Subject: Re: convert very large string to numeric
Posted by Paul Van Delst[1] on Mon, 25 Aug 2003 21:08:46 GMT
View Forum Message <> Reply to Message

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

--
Paul van Delst
CIMSS @ NOAA/NCEP/EMC
Ph: (301)763-8000 x7748
Fax:(301)763-8545

Subject: Re: convert very large string to numeric
Posted by R.Bauer on Mon, 25 Aug 2003 21:31:04 GMT
View Forum Message <> Reply to Message

Mirko Vukovic wrote:

> Hello,
>
> I have a large two column matrix stored as a string, and I want to
> convert it to a two column float matrix.
>
> I've been using a rather convoluted method of flattening out the
> string matrix by replacing end-of-lines with spaces (having converted
> to byte array first), and then using ParseLine (from David F.'s web
> site) to get a vector of numbers:
>
>
> ByteBuffer = byte(self.CharBuffer)
> iSubs = where((ByteBuffer eq 9b) or $
> (ByteBuffer eq 10b) or $
> (ByteBuffer eq 13b), cSubs)
> if cSubs ne 0 then begin

> ByteBuffer[iSubs] = 32b
> endif
>
> ... etc

Oh

why did you operate with bytes?

Ok we do it in some cases too.

I would suggest to use bytes2strarr
 http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_source/idl _html/dbase/bytes2strarr_dbase.pro.html

to convert the bytes to a stringarray. This means it is not necessary to
convert 9b,10b,13b to 32b.

result is a string array.

From a string array it is easy to read by reads values from strings.

If you original have an ascii file I would suggest to use our
read_data_file.
It has two modes. default is table orientated with the /vstruct keyword you
can change it into column orientated.
 http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_source/idl
_html/dbase/read_data_file_dbase.pro.html

regards

Reimar

--
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
http://www.fz-juelich.de/icg/icg-i/
 ========== ======
a IDL library at ForschungsZentrum Juelich
 http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro. html

---

Subject: Re: convert very large string to numeric
Posted by mvukovic on Tue, 26 Aug 2003 16:42:07 GMT

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 comming 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.
- Now, inside this reader, the data is in a very large character
string (character buffer). The string contains the verbatim contents
of that particular part of the file. Thus it includes line-feeds,
carriage returns, spaces, tabs, numerals, everything:

239843420958.0 23049823048.023984032
3240.83240 0239483.2094

20348.3204 20394803.24
.
.
.
39458.7435 348324.497324

I have to convert this very long string to a 2*N matrix.

If you look at my original post, the way I do it is first ``flatten
the string'' by removing all line-feeds and carriage returns, and
replacing them with spaces (I do this by converting it to BYTE, doing
a WHERE and replacing. Now my string corresponds to a very loong line
of text. Before, it had line breaks.

239843420958.0 23049823048.023984032 3240.83240 0239483.2094
20348.3204 20394803.24 . . . 39458.7435 348324.497324

At this point, I need to pluck out individual groups of numbers (which
are separated by spaces), and convert them to floats or doubles, and
store them into a vector. I use PARSELINE.

Finally I REFORM the vector to desired dimensions. And that part
takes some time that I was hoping to shorten.

How much time? Oh, 3-5 sec per data set. So far, since yesterday I
have spent a total of about 2 minutes waiting for PARSELINE.
Composing the original post, reading the replies, and writing this,
took another 15min. :-)

Hope this explains my problem better. Thanks for all replies. I'm
off now to check Mr. Bauer's suggestions.

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.

Mirko

Subject: Re: convert very large string to numeric
Posted by Paul Van Delst[1] on Tue, 26 Aug 2003 16:59:34 GMT
View Forum Message <> Reply to Message

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 comming 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

--
Paul van Delst
CIMSS @ NOAA/NCEP/EMC
Ph: (301)763-8000 x7748
Fax:(301)763-8545

Subject: Re: convert very large string to numeric
Posted by Rick Towler on Wed, 27 Aug 2003 17:59:52 GMT
View Forum Message <> Reply to Message

> 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.

Since you already have a solution I would probably stick with it. But, if performance is that important, I would suggest either changing your XML file structure (say by adding a <row> </row> tag) or writing your own parser that doesn't rely on char data. The latter can be done easily (not with IDLffXMLSAX though). I posted some code on the newsgroup a while ago...

-Rick
