Subject: Re: Unknown #INFO
Posted by davidf on Wed, 11 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Eric J. Korpela (korpela@islay.ssl.berkeley.edu) writes:

> David Fanning <davidf@dfanning.com> wrote:
>>
>> That is the IDL notation for "infinite value".
>>
>> IDL> Print, !Values.F_Infinity
>> 1.#INF0
>>
>
> Whatever happened to 'Inf' and '-Inf'? Is this new notation
> with the latest IDL version? If so, what's the new notation for
> 'NaN'?

IDL> Help, !Values, /Structure
** Structure !VALUES, 4 tags, length=24:
  F_INFINITY FLOAT 1.#INF0
  F_NAN FLOAT 1.#QNAN
  D_INFINITY DOUBLE 1.#INF000
  D_NAN DOUBLE 1.#QNAN00

IDL> Print, !Version
{ x86 Win32 Windows 5.1.1 Jul 20 1998}

I don't know. This may be a Windows thing. Looks a little
Bill Gatesh, doesn't it. :-)

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Unknown #INFO
Posted by korpela on Wed, 11 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.10b38b6d3b35c1d2989719@news.frii.com>,
David Fanning <davidf@dfanning.com> wrote:

>
> That is the IDL notation for "infinite value".
>
>   IDL> Print, !Values.F_Infinity
>   1.#INF0
>

Whatever happened to 'Inf' and '-Inf'?   Is this new notation
with the latest IDL version? If so, what's the new notation for
'NaN'?

Eric

--
Eric Korpela | An object at rest can never be
korpela@ssl.berkeley.edu | stopped.
Click for home page.

Subject: Re: Unknown #INFO
Posted by davidf on Wed, 11 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Luis Oliveira (eu_luis@hotmail.com) writes:

> I have received this output from PRINT, when I passed it an array:
>
> prf->tprfdata::get(w='velocitycomponents')
>   1.#INF0   1.#INF0   1.#INF0   1.#INF0   1.#INF0
>
> Please, if you have steped already in this "1.#INFO" or "-1#INFO" thing let
> me know,

That is the IDL notation for "infinite value".

   IDL> Print, !Values.F_Infinity
      1.#INF0

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Note: A copy of this article was e-mailed to the original poster.

Subject: Re: Unknown #INFO
Posted by mgs on Thu, 12 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <72fd75isf1@hammer.msfc.nasa.gov>, mallors@msfc.nasa.gov
(Robert S. Mallozzi) wrote:

> In article <MPG.10b49db91db8b3ae98971b@news.frii.com>,
> davidf@dfanning.com (David Fanning) writes:
>>
>> Hang on there. Here is what I get in IDL 5.2 beta 2:
>>
>> IDL> Print, !Version
>> { x86 Win32 Windows 5.2 Beta2 Sep 14 1998}
>>
>> IDL> Help, !Values, /Structure
>> ** Structure !VALUES, 4 tags, length=24:
>> F_INFINITY FLOAT Inf
>> F_NAN FLOAT NaN
>> D_INFINITY DOUBLE Infinity
>> D_NAN DOUBLE NaN
>>
>> IDL> Print, !Values
>> { Inf NaN Infinity NaN}
>>
>> IDL> Print, Float(['Inf', 'NaN', 'Infinity', 'NaN'])
>> Inf NaN Inf NaN
>>
>
> FYI,
>
> Identical results on
> { x86 linux unix 5.2 Beta2 Sep 14 1998}

And Identical results on
IDL> print,!version
{ PowerMac MacOS MacOS 5.2 Beta2 Sep 22 1998}

--
Mike Schienle Interactive Visuals, LLC
mgs@ivsoftware.com http://www.ivsoftware.com/

Subject: Re: Unknown #INFO
Posted by mallors on Thu, 12 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.10b49db91db8b3ae98971b@news.frii.com>,
davidf@dfanning.com (David Fanning) writes:

```
>
> Hang on there. Here is what I get in IDL 5.2 beta 2:
>
> IDL> Print, !Version
> { x86 Win32 Windows 5.2 Beta2 Sep 14 1998}
>
> IDL> Help, !Values, /Structure
> ** Structure !VALUES, 4 tags, length=24:
>    F_INFINITY     FLOAT            Inf
>    F_NAN          FLOAT            NaN
>    D_INFINITY     DOUBLE           Infinity
>    D_NAN          DOUBLE           NaN
>
> IDL> Print, !Values
> {        Inf        NaN       Infinity           NaN}
>
> IDL> Print, Float( [ 'Inf', 'NaN',  'Infinity', 'NaN' ] )
>          Inf        NaN        Inf        NaN
>
```

