
Subject: array dimensions

Posted by [francesco.spada](#) on Sat, 07 Apr 2001 11:16:40 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi to everybody,

I have an array

myarray fltarr[dimx,dimy,dimz]

How can I get the value of dimx, dimy and dimz?

Whith n_elements I have only the total number.

Thanks

Ciao

Francesco

Subject: Re: array dimensions

Posted by [davidf](#) on Mon, 09 Apr 2001 13:49:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt (craigmnet@cow.physics.wisc.edu) writes:

> Yes, it's called the degenerate case. REBIN and REFORM can
> essentially do this.

Must be a math thing. I don't find "degenerate" in the
index of any of my programming books. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: array dimensions

Posted by [Martin Schultz](#) on Mon, 09 Apr 2001 15:18:52 GMT

Paul van Delst wrote:

```
>
> Martin Schultz wrote:
>>
>> "Richard G. French" wrote:
>>>
>>> David Fanning wrote:
>>>> > dimensions = Size(myarray, /Dimensions)
>>>>
>>>> Cheers,
>>>
>>> David is the master of keywords - I've used SIZE all these years without
>>> knowing that there was a /DIMENSIONS keyword. I guess the only way to
>>> know about these things is to read through the documentation over and
>>> over again and take notes on potentially useful keywords and options.
>>>
>>> Which brings me to my question - does anyone out there have a favorite
>>> keyword on a routine that we mere mortals might not know about, but
>>> which
>>> might make our lives much easier? Any suggestions welcome, except for
>>> HISTOGRAM and PLOT!
>>>
>>> Dick French
>>
>> Not a favorite keyword, but a favorite tool: Get a decent system (if
>> you haven't one), install emacs with idlwave, run the routine-info
>> collection, and then you have all the keywords you never knew but
>> always dreamt about literally at your fingertips.
>
> Similar to typing "?" at the IDL prompt and getting similar info - with examples no less!
```

Not quite, because after "?" you still need to navigate around in several index lists etc. before you find the calling syntax. But, of course, the information itself is the same - that's where Carsten took it from ;-)

Martin

```
--
[[[
[[ Dr. Martin Schultz  Max-Planck-Institut fuer Meteorologie  [[
[[          Bundesstr. 55, 20146 Hamburg          [[
[[          phone: +49 40 41173-308          [[
[[          fax:  +49 40 41173-298          [[
[[ martin.schultz@dkrz.de          [[
[[[
[[[
```

Subject: Re: array dimensions
Posted by [davidf](#) on Mon, 09 Apr 2001 15:51:23 GMT
[View Forum Message](#) <> [Reply to Message](#)

James Kuyper (kuyper@wizard.net) writes:

> It would have occurred to me; part of my group's standard testing
> routine is to check all boundary cases, which would have focused my
> attention on n=1.

Yeah, but I put guys who follow standard testing routines
right up there with guys who always hit the right tennis
shot. I mean, gosh, where's the let-it-all-hang-out
passion? Where's the juice? Where's the hell-bent-for-leather,
smash-it-down-his-throat winner?

Cheers,

David

P.S. Let's just say I've **never** understood, either, why
God lets those other guys win all the time. :-(

--

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: array dimensions
Posted by [John-David T. Smith](#) on Mon, 09 Apr 2001 16:25:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst wrote:

>
> Martin Schultz wrote:
>>
>> "Richard G. French" wrote:
>>>
>>> David Fanning wrote:
>>>> > dimensions = Size(myarray, /Dimensions)
>>>>
>>>> Cheers,
>>>
>>> David is the master of keywords - I've used SIZE all these years without
>>> knowing that there was a /DIMENSIONS keyword. I guess the only way to

```

>>> know about these things is to read through the documentation over and
>>> over again and take notes on potentially useful keywords and options.
>>>
>>> Which brings me to my question - does anyone out there have a favorite
>>> keyword on a routine that we mere mortals might not know about, but
>>> which
>>> might make our lives much easier? Any suggestions welcome, except for
>>> HISTOGRAM and PLOT!
>>>
>>> Dick French
>>
>> Not a favorite keyword, but a favorite tool: Get a decent system (if
>> you haven't one), install emacs with idlwave, run the routine-info
>> collection, and then you have all the keywords you never knew but
>> always dreamt about literally at your fingertips.
>
> Similar to typing "?" at the IDL prompt and getting similar info - with examples no less!
> Gasp!

