the index may itself be a variable.

Tom

Randall Skelton wrote:
>
> Thanks Tom... I did think of that. However, in this particular case there
> is some merit in having the name of the structure be a useful and human
> readable tag. Nobody would be happy trying to remember yet another
> arbitrary numbering scheme for molecules when they'd rather just type the
> name ;) My suspicion is that there isn't an easy way to do what I want...
>
> Randall
>
> On Thu, 19 Apr 2001, tam wrote:
>
>> Just use the numeric tag syntax for structures.
>>
>> for i=0,n_tags(data)-1 do begin
>> data.(i).values = .. some expresion ...
>> endfor
>>
>> where the (i) indicates use the i'th element of the structure.
>> So you don't need to use the names at all.
>>
>> Regards,
>> Tom McGlynn
>>

Subject: Re: indexing over structure tags
Posted by tam on Thu, 19 Apr 2001 19:25:38 GMT
View Forum Message <> Reply to Message

tam wrote:
>
...
> if (w(0) ne -1) then begin

> ind = w[0]
> data.(ind).values = ...
> ....
>

Mea culpa for the w(0) followed on the next line by w[0]. Just trying
to induce maximal confusion.

Tom

Subject: Re: indexing over structure tags
Posted by R.Bauer on Tue, 24 Apr 2001 07:45:13 GMT
View Forum Message <> Reply to Message

Randall Skelton wrote:
>
> Thanks Tom... I did think of that. However, in this particular case there
> is some merit in having the name of the structure be a useful and human
> readable tag. Nobody would be happy trying to remember yet another
> arbitrary numbering scheme for molecules when they'd rather just type the
> name ;) My suspicion is that there isn't an easy way to do what I want...
>
> Randall
>

Dear Randall,

I will show you an easy way, how to access any level of a structure
by incrementing of a an array.

may be we have a structure like

pi={name:'p.mustermann'}
global={pi:pi}
n2o={param:fltarr(10),short_name:'N2O',long_name:'mixing
ratio',units:'ppb',flag:'NONE'}
time={param:fltarr(10),short_name:'time',long_name:'time
[UT]',units:'seconds since 2000-01-01 00:00:00 UTC', flag:'NONE'}

icgs={!global:global,time:time,n2o:n2o}

more attribute of this data structure are explained by my publication.
http://www.fz-juelich.de/zb/text/publikation/juel3786.html

The following routine creates an array of tagnames and an

array of pointers of any submitted structure
The return value of this function is n_elements(names)
and the keywords names and ptr_values

```
 n=struct2names_and_ptrs(icgs,names=names,ptr_values=ptr_valu es)
PRINT,n
IDL> 11

IDL> PRINT,names
!GLOBAL.PI.NAME TIME.PARAM TIME.SHORT_NAME TIME.LONG_NAME TIME.UNITS
TIME.FLAG
 N2O.PARAM N2O.SHORT_NAME N2O.LONG_NAME N2O.UNITS N2O.FLAG

IDL> FOR I=0,n-1 do help,*(ptr_values[i])
<PtrHeapVar1>   STRING    = 'p.mustermann'
<PtrHeapVar2>   FLOAT     = Array[10]
<PtrHeapVar3>   STRING    = 'time'
<PtrHeapVar4>   STRING    = 'time [UT]'
<PtrHeapVar5>   STRING    = 'seconds since 2000-01-01 00:00:00 UTC'
<PtrHeapVar6>   STRING    = 'NONE'
<PtrHeapVar7>   FLOAT     = Array[10]
<PtrHeapVar8>   STRING    = 'N2O'
<PtrHeapVar9>   STRING    = 'mixing ratio'
<PtrHeapVar10>  STRING    = 'ppb'
<PtrHeapVar11>  STRING    = 'NONE'
```

http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_source/idl_
html/dbase/download/struct2names_and_ptrs.tar.gz

This routine was written to use by add_tag, delete_tag, replace_tagvalue
and rename_tag
on each level of a structure with the following routine you are able to
build any structure defined from names and ptr_values.

http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_source/idl_
html/dbase/download/names_and_ptrs2struct.tar.gz

For further routines and licensing please look at
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml

Is this the solution of your problem?

Reimar

--
Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-1)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
http://www.fz-juelich.de/icg/icg1/
==========
a IDL library at ForschungsZentrum Jï¿½lich
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml

http://www.fz-juelich.de/zb/text/publikation/juel3786.html