FYI,

Identical results on
{ x86 linux unix 5.2 Beta2 Sep 14 1998}

--
~~~~~~ ~~~~~~~~~~~~
Robert S. Mallozzi 256-544-0887
Mail Code ES 84
Work: http://www.batse.msfc.nasa.gov/ Marshall Space Flight Center
Play: http://cspar.uah.edu/~mallozzir/ Huntsville, AL 35812
~~~~~~ ~~~~~~~~~~~~

Subject: Re: Unknown #INFO
Posted by Martin Schultz on Thu, 12 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Stein Vidar Hagfors Haugan wrote:

> In article <72d8sr$9at$1@agate.berkeley.edu>
> korpela@albert.ssl.berkeley.edu (Eric J. Korpela) writes:
>
> [...]

Oh yeah, I love it!

```
IDL> print,!version
{ mipseb IRIX unix 5.1.1 Jul 20 1998}
IDL> print,!values
{         inf nan0x2000000             inf   nan0x70000000}
```

> IDL> print,float(['INF','Inf','NaN','nanq'])

```
IDL> print,float(['INF','Inf','NaN','nanq'])
% Type conversion error: Unable to convert given STRING to Float.
% Detected at:  $MAIN$
% Type conversion error: Unable to convert given STRING to Float.
% Detected at:  $MAIN$
% Type conversion error: Unable to convert given STRING to Float.
% Detected at:  $MAIN$
% Type conversion error: Unable to convert given STRING to Float.
% Detected at:  $MAIN$
      0.00000      0.00000      0.00000      0.00000
```

"All [systems] are equal, but some are more equal than others."
(G.Orwell)

BTW: I would say that it is alright for IDL to pick up the sign in
-float('Inf'). That's what people write below integrals all the time ;-)
So at least in theory you could test
for -Inf and replace it by the smallest (largest negative) number that
your system represents. But, I realize that
symbolic algebra seems rather mathematica's realm, and an average IDLer
would rather only have to test for
value eq !values.Inf instead of abs(value) ...

So long,
Martin.

PS: Hope they fixed this inconsistency across all platforms in 5.2

--- -------
Dr. Martin Schultz
Department for Engineering&Applied Sciences, Harvard University
109 Pierce Hall, 29 Oxford St., Cambridge, MA-02138, USA

phone: (617)-496-8318
fax : (617)-495-4551

e-mail: mgs@io.harvard.edu
Internet-homepage: http://www-as.harvard.edu/people/staff/mgs/
---------- -------

Subject: Re: Unknown #INFO
Posted by davidf on Thu, 12 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Stein Vidar Hagfors Haugan (steinhh@ulrik.uio.no) writes:

>> David Fanning <davidf@dfanning.com> wrote:
>>>
>>> IDL> Help, !Values, /Structure
>>> ** Structure !VALUES, 4 tags, length=24:
>>> F_INFINITY FLOAT 1.#INF0
>>> F_NAN FLOAT 1.#QNAN
>>> D_INFINITY DOUBLE 1.#INF000
>>> D_NAN DOUBLE 1.#QNAN00
>>>

>> My sun gives this:
>> IDL> print,!version
>> { sparc sunos unix 5.0 Apr 28 1997}
>> IDL> print,!values
>> { Inf NaN Infinity NaN}
>> IDL> print,float('Inf')
>> Inf
>> IDL> print,float('1.#INF0')
>> 1.00000
>
> I guess you're right on the spot with the assumption that the
> compiler (or rather, the c library) makes the difference:
>
> IDL> print,!version
> { alpha OSF unix 5.1.1 Jul 20 1998}
> IDL> print,!values
> { INF NaNQ INF NaNQ}
> IDL> print,float(['INF','Inf','NaN','nanq'])
> % Type conversion error: Unable to convert given STRING to Float.
> % Detected at: $MAIN$
> INF INF 0.00000 NaNQ

Hang on there. Here is what I get in IDL 5.2 beta 2:

```
IDL> Print, !Version
{ x86 Win32 Windows 5.2 Beta2 Sep 14 1998}

IDL> Help, !Values, /Structure
** Structure !VALUES, 4 tags, length=24:
   F_INFINITY      FLOAT             Inf
   F_NAN           FLOAT             NaN
   D_INFINITY      DOUBLE            Infinity
   D_NAN           DOUBLE            NaN

IDL> Print, !Values
{         Inf         NaN        Infinity            NaN}

IDL> Print, Float( [ 'Inf', 'NaN',  'Infinity', 'NaN' ] )
         Inf         NaN         Inf         NaN
```

