

Re: Data table text I/O package?

Source: <http://coding.derkeiler.com/Archive/Ada/comp.lang.ada/2005-06/msg00289.html>

- *From:* "Dmitry A. Kazakov" <mailbox@xxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Tue, 21 Jun 2005 14:44:26 +0200
-

On Tue, 21 Jun 2005 13:41:25 +0200, Georg Bauhaus wrote:

- > Dmitry A. Kazakov wrote:
- >
- >> See below. It is a table. It has bracketing: rows and columns.
- >
- > Back to step one: brackets in computer tables are not named,
- > a computer doesn't have accountants' abilities in pattern matching
- > when looking at rows and columns in a table. Again, I said XML is good
- > for parsing of data if you cannot tell in advance that the data stream
- > is totally free of errors.

No it is bad, because missing one bracket may lead to loss of the whole data set. As a medium XML is as awful as readable.

- > XML provides means to build robust data
- > streams in the absence of tight definitions and reliable procedures.

- > As for whitespace, read Stroustrup's article on defining operator
- > whitespace.

Delimiter /= whitespace.

- >> This form
- >> existed for centuries before XML. Who would print tables of logarithms in
- >> XML?
- >
- > You're missing the point: XML is **not** about rendering data.

Sorry, but the thread's subject reads "Data table text I/O package". Text = rendered data.

- > Logarithms are logarithms, not printed logarithms, this is a second
- > step. Data formats for exchange or storage on the one hand and
- > a print-out of some data on the other hand are two very different beasts,
- > with different purposes. Consider the MVC paradigm.

This is obviously wrong, clearly print-outs serve both data exchange and data storage when humans are involved.

Re: Data table text I/O package?

>>>If you care about robust data interchange in a "sloppy
>>>field", you employ standard tools to help you get the correct
>>>data.
>>
>> That is a different problem for which I would use a well-defined binary
>> format instead of fancy 3.15. What is the *accuracy* of this value, huh?
>
> It is totally unimportant what you or I would want, sorry.
> For a robust data interchange, absent comprehensive definitions
> and guarantees about data production, you need redundancy, period.
>
> The accuracy is well defined and most importantly,
> it is up to the application, yours and mine respectively.
> We both use the accuracy that is most appropriate, and I won't
> tell you not to use an internal type when it suits your application.
> I expect the same of you. If all I have to do is to store kilometers
> measuring straight lines inside the Netherlands in a relational database,
> I know the datatype I can use, no matter what you think is best
> in your application.

This is a wrong approach of course. Because the accuracy of the data is *not* defined by the internal type used. And in any case the internal type is irrelevant to the data format used. Note that binary format has nothing to do with any internal format.

> This has been discussed for years during the development of
> XML Schema. What do you care about my accuracy as long as
> I compute values from your data that are within application
> bounds? 3.15 is as accurate as can be, and independent of
> bits.

Is it 3.14998751 or 3.150000? Floating-point numbers are intervals. Transporting them you should either use explicit bounds: [3.1499, 3.1600] or accuracy: 3.15 +/-0.0001. "As accurate as can be" is nice, but what if the application is a gateway, which reads 3.15 as accurate as 4 bytes float is and then sends it away? Two other applications communicating through it and using long long float will be quite perplexed...

>> But, lack of readability is not in the ugly </> brackets. Tabulated data
>> are readable because they are tabulated.
>
> This is the *View* in MVC, XML is about *data*. So there is no point in
> talking about final looks, it is important to know how data will have
> to be seen. For example, can you debug datastreams using the simplest
> tools? Think of a log file of a concurrent application, processing data
> from several heterogenous input sources on the net.

Really? A normal log file of our data acquisition and control system (3-4 nodes, 500-1000 channels each) is about 10-100 MB. A trace file of the same system is typically about 10-100GB. The first is a highly dense binary

Re: Data table text I/O package?

Re: Data table text I/O package?

format. The second is dense ASCII. Do you know any editor capable to load 10GB? In UltraEdit you need about 10 minutes to wait, before it becomes ready to do anything. Now, you propose me to convert all that into XML? How much is SCSI terabyte now? But more importantly each extra byte of rubbish you write is multiplied by the number of channels and their frequencies, that costs system performance.

>> That is: the names, the types and
>> units are *factored* out to the table header, which allows the reader to
>> concentrate on the *values*. Thus a table looks as:
>>
>> Distance [km] Temperature [°C] ...
>> 3.15 29.0 ...
>> 2.10 14.4 ...
>>
>> This is readable.
>
> This is irrelevant in data exchange. This is print.
>
>> To make difference more visible, consider bitmaps stored XML format. Would
>> you be able to recognize a person's face in it?
>
> You do know about NOTATION?
> I think it is very hard to find someone suggesting that we should recode
> bitmap graphics formats as pixel tags.

So an image is not print whereas a table is?

--

Regards,
Dmitry A. Kazakov
<http://www.dmitry-kazakov.de>

-
- *Follow-Ups:*
 - ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus

 - *References:*
 - ◆ **Data table text I/O package?**
◇ From: Jacob Sparre Andersen
 - ◆ **Re: Data table text I/O package?**
◇ From: Preben Randhol
 - ◆ **Re: Data table text I/O package?**
◇ From: Jacob Sparre Andersen
 - ◆ **Re: Data table text I/O package?**
◇ From: Preben Randhol
 - ◆ **Re: Data table text I/O package?**
◇ From: Jacob Sparre Andersen

Re: Data table text I/O package?

- ◆ **Re: Data table text I/O package?**
◇ From: Randy Brukardt
- ◆ **Re: Data table text I/O package?**
◇ From: Jacob Sparre Andersen
- ◆ **Re: Data table text I/O package?**
◇ From: Marius Amado Alves
- ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus
- ◆ **Re: Data table text I/O package?**
◇ From: Dmitry A. Kazakov
- ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus
- ◆ **Re: Data table text I/O package?**
◇ From: Dmitry A. Kazakov
- ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus
- ◆ **Re: Data table text I/O package?**
◇ From: Dmitry A. Kazakov
- ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus
- ◆ **Re: Data table text I/O package?**
◇ From: Dmitry A. Kazakov
- ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus
- ◆ **Re: Data table text I/O package?**
◇ From: Dmitry A. Kazakov
- ◆ **Re: Data table text I/O package?**
◇ From: Georg Bauhaus

- Prev by Date: **Re: Data table text I/O package?**
- Next by Date: **Re: ADA vs Java**
- Previous by thread: **Re: Data table text I/O package?**
- Next by thread: **Re: Data table text I/O package?**
- Index(es):
 - ◆ **Date**
 - ◆ **Thread**