

## Re: Migration of Teradata and DB2 Timestamp fields.

**Source:**

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.datawarehouse/2004-05/0074.html>

---

**From:** Jacco Schalkwijk (*NOSPAMjaccos\_at\_eurostop.co.uk*)

**Date:** 05/19/04

Date: Wed, 19 May 2004 15:25:41 +0100

There is no datatype in SQL Server that has a precision of 1 millisecond. I can suggest 3 different workarounds:

- Store the timestamp in a CHAR(23) column in format yyyy-mm-ddThh:mi:ss.nnn. You CONVERT this to and from datetime with style 126. Note that the precision will go back to 3.33 milliseconds when you convert to datetime. This uses 23 bytes of storage.
- Store the timestamp as a UNIX timestamp (milliseconds since 1970-1-1) in a BIGINT column. This uses 8 bytes of storage.
- Store the timestamp in 2 columns, one is a SMALLDATETIME (precision 1 minute), the second one is a INT, to store the number of milliseconds. This uses 8 bytes of storage. (You can also use a SMALLINT if you want to save 2 bytes, but then you have to handle negative milliseconds.

Whichever is the most convenient of course depends on the calculations you are going to do with it.

--

Jacco Schalkwijk  
SQL Server MVP

"Arumugam" <arumugam@optusnet.com.au> wrote in message  
news:de3401c43bd6\$5e60f870\$a501280a@phx.gbl...

> I am trying to migrate timestamp fields in Teradata and  
> DB2 to SQL server. The problem is that Teradata and DB2  
> store timestamps to an accuracy of 1 microsecond, while  
> SQL Server stores it to the nearest 3.33 milliseconds  
> (the datetime datatype). The original accuracy needs to  
> be preserved as the columns are part of the primary key.  
>

> Is there an equivalent datatype in SQL Server or a quick  
> workaround for this problem? Date / Time functions need  
> to be performed on these columns.