Subject: Re: How does REFORM work in PV-Wave
Posted by Mike Schienle on Tue, 30 Nov 1999 08:00:00 GMT
View Forum Message <> Reply to Message

In article <8214p7kcq1@news.wrc.xerox.com>, jeyadev@wrc.xerox.com wrote:

...
> I guess what I am missing is the *order* in which elements are stored.
> I think that PV-Wave stores 2d arrays in the 'row' format (first index
> varies fastest), but what about higher dimensional arrays?

You can probably find more than you wanted to know abot row and column order by visiting the IDL FAQ at <http://www.ivsoftware.com:8000/FAQ/>. Select the "Search FAQ" button. Enter the word "major" in the "Question" field and press the "Start Search" button. You'll be treated to a fairly detailed discussion on column- and row-major, as well as memory access into the arrays.

Folks, as long as we're on the PV-WAVE subject, I have an old FAQ for PV-WAVE that was done in 1995. It's at <http://www.ivsoftware.com/pvwave_faq.html>. If someone would like to take it over, drop me a line. If someone just wants to update it, I can put it into the same format as the IDL FAQ and serve it from this site.

--
Mike Schienle Interactive Visuals, Inc.
mgs@ivsoftware.com Remote Sensing and Image Processing
http://www.ivsoftware.com/ Analysis and Application Development

Subject: Re: How does REFORM work in PV-Wave
Posted by jeyadev on Wed, 01 Dec 1999 08:00:00 GMT
View Forum Message <> Reply to Message

In article <mgs-52612D.20571630111999@news.silcom.com>, Mike Schienle <mgs@ivsoftware.com> wrote:
>
> You can probably find more than you wanted to know abot row and column
> order by visiting the IDL FAQ at <http://www.ivsoftware.com:8000/FAQ/>.
> Select the "Search FAQ" button. Enter the word "major" in the "Question"
> field and press the "Start Search" button. You'll be treated to a fairly
> detailed discussion on column- and row-major, as well as memory access
> into the arrays.

Found it, at last, by listing all the questions, but I know all *that* stuff.

My question was what happens beyond 2 dimensions and how REFORM treats a 2d to 3d coversion. I will simplify my question in the hope that some kind soul will help me out.

Let us say that I have the data file

```
 1  13
 2  14
 3  15
 4  16
 5  17
 6  18
 7  19
 8  20
 9  21
10  22
11  23
12  24
```

and that the first column represents data for a variable that is defined on a 3 x 4 (i.e. 3 column and 4 rows) grid and the second column is for another variable on the same grid. Assume that the data is stored in the the array odat(2,12).

What is I want to do is the following: I want to create a 3 data array with two planes of 3 x 4 elements so that each plane contains the the data for one variable.

The REAL QUESTION: The command

data = reform(odat,2,3,4)

seems to do the job. For example

```
WAVE> a = data(0,*,*)
WAVE> info, a
A               INT       = Array(1, 3, 4)
WAVE> a = reform(a)
WAVE> info, a
A               INT       = Array(3, 4)
WAVE> print, a
      1       2       3
      4       5       6
      7       8       9
     10      11      12
```

which is exactly what I want. Now, what I would like to know is why the number of planes (2) had to be the *first* index in the reform statement.

thanks
--

Surendar Jeyadev jeyadev@wrc.xerox.com