Subject: Re: strange behaviour of bytscl by large arrays
Posted by Kenneth P. Bowman on Wed, 25 Apr 2012 21:11:43 GMT
View Forum Message <> Reply to Message

In article < 29512254.739.1335282014771.JavaMail.geo-discussion-forums@yn y11 >,
Chris Torrence <gorthmog@gmail.com> wrote:

```
>  Okay, alx has convinced me to not change anything. Try the following:
>
>  IDL> print, 16777216 + findgen(10), format='(f25.0)'
>                 16777216.
>                 16777216.
>                 16777218.
>                 16777220.
>                 16777220.
>                 16777220.
>                 16777222.
>                 16777224.
>                 16777224.
>                 16777224.
>
>  So even if you did the computation using long64's, as soon as you convert
>  them back to floats, you are going to get "jumps" in the findgen because of
>  the loss of precision. I suppose you could argue that this might be better
>  than having the findgen get "stuck" on the number 16777216, but I think the
>  speed of findgen is more important.
>
>  Thanks.
>
>  -Chris
>  Exelis VIS
```

Since this turns out to be a floating-point precision issue, does DINDGEN use a long64 counter?

And more importantly, could this possibly be documented in the manuals for the sake of future generations?

I know it is not the IDL way to document implementation details, but sometimes they are important when trying to understand how things work or why they don't.

Ken Bowman