```

Not even *sort of* similar. Yes, it's the same information (Carsten simply process the pdf manuals to get the documentation). But the access is amazingly better. Example:

```
IDL> d=size(myvar,/
```

Hmm what was that keyword to size? I'll hit [Tab] to see. Up comes:

```

Possible completions are:
DIMENSIONS  FILE_LUN
L64         N_DIMENSIONS
N_ELEMENTS  STRUCTURE
TNAME       TYPE

```

Aha, DIMENSIONS, I think that's it. Why don't I right click on it to find out. Up pops in the help window with help on size() displayed, and queued up to:

```

DIMENSIONS
Set this keyword to return the dimensions
of Expression. If Expression is scalar, the
result is a scalar containing a 0. For arrays,
the result is an array containing the array
dimensions. ...

```

OK, dimensions it is, middle click on it:

```
IDL> d=size(myvar,/DIMENSIONS
```

Aren't there some other arguments to size()? I don't recall. I'll just show the calling syntax. [C-c ?]. Here comes:

Usage: Result = SIZE(Expression)

Keywords: DIMENSIONS FILE_LUN L64 N_DIMENSIONS N_ELEMENTS STRUCTURE
TNAME TYPE

Source: Builtin

Hmm. I guess not. But what is that STUCTURE keyword all about. I don't remember that one. Why not right click on it (it's blue, which means it exists in the help).

STRUCTURE

Set this keyword to return all available
information about Expression in a structure.

OK, I'll file that away for a rainy day. As you can see, you soon become intimately aware of all the keywords in all your favorite routines, and able to summon them into existence with a few keystrokes. One positive side effect of keyword completion is that I am no longer lazy when it comes to partial keywords (OK I'm still lazy, but IDLWAVE is not). No longer is my code littered with fragments like "a=size(var,/DIME)" and other non-sequiturs.

This is not to mention help with finding the correct routine in the first place. I haven't thrown away my regular on-line data-files, I just visit them only about once a month.

To see (and work through) even more examples, have a look at the tutorial:

<http://www.strw.LeidenUniv.nl/~dominik/Tools/idlwave/idlwave.html#SEC3>

JD

P.S. There are a couple of new features in the offing for an IDLWAVE release. If you've ever been tired of the wait for routine info to load, look for a nice improvement "real soon now". A nice new drag-to-print feature for instant debugging fun is also in the works. Stay tuned.

Subject: Re: array dimensions

Posted by [Paul van Delst](#) on Mon, 09 Apr 2001 16:49:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

JD Smith wrote:

>

> Paul van Delst wrote:

```

>>
>> Martin Schultz wrote:
>>>
>>> Not a favorite keyword, but a favorite tool: Get a decent system (if
>>> you haven't one), install emacs with idlwave, run the routine-info
>>> collection, and then you have all the keywords you never knew but
>>> always dreamt about literally at your fingertips.
>>
>> Similar to typing "?" at the IDL prompt and getting similar info - with examples no less!
>> Gasp!
>
> Not even *sort of* similar. Yes, it's the same information (Carsten
> simply process the pdf manuals to get the documentation). But the
> access is amazingly better. Example:
>
> IDL> d=size(myvar,/
>
> Hmm what was that keyword to size? I'll hit [Tab] to see. Up comes:

```

I type "size" <Enter> in the Online help window (after alt-tab-ing to the app, or moving mouse over app).

```

>
> Possible completions are:
> DIMENSIONS          FILE_LUN
> L64                  N_DIMENSIONS
> N_ELEMENTS          STRUCTURE
> TNAME                TYPE
>
> Aha, DIMENSIONS, I think that's it. Why don't I right click on it to
> find out. Up pops in the help window with help on size() displayed,

```

Help on size() is displayed for me and, if required <pageDown> gets me to "DIMENSIONS"

```

> and queued up to:
>
>   DIMENSIONS
>   Set this keyword to return the dimensions
>   of Expression. If Expression is scalar, the
>   result is a scalar containing a 0. For arrays,
>   the result is an array containing the array
>   dimensions. ...
>
> OK, dimensions it is, middle click on it:

```

don't take my hands off keyboard, but

```

> IDL> d=size(myvar,/DIMENSIONS

```

that's pretty neat (inserting it automagically like that).