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Unknown #INFO
Posted by steinhh on Thu, 12 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <72d8sr$9at$1@agate.berkeley.edu>
korpela@albert.ssl.berkeley.edu (Eric J. Korpela) writes:

> In article <MPG.10b3c2392717b0898971a@news.frii.com>,
> David Fanning <davidf@dfanning.com> wrote:
>>
>> IDL> Help, !Values, /Structure
>> ** Structure !VALUES, 4 tags, length=24:
>>   F_INFINITY   FLOAT   1.#INF0
>>   F_NAN   FLOAT   1.#QNAN
>>   D_INFINITY   DOUBLE   1.#INF000
>>   D_NAN   DOUBLE   1.#QNAN00
>>
>> I don't know. This may be a Windows thing. Looks a little
>> Bill Gatesh, doesn't it. :-)

>
> Yes it does look windowish. It probably has something to do
> with the compiler they used to make their windows version. What
> concerns me is how this affects reading and writing of text files.
> In the past I have relied on IDL correctly parsing 'Inf' and 'NaN'
> in input files. You can see the obvious portability problem.
>
> I'm not sure if IEEE specifies the text representation of 'Inf'
> and 'NaN' or if a program is required to be able to correctly parse
> them on input.
>
> Out of curiousity, could you try the following?
>
> print,float('Inf')
> print,float('1.#INF0')
>
> My sun gives this:
> IDL> print,!version
> { sparc sunos unix 5.0 Apr 28 1997}
> IDL> print,!values
> { Inf NaN Infinity NaN}
> IDL> print,float('Inf')
> Inf
> IDL> print,float('1.#INF0')
> 1.00000

I guess you're right on the spot with the assumption that the
compiler (or rather, the c library) makes the difference:

```
IDL> print,!version
{ alpha OSF unix 5.1.1 Jul 20 1998}
IDL> print,!values
{         INF        NaNQ            INF           NaNQ}
IDL> print,float(['INF','Inf','NaN','nanq'])
% Type conversion error: Unable to convert given STRING to Float.
% Detected at:  $MAIN$
         INF         INF     0.00000        NaNQ
```

So apparently, case doesn't matter, but the particular *spelling*
convention does make a difference, and that's even between two
unix flavors! What a bummer.. So much for the days when ascii
files were portable.. :-)

One other caveat:

```
IDL> print,float(['-Inf','Inf']),-float('Inf')
         INF         INF
        -INF
```

So the sign isn't picked up. But this again may be platform
specific.... then again, I never expected it to be possible
to do float('NaNQ') in the first place :-)

Regards,

Stein Vidar

Subject: Re: Unknown #INFO
Posted by korpela on Thu, 12 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.10b3c2392717b0898971a@news.frii.com>,
David Fanning <davidf@dfanning.com> wrote:
>
> IDL> Help, !Values, /Structure
> ** Structure !VALUES, 4 tags, length=24:
>   F_INFINITY   FLOAT   1.#INF0
>   F_NAN   FLOAT   1.#QNAN
>   D_INFINITY   DOUBLE   1.#INF000
>   D_NAN   DOUBLE   1.#QNAN00
>
> I don't know. This may be a Windows thing. Looks a little
> Bill Gatesh, doesn't it. :-)

Yes it does look windowish. It probably has something to do
with the compiler they used to make their windows version. What
concerns me is how this affects reading and writing of text files.
In the past I have relied on IDL correctly parsing 'Inf' and 'NaN'
in input files. You can see the obvious portability problem.

I'm not sure if IEEE specifies the text representation of 'Inf'
and 'NaN' or if a program is required to be able to correctly parse
them on input.

Out of curiousity, could you try the following?

print,float('Inf')
print,float('1.#INF0')

(I'm guessing that this will give the same result as read on the same strings.)

My sun gives this:
IDL> print,!version
{ sparc sunos unix 5.0 Apr 28 1997}
IDL> print,!values

{ Inf NaN Infinity NaN}
IDL> print,float('Inf')
Inf
IDL> print,float('1.#INF0')
1.00000

Thanks,

Eric

--
Eric Korpela | An object at rest can never be
korpela@ssl.berkeley.edu | stopped.
Click for home page.

Subject: Re: Unknown #INFO
Posted by davidf on Fri, 13 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

From one of my Aussie friends:

Using v5.2 Beta on OpenVMS I get :-

** Structure !VALUES, 4 tags, length=24:
F_INFINITY FLOAT Inf
F_NAN FLOAT NaN
D_INFINITY DOUBLE Infinity
D_NAN DOUBLE NaN

Looks like a nice pattern shaping up here. :-)

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: http://www.dfanning.com/
