PROWESS

Large Microsoft® SQL Server® Database Performance on Dell EMC™ PowerEdge™ Servers and VMware vSphere®: Methodology Report

This methodology report outlines the configuration and testing steps performed by the Prowess Consulting engineers to test large Microsoft® SQL Server® database performance. These steps assume the use of a Dell EMC™ PowerEdge™ server with an Integrated Dell Remote Access Controller (iDRAC) and an external SAN storage device. For this performance report, we used a Dell EMC PowerStore 7000T configured as a VMware datastore.

Throughout this report, the term “host” refers to the physical server you are configuring.

Configure the Host

1. Create volumes greater than 40 TB on the SAN storage device.
2. Map the storage volumes to the host.
3. Attach the server’s operating system media to the server’s iDRAC using the virtual console.
4. Install VMware ESXi™ on the server’s local hard disk or solid-state drive (SSD). In our tests, we installed VMware ESXi 6.7 on a Dell EMC PowerEdge R730xd, and we installed VMware ESXi 7 on a Dell EMC PowerEdge R740xd.
5. Reboot the server.
6. Once ESXi is running, configure a static IP address and enable Secure Shell (SSH).

Configure VMware vCenter® and Create a Virtual Machine (VM)

1. Deploy the VMware vCenter VM to the host for the version of VMware vSphere® that you are running.
2. Once vCenter® is deployed, configure a static IP address for the vCenter VM.
3. Connect to vCenter using a web browser and the static IP address.
4. Create a new data center and add the ESXi host to the data center.
5. Create a new datastore using the attached SAN storage volume.
6. Create a VM on the newly created datastore with the following configuration:

VM Configuration Parameter	VMware vSphere® 6.7 and Windows Server® 2012 R2	vSphere 7 and Windows Server 2019
CPU	**16 vCPUs**	**24 vCPUs**
Memory	**128 GB**	**128 GB**
OS virtual disk	**50 GB using an IDE controller for the older operating system, placed on SAN storage**	**50 GB, placed on SAN storage**
SQL Server data virtual disk	**31 TB thick eager zeroed virtual disk, placed on SAN storage**	**31 TB thick eager zeroed virtual disk, placed on SAN storage**
SQL Server log file virtual disk	**7.5 TB thick eager zeroed virtual disk, placed on SAN storage**	**7.5 TB thick eager zeroed virtual disk, placed on SAN storage**
Virtual disk storage controller:	**Physical sharing: Paravirtual type**	**Physical sharing: Paravirtual type**

7. Install the Windows Server guest operating system on the VM with a 50 GB virtual disk.

Configure Windows Server for Each VM

From the Windows Server Manager:

1. Configure a static IP address for the Windows Server VM.
2. Disable the Windows Defender firewall.
3. Disable Defender Antivirus.
4. Disable Internet Explorer (IE) protection.
5. Enable remote desktop.
6. Install updates.
7. Once the updates are installed, pause any further updates.

Configure the VM Disks for Use by SQL Server

1. In the VMware console or from a Remote Desktop session to the Windows Server VM, launch a command prompt.
2. Enter the following commands to enable quick format:

```
fsutil behavior query DisableDeleteNotify, verify settings
fsutil behavior set disabledeletenotify 1
```

3. Launch the Windows Disk Management tool.
4. Right-click the data disk, and then select **Online**.
5. Right-click the data disk, and then select **Initialize**.
6. Leave the partition style as **GPT**, and then select **OK**.
7. Right-click the data disk, and then select **New Simple Volume**.
8. The New simple volume wizard displays. Select **Next**.
9. Assign an appropriate drive letter, and then select **Next**.

