

---

Subject: Case for XML (Was: convert very large string to numeric)

Posted by [mvukovic](#) on Wed, 27 Aug 2003 16:43:44 GMT

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

---

Paul van Delst <paul.vandelst@noaa.gov> wrote in message

news:<3F4B91F6.9F287A27@noaa.gov>...

> Mirko Vukovic wrote:

>>

>> Paul van Delst <paul.vandelst@noaa.gov> wrote in message

news:<3F4A7ADE.AF8396AD@noaa.gov>...

>>> Mirko Vukovic wrote:

>>>>

>>>> Hello,

>>>>

>>>> I have a large two column matrix stored as a string,

>>>

>>> Forgive my denseness, but what do you mean exactly when you say you "have a large two

>>> column matrix stored as a string"? By stored do you mean on disk as an ASCII file, or in a

>>> variable as an actual character variable?

>>>

>>> If the latter, my next question is: how did it get that way? (It's not a facetious

>>> question...I'm fishing for more details)

>>>

>>> paulv

>>

>> Hmmm. It seems that my exposition was lacking in crucial details.

>>

>> The data is coming from an E&M simulation program (Maxwell 2D,

>> student version). The really gory details are as follows:

>>

>> - From Maxwell I generate the text file with the data.

>> - With an editor, and insert some XML tags. The file now has a

>> snippet that looks as follows, and whose contents I need to get into

>> IDL

>>

>> <Data-Set>

>> 239843420958.0 23049823048.023984032

>> 3240.83240 0239483.2094

>> 20348.3204 20394803.24

>> .

>> .

>> .

>> 39458.7435 348324.497324

>> </Data-Set>

>>

>> - I use IDL's XML reader (properly customized via inheritance) to read

>> the data.

>

> O.k., so it's the XML read that sticks the data into one big string.  
>  
> Why not just read the ASCII datafile in one big block and skip the XML read? It'll be a  
> lot faster.  
>  
>> You may wonder why use XML. Well, It strated out as a challenge.  
>> But, after I did it for the first time, I was really impressed that I  
>> could add some intelligent information to my data files, and my file  
>> reader would be able to read them, or skip them, or whatever. So for  
>> now, I continue to use them.  
>  
> How about rather than <Data-Set> you add the number of lines in this data set? (That's  
> intelligent information too :o) Then your reader can read the number of lines, allocate  
> the required size array and read everything in at once. Using XML may be a little bit  
> easier (don't have to count the lines) but you're effectively reading the data twice -  
> once from file and once from string->variable.  
>  
> I doubt this will solve your problem because it seems too simple (my solution, I mean. Not  
> your problem.)  
>  
> paulv

You are absolutely correct. I could do it that way. I used to do it that way, but decided that it was time to try and learn something new. In this case XML. And the end result of this learning experience \_may\_ be that it is not terribly usefull for what I need right now.

The way I see it right now, the XML data file becomes a bit of a data base. It contains not just data, but comments, experimental parameters, info on experiment configuration, etc, all of which can be retrieved at will. Furthermore, it is \_extendable\_. I can add additional information to the file, and not worry that my reader will not be able to parse it. So in the end, the main advantage is \_EXTENDABILITY\_. I guess that where the X comes from :-)

So far I am rather pleased with it's (xml) performance. I just need to speed it up a bit, or upgrade from my 0.5GHz machine.

Mirko

PS. I looked at the routine that (I forget first name, Reiner?) Bauer suggested. I think I can easily adapt it to replace the use of PARSELINE. Thanks.

---

Subject: Re: CASE  
Posted by [David Fanning](#) on Fri, 04 Feb 2005 11:39:25 GMT

David Lopez Pons writes:

```
> I want to write this with CASE:
>
> CASE str OF
>   'a':      print,'joy!'
>   'b' OR 'c': print,'ouch!'
>   ELSE:     print, '??'
> ENDCASE
>
> obviously this syntax doesn't work
```

I would try something like this:

```
CASE 1 OF
  (str EQ 'a'): Print, 'joy'
  (str EQ 'b') OR (str EQ 'c'): Print, 'ouch!'
  ELSE: Print, '??'
ENDCASE
```

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

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

---

---

Subject: Re: CASE

Posted by [eddie haskell](#) on Fri, 04 Feb 2005 16:26:34 GMT

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

---

```
>> I want to write this with CASE:
>>
>> CASE str OF
>>   'a':      print,'joy!'
>>   'b' OR 'c': print,'ouch!'
>>   ELSE:     print, '??'
>> ENDCASE
>>
>> obviously this syntax doesn't work
```

Unless you have your heart really set on using the CASE you could use a SWITCH instead of a CASE.

Example:

```
SWITCH str OF
'a' : BEGIN
  print,'joy!'
  BREAK
END
'b' :
'c' : BEGIN
  print, 'ouch!'
  BREAK
END
ELSE : print, '??'
ENDSWITCH
```

Cheers,  
eddie

---