- > Aren't there some other arguments to size()? I don't recall. I'll just
- > show the calling syntax. [C-c ?]. Here comes:
- >
- > Usage: Result = SIZE(Expression)
- > Keywords: DIMENSIONS FILE_LUN L64 N_DIMENSIONS N_ELEMENTS STRUCTURE
- > TNAME TYPE
- > Source: Builtin

I get that up front.

- >
- > Hmm. I guess not. But what is that STUCTURE keyword all about. I don't
- > remember that one. Why not right click on it (it's blue, which means it
- > exists in the help).

Once again: <pageDown> (if required)

- > STRUCTURE
- > Set this keyword to return all available
- > information about Expression in a structure.
- >
- > OK, I'll file that away for a rainy day.

Me too.

- > As you can see, you soon
- > become intimately aware of all the keywords in all your favorite
- > routines, and able to summon them into existence with a few keystrokes.

Me too.

- > One positive side effect of keyword completion is that I am no longer
- > lazy when it comes to partial keywords (OK I'm still lazy, but IDLWAVE
- > is not). No longer is my code littered with framents like
- > "a=size(var,/DIME)" and other non-sequiturs.

I never did that anyway (except for /INFO instead of /INFORMATIONAL since "info" is a common, understood contraction.) since I have an horrendous memory - but I concede that keyword completion is an extremely handy/good-for-eliminating-fat-finger-errors type of thing. When I first started using emacs, I found there was just too many ctrl-alt-esc-shift-meta-letter/number sequences to remember.

Diff'rnt strokes, I guess.

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: array dimensions
Posted by [Carsten Dominik](#) on Mon, 09 Apr 2001 17:57:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

>>>> > "JS" == JD Smith <jdsmith@astro.cornell.edu> writes:

JS> Not even *sort of* similar. Yes, it's the same information
JS> (Carsten simply process the pdf manuals to get the documentation).

I heavily object to "simply". This was one hell of a job, to
reverse-engineer thousands of pages of (not always consistent)
documentation.

:-)

- Carsten

Subject: Re: array dimensions
Posted by [John-David T. Smith](#) on Mon, 09 Apr 2001 18:09:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst wrote:

>
> JD Smith wrote:
>>
>> Paul van Delst wrote:
>>>
>>> Martin Schultz wrote:
>>>>
>>>> Not a favorite keyword, but a favorite tool: Get a decent system (if
>>>> you haven't one), install emacs with idlwave, run the routine-info
>>>> collection, and then you have all the keywords you never knew but
>>>> always dreamt about literally at your fingertips.
>>>
>>> Similar to typing "?" at the IDL prompt and getting similar info - with examples no less!
>>> Gasp!

OK, just to make the distinction crystal clear: With the RSI online help, you can search through the index, by typing "size" for example. This (sometimes) puts you on the routine in question. Hmm, in this case, it pulls up "SIZE Executive Command". Luckily "SIZE Function" is the next entry. Try "window" though. Hmm... window object, color model..., WINDOW system variable, etc. In any case, once you've (finally) targetted in on your routine of choice, you open up the help topic (e.g. for WINDOW Procedure) by double-clicking. Sometimes it opens right away. In this case (and many others), it wants more information. Do I really want "WINDOW", or perhaps I'm interested in "Window Systems". OK, "WINDOW" it is. Finally the routine documentation opens and I can scroll through manually looking for keywords.

IDLWAVE does all these operations for you instantaneously. If you're sitting on:

```
window,PIXMAP=
```

and invoke help, IDLWAVE *knows* you're not looking for the !D.WINDOW system variable, or random notes about window objects, or windowing functions, or window washing, etc. It knows you'd like some info on the "PIXMAP" keyword to the "WINDOW" procedure, and dials directly and instantaneously to:

PIXMAP

Set the PIXMAP keyword to specify that the window being created is actually an invisible portion of the display memory called a pixmap.

You get the info you need on the routine/procedure/keyword/system_variable at hand, and get on with your work. I invoke help this way so much now I don't even notice it really. It has become an integral part of my coding technique. As someone who spent many an hour doing it the RSI way with their help browser (type into index name, (possibly) scroll down nearby because it found the wrong thing, double-click, (possibly) dismiss another pop-up, scroll down looking for keywords, etc.), I can guarantee you the IDLWAVE way is *much* faster. It may not be for everyone, but it sure saves me lots of time. One consideration before you throw out the online manuals: IDLWAVE's version is text only, so those cool formatted mathematics and graphics don't make it through very well. This is not a problem 99.5% of the time. I keep RSI's browser handy for the other 0.5%.

I can get at help in many ways: through a keyword completion list, through the routine info, by C-S right clicking on anything (keyword, routine, system variable, object, obj_new invocation, etc.) in the

buffer *or* shell, etc.

And here's the best. Lest we not forget some of the best IDL routines are not supplied by RSI... if there is no system help topic for a routine you invoke help on, but its source code is available, that will be loaded up instead. If the source code has a standard documentation header, it will be visited for keywords (with a keyword section), and syntax too. E.g. the nasa procedure, FITS_WRITE:

```
IDL> fits_write, NANVALUE=
```

Invoke help... presto:

```
;NaNvalue: data value in DATA to be replaced with IEEE NaN in the output  
; file.  
...
```

So it seamlessly integrates help (and calling syntax, and keyword lists, and completion, and...) from all external libraries too, and even from your own routine collection! This is the reason I have started to use well formed documentation headers again (also because [C-c C-h] inserts one automatically for me, and [C-c Ret] puts the date, my name, etc. for a Modification Entry). Even if you don't have a good doc header, IDLWAVE pulls up the routine declaration.

Never be without routine/syntax/keyword/sysvar/object_method information again.

JD

Subject: Re: array dimensions

Posted by [m.hadfield](#) on Mon, 09 Apr 2001 21:07:53 GMT

[View Forum Message](#) <> [Reply to Message](#)

From: "JD Smith" <jdsmith@astro.cornell.edu>

> ...

> So it seamlessly integrates help (and calling syntax, and keyword lists,
> and completion, and...) from all external libraries too, and even from
> your own routine collection! This is the reason I have started to use
> well formed documentation headers again (also because [C-c C-h] inserts
> one automatically for me, and [C-c Ret] puts the date, my name, etc. for
> a Modification Entry). Even if you don't have a good doc header,
> IDLWAVE pulls up the routine declaration.

Damn, I'm running out of excuses not to use that thing !

Mark Hadfield
m.hadfield@niwa.cri.nz <http://katipo.niwa.cri.nz/~hadfield>
National Institute for Water and Atmospheric Research

--

Posted from clam.niwa.cri.nz [202.36.29.1]
via Mailgate.ORG Server - <http://www.Mailgate.ORG>

Subject: Re: array dimensions
Posted by [Richard French](#) on Tue, 10 Apr 2001 03:29:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst wrote:

```
>  
> IF ( n GT 1 ) THEN SMOOTH( array, n )
```

```
>  
> ?
```

That's fine if you want this to be on a single line, but there are plenty of times when SMOOTH(array, n) is in a plot command, or in part of a complex expression. I think the argument about rebin is exactly similar - of course, you can do the checking ahead of time, but who wants to add this check for every time you do dynamic smoothing? adds lots of unneeded hair to the code. Smoothing by 1 is not the same as dividing by zero. It is like convolving with a one-element array, which IDL is perfectly willing to do without complaining and dying.

Dick French

Subject: Re: array dimensions
Posted by [Martin Schultz](#) on Tue, 10 Apr 2001 10:37:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst wrote:

```
>  
>  
> I type "size" <Enter> in the Online help window (after alt-tab-ing to the app, or moving  
> mouse over app).
```

Try this with PLOT ;-)

IDL> plot
and here are your PLOT keywords.

well, yes. But how many "page downs". Wait until you write widget code. What did I read yesterday in David's book: there are over 60 keywords to `widget_info` alone!

Martin

```
--
[[ Dr. Martin Schultz  Max-Planck-Institut fuer Meteorologie  [[
[[ Bundesstr. 55, 20146 Hamburg  [[
[[ phone: +49 40 41173-308  [[
[[ fax: +49 40 41173-298  [[
[[ martin.schultz@dkrz.de  [[
--
```

Mark Hadfield wrote:

Page 12 of 15 ---- Generated from [comp.lang.idl-pvwave](#) archive

Just say you work on a Windows computer ;-)

Martin

PS: for Pavel - delete "Windows" insert "Mac"

--

```
[[ Dr. Martin Schultz  Max-Planck-Institut fuer Meteorologie  [[  
[[ Bundesstr. 55, 20146 Hamburg  [[  
[[ phone: +49 40 41173-308  [[  
[[ fax: +49 40 41173-298  [[  
[[ martin.schultz@dkrz.de  [[  
[[
```

Subject: Re: array dimensions

Posted by [Carsten Dominik](#) on Tue, 10 Apr 2001 14:06:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

>>>> > "MS" == Martin Schultz <martin.schultz@dkrz.de> writes:

MS> Mark Hadfield wrote:

>>

>> Damn, I'm running out of excuses not to use that thing !

>>

MS> Just say you work on a Windows computer ;-)

Thats no excuse. Only the shell is not working under windows, all the rest is.

- Carsten

MS> Martin

MS> PS: for Pavel - delete "Windows" insert "Mac"

MS> --

```
MS> [[ Dr. Martin Schultz Max-Planck-Institut fuer Meteorologie [[ [[  
MS> Bundesstr. 55, 20146 Hamburg [[ [[ phone: +49 40 41173-308 [[ [[  
MS> fax: +49 40 41173-298 [[ [[ martin.schultz@dkrz.de [[  
MS> [[
```

Subject: Re: array dimensions

Posted by [Paul van Delst](#) on Tue, 10 Apr 2001 14:08:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

"Richard G. French" wrote:

>

> Paul van Delst wrote:

>>

>> IF (n GT 1) THEN SMOOTH(array, n)

>>

>> ?

> That's fine if you want this to be on a single line, but there are

> plenty

> of times when SMOOTH(array, n) is in a plot command, or in part of a

> complex

> expression. I think the argument about rebin is exactly similar - of

> course,

> you can do the checking ahead of time, but who wants to add this check

> for every time you do dynamic smoothing? adds lots of unneeded hair to

> the code. Smoothing by 1 is not the same as dividing by zero. It is like

> convolving with a one-element array, which IDL is perfectly willing to

> do without complaining and dying.

Fair enough. Complaining and dying is a decidedly poor response. I guess it would be nice to be able to put a

```
IF ( n LE 1 ) THEN RETURN, array
```

or

```
IF ( n LE 1 ) THEN BEGIN
```

```
  MESSAGE, 'N < or = 1, exiting...'
```

```
  RETURN, array
```

```
ENDIF
```

as the first executable statement(s) in a smooth.pro, if it existed for users to meddle with. I just checked the IDL documentation and it doesn't state that n=1 is invalid. In fact, it is implicitly valid since it is an odd number. So, wouldn't the problem with SMOOTH when n=1 be a bug ... that RSI should fix?

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: array dimensions

Posted by [davidf](#) on Tue, 10 Apr 2001 14:21:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst (paul.vandelst@noaa.gov) writes:

- > So, wouldn't the problem with
- > SMOOTH when n=1 be a bug ... that RSI should fix?

I'm sure after all this sound and fury, someone has thought to give them a call. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155
