Subject: Re: Problems with double precision in IDL
Posted by ryan on Wed, 16 Jun 1993 17:57:30 GMT
View Forum Message <> Reply to Message

When you convert a number from single precision to double
precision, you don't gain any precision better than the rounding
from the initial assignment. Make the initial assignment to a double
rather than converting. For example

z = .32d

Ryan

|-- ----|
| Ryan McLean ryan@physics.berkeley.edu |
| 450 LeConte Hall |
| UC Berkeley (510) 643-9690 |
| Berkeley, CA 94720 |
|-- ----|

Subject: Re: Problems with double precision in IDL
Posted by zawodny on Fri, 18 Jun 1993 16:41:06 GMT
View Forum Message <> Reply to Message

In article <16JUN199312122750@stars.gsfc.nasa.gov> isaacman@stars.gsfc.nasa.gov (Subvert the Dominant Paradigm! (301) 513-7769) writes:
> We have a potentially VERY serious problem with the COBE data analysis
> because of the way IDL seems to be (mis?)handling double precision
> numbers. Here is an example of how IDL treats floating point numbers
> when converting them to double precision. The operations were performed
> on a DECstation.
>
> z=.32
> print,f2,double(z) ; If a variable is declared DOUBLE this is what happens.
> 0.319999992847
> print,f2,.32d ; If the "d" notation is used instead it's accurate.
> 0.320000000000
> print,double(z)-.32D
> -7.15255737e-09
> print,[double(z)-.32D]/.32D
> -2.23517418e-08

Stuff deleted

>

>
> Thanks,
>
> Rich Isaacman
>
> COBE Project
> NASA/Goddard Space Flight Center

Isn't this apparent error at about the magnitude of the "fluctuations" in the residual temperature maps from COBE that are being used to "prove" some aspects of Big Bang theories? Gee, I hope that all the kudos and PR that has been given to COBE are not for nought because of a programming error. I was excited by those results myself.

Obviously a prior post is correct, numbers are not stored in computers as ASCII representations. You cannot gain precision in a previously declared varaible by simply declaring it to be double precision later on. This is true of a program written in any language and is not a "problem" with IDL.

--

Joseph M. Zawodny (KO4LW) NASA Langley Research Center
Internet: zawodny@arbd0.larc.nasa.gov MS-475, Hampton VA, 23681-0001
Packet: ko4lw@wb0tax.va.usa

Subject: Re: Problems with double precision in IDL
Posted by thompson on Fri, 18 Jun 1993 19:04:05 GMT
View Forum Message <> Reply to Message

In article <16JUN199312122750@stars.gsfc.nasa.gov> isaacman@stars.gsfc.nasa.gov (Subvert the Dominant Paradigm! (301) 513-7769) writes:
> We have a potentially VERY serious problem with the COBE data analysis
> because of the way IDL seems to be (mis?)handling double precision
> numbers. Here is an example of how IDL treats floating point numbers
> when converting them to double precision. The operations were performed
> on a DECstation.
>
> z=.32
> print,f2,double(z) ; If a variable is declared DOUBLE this is what happens.
> 0.319999992847
> print,f2,.32d ; If the "d" notation is used instead it's accurate.
> 0.320000000000
> print,double(z)-.32D
> -7.15255737e-09
> print,[double(z)-.32D]/.32D
> -2.23517418e-08

Stuff deleted

Here's a counter example from FORTRAN. I ran the following program on my Sun workstation

```
      program main
c
      real*4 a
      real*8 b
c
      a = 0.32
      b = a
      write (*,*) a,b
c
      end
```

and got the following results

```
   0.320000   0.31999999284744
```

This is the closest FORTRAN equivalent that I can think of to the IDL commands

```
z=.32
print,double(z)
```

that I can think of.

As has been mentioned before, this sort of behavior is endemic to all programming languages, and must be kept in mind when writing programs involving double precision variables. For example, I've seen FORTRAN programs that had statements in it like

```
DOUBLE PRECISION VAR1
VAR1 = 7./9.
```

which will introduce the same sorts of problems. In fact, this was in a commercial software package!

Bill Thompson

Subject: Re: Problems with double precision in IDL
Posted by isaacman on Sat, 19 Jun 1993 01:34:00 GMT
View Forum Message <> Reply to Message

In article <1vsr72INNgtf@rave.larc.nasa.gov>, zawodny@arbd0.larc.nasa.gov (Dr. Joseph M Zawodny) writes...
> In article <16JUN199312122750@stars.gsfc.nasa.gov> isaacman@stars.gsfc.nasa.gov (Subvert the Dominant Paradigm! (301) 513-7769) writes:

>> We have a potentially VERY serious problem with the COBE data analysis
>> because of the way IDL seems to be (mis?)handling double precision
>> numbers.
>
> Isn't this apparent error at about the magnitude of the "fluctuations" in the
> residual temperature maps from COBE that are being used to "prove" some aspects
> of Big Bang theories? Gee, I hope that all the kudos and PR that has been
> given to COBE are not for nought because of a programming error. I was excited
> by those results myself.
>

Not to worry. Those cosmic background fluctuations were all derived and
checked with a bunch of different double-precision Fortran programs.

Rich Isaacman