10. Format the volume with the following settings:
    - File System: NTFS
    - Allocation Unit Size: 64K
    - Volume Label: Data
11. Select **Next**, and then select **Finish**.
12. Repeat steps 4 through 10 for the log virtual hard disk.
13. Launch a command prompt.
14. At the command prompt, enter the following commands to disable quick format:

```
Fsutil behavior query DisableDeleteNotify, verify settings
Fsutil behavior set DisableDeleteNotify 0
```

Install SQL Server 2019 or SQL Server 2012 and HammerDB

1. Attach the SQL Server media to the Windows Server VM.
2. Run setup.exe.
3. Select **Installation**.
4. Select **New SQL Server stand-alone installation or add features to an existing installation.**
   Perform a new installation.
5. Enter the product key, and then select **Next.**
6. Accept the license agreement, and then select **Next.**
7. Once the installer has finished the Install Rules scan, select **Next.**
8. Select **Database Engine Services**, and then select **Next**
9. Select **Default instance**, and then select **Next.**
10. Accept the default for **Service Accounts,** and then select **Next.**
11. On the **Server Configuration** tab:
    a. Select **Mixed Mode**.
    b. Enter a valid system administrator (SA) password in the **Enter password** and **Confirm password** fields.
    c Select **Add Current User** to add the current user as a SQL Server administrator.
12. On the Data Directories tab:
    a. Enter a data drive path in the **User Database Directory** field.
    b. Enter a log drive path in the **User database log directory** field
13. Complete the installation wizard.
14. Install SQL Server Management Studio.
15. Install HammerDB using the default settings.

Apply SQL Server Best Practices

1. Using SQL Server Management Studio, enable Max Server Memory using the following commands. Note that the value after ‘max server memory’ will vary depending on your server configuration.

```
sp_configure 'show advanced options', 1;

GO

RECONFIGURE;

GO

sp_configure 'max server memory',125824;

GO

RECONFIGURE;

GO
```

2. Configure Lock Pages in Memory:
    a. In the Local Group Policy Editor console, expand **Computer Configuration**, and then expand **Windows Settings**.
    b. Expand **Security Settings**, and then expand **Local Policies**.
    c. Select the **User Rights Assignment** folder. The policies are displayed in the details pane.
    d. Double-click **Lock Pages in Memory**.
    e. On the Local Security Setting tab, select **Add User or Group**.
    f. In the **Select Users, Service Accounts, or Groups** dialog box, add an account that has privileges to run sqlservr.exe.
3. Sign out, and then sign back in to implement the change.

Configure the ODBC Driver to Connect to the SQL Server Instance

1. Select the **Windows Start** button, and then search for **ODBC Data Sources (64-bit)**.
2. On the **User DSN** tab:
    a. Select **Add**.
    b. Select **ODBC Driver 17 for SQL Server (Driver 11 for W12R2)**.
    c. Select **Finish**.
3. Specify a name for the connection in the **Name** field.
4. Add the SQL server connection hostname or IP address in the **Server** field.
5. Select **Next**.
6. Leave the authentication settings default if SQL was installed under the current user. Otherwise, specify the correct SQL authentication details.
7. Select **Next,** and then select **Finish**.
8. Select **Test Data Source** to verify your SQL connection.

Configure HammerDB

1. In Windows Explorer, navigate to the HammerDB installation folder. Normally this is located under C:\Program Files.
2. In the config directory, edit the mssqlserver.xml file so that it matches the following:

```
<?xml version="1.0" encoding="utf-8"?>
<mssqlserver>
    <connection>
        <mssqls_server>(local)</mssqls_server>
        <mssqls_linux_server>localhost</mssqls_linux_server>
        <mssqls_tcp>false</mssqls_tcp>
        <mssqls_port>1433</mssqls_port>
        <mssqls_azure>false</mssqls_azure>
        <mssqls_authentication>sql</mssqls_authentication>
        <mssqls_linux_authent>sql</mssqls_linux_authent>
      <mssqls_odbc_driver>ODBC Driver 17 for SQL Server</mssqls_odbc_driver>
      <mssqls_linux_odbc>ODBC Driver 17 for SQL Server</mssqls_linux_odbc>
        <mssqls_uid>sa</mssqls_uid>
        <mssqls_pass>Password2020!</mssqls_pass>
    </connection>
    <tpcc>
        <schema>
            <mssqls_count_ware>1</mssqