

# Configuring Microsoft SQL Server 2008 and 2012 to use Virtual Clocks from Time Machine®

The following guide is intended as a basic outline for running Microsoft SQL Server 2008 and 2012 under a specified user account for performing date & time based software tests, time-lapsed simulations, or training new application users how to handle the date-triggered events within applications that are tied to a SQL Server database backend.

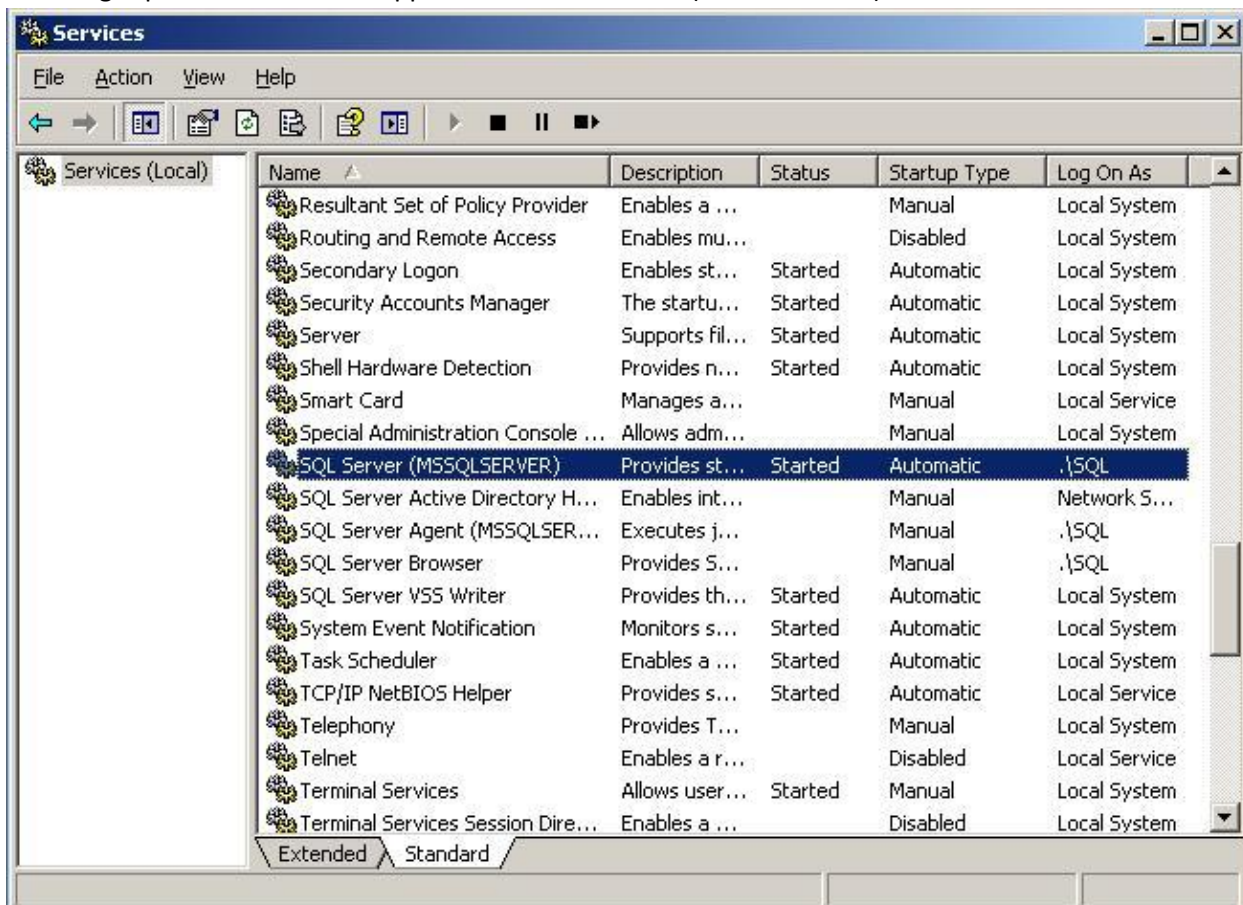
There are many applications that rely on Microsoft SQL Server to provide a backend data store for application data. In order to run simulations or test these applications with future or past dates, time zone differences or even simple leap year events it is necessary to provide Time Machine virtual clocks to the applications and the SQL server. If applications are running in user space you will need to provide a virtual clock for the logged in user's account as well as the account for the SQL services.

This paper shows how to setup and/or verify the SQL services are running with a 'username' defined by a system administrator. Additionally, the procedures in this paper can be followed as a rough guide for configuring other Windows services that may be part of the application software stack.

