

Re: Inconsistent negative numerical values in SQL2K database table

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.server/2005-02/1240.html>

From: Steve Kass (*skass_at_drew.edu*)

Date: 02/04/05

Date: Fri, 04 Feb 2005 01:25:19 -0500

Earl Newcomer wrote:

>Steve,

>

>Well done. That was exactly the issue. Thank you very much. I tested the
>fix of I*value and it works as you mentioned. How does that fix the data do
>you know?

>

>

As far as I can tell, so long as SQL Server has to do an arithmetic problem, it will create a new and proper decimal value and overwrite the existing one. Fortunately, it appears that the decimal arithmetic algorithms were not highly tuned to do clever things with the existing sign byte/bit. Only very simple updates, like SET D = -D might operate directly on the existing sign information.

SK

>To answer your question about copying between SQL Server instances I don't
>think this condition is caused by copying between SQL Servers. I do know
>that once a column has an incorrect sign bit in a SQL Server table that it
>will copy between SQL Server tables carrying with it the incorrect sign bit.

>

>In our case, I suspect the developers have used different ODBC drivers over
>time and some values were copied correctly while other values suffer from the
>sign bit problem. Since the cube is partitioned some of the partitions are
>built with correctly stored data and AS reports those values correctly while
>other partitions are built with incorrectly stored data and AS reports those
>incorrectly.

>

>We will add another DTS task to check the sign bit from now on.

>

>Thanks again,

>Earl Newcomer

>
> "Steve Kass" wrote:
>
>
>
>> Earl,
>>
>> There are probably some values for which the sign byte of the
>> decimal value is something other than 01 or 00. See the explanation
>> here:
>>
>> <http://groups.google.co.uk/groups?q=218AE3EB-60D7-4E5E-A2F1>
>>
>> You should be able to find the values with the funky sign and fix them
>> as shown in the above thread like this:
>>
>> update T set
>> decVal = 1*decVal
>> where [if a large table, identify just the ones that have bad signs if
>> possible]
>>
>> or something similar if the values should be negative. A handful of
>> data providers seem to cause this problem, but I don't recall seeing it
>> between SQL Servers before. How did you move the data from
>> one to the other?
>>
>> Steve Kass
>> Drew University
>>
>>
>>
>> Earl Newcomer wrote:
>>
>>
>>
>>> I have a case where the sum of a column containing positive, zero and
>>> negative values is correct in Query Analyzer but the values in the column do
>>> not contain a '-' (minus sign). The rows sort as expected but they do not
>>> have a minus sign. I can update a positive row value by multiplying it by -1
>>> and that value shows with a minus sign. The newly updated row sorts with the
>>> other negative values but it is the only row with a minus sign.
>>>
>>> This "condition" came to light when a cube built on the fact table (where
>>> this is occurring) was not summing correctly. The cube sees all values as
>>> positive as if an absolute function was being used. When the one positive
>>> row was updated to a negative value as described above, the cube recognized
>>> the negative value and the measure changed accordingly. What is it in the
>>> SQL Server table or in the cube interpretation of the table values that is
>>> causing this? Is there a workaround?
>>>
>>> Another twist. The table, pmix_fact1, with which we see this behavior was

microsoft.public.sqlserver.server: Re: Inconsistent negative numerical values in SQL2K database table

>>>

>>>

>>>