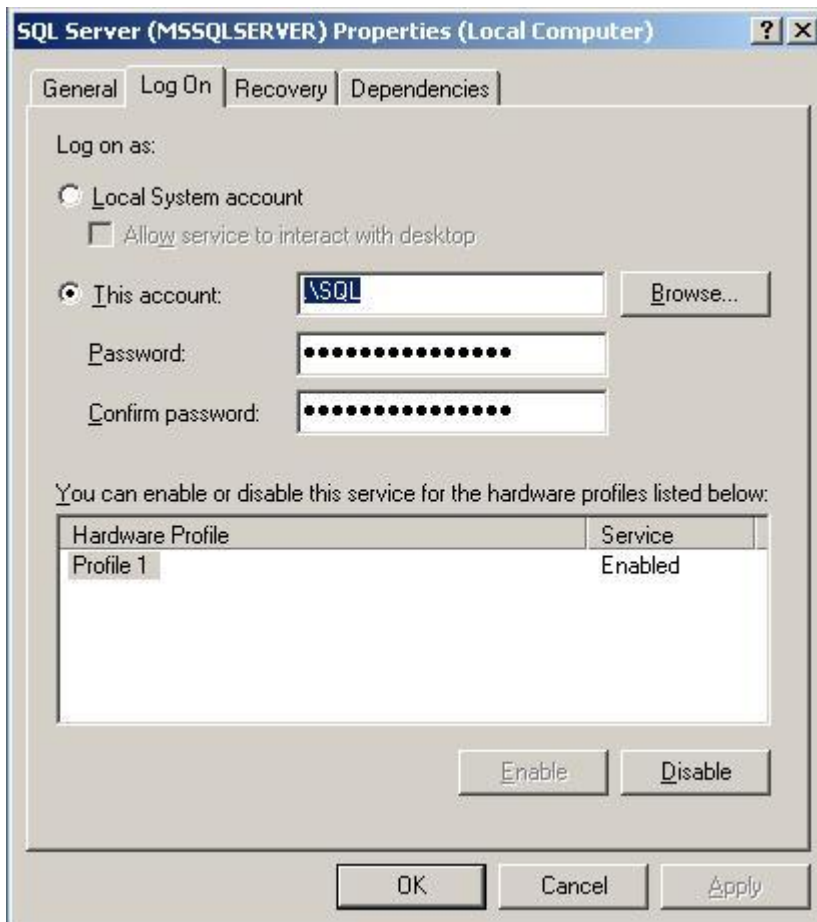
**System Details: The development & deployment for this documentation was performed on the following:**

Operating System	Microsoft® Windows Server® 2003 Enterprise
SQL Server	version 2008
Time Machine	Version 7.0.0-S
Note:	<a href="#">Time Machine has been certified on Microsoft SQL Server 2008</a>

- 1) Click [Start] then “Administrative Tools” and select Services.
- 2) In the right pane of the Service applet locate “SQL Server (MSSQLSERVER)”.



- 3) Right-click on the entry for “SQL Server (MSSQLSERVER)” and select the “Log On” tab.
- 4) Click the “This account” radio button and enter a user name other than NETWORK or LOCAL SERVICE.



- 5) Repeat the same steps for the service labeled "SQL Server Agent (MSSQLSERVER)".
- 6) Verify the change has been accepted by restarting all three services. To do this just right click the entries for each and select Restart.
- 7) From a command line, assign a virtual date to the SQL Server account.  
 Example: `tmuser -a -u SQL -y 5`  
 (Sets SQL server account to see a virtual date 5 years ahead of the system time.)
- 8) Verify the virtual date is running for SQL account by listing the running virtual clocks.

EXAMPLE:

```
C:\>tmuser -a -u SQL -y 5
```

```
C:\>tmuser list
```

Listing all virtual clocks:

```
ACCOUNT                CLOCK INFORMATION
-----
(U) W2003SE\SQL        Wed May 18 2016 17:38:05
                        GMT-08:00) Pacific Time (US & Canada)
                        Virtual date and time in system's time zone.
                        Clock runs in normal speed.
-----
```

- 9) Verify the SQL Server is getting the virtual date by executing a "date" call to the SQL server from the command line (CMD.exe).

```
C:\>sqlcmd -d AdventureWorks -Q "select getdate()"
```

```
-----
2016-05-18 17:44:59.310
(1 rows affected)
```

## **Setting Up a Second Instance**

With one instance of SQL; all users connect to that instance will see the same virtual clock. It is sometimes useful to have multiple SQL instances; so that different users can test concurrently with different virtual clocks. Or one group of users can see one virtual clock and other group can see the system clock.

To set up a second instance for concurrent time testing, install SQL under a different user with a different SID.

Remember to change the Log On As in Services for the second instance of SQL under the new SID.

Then different virtual clocks can be set against different SID to enable concurrent testing of different virtual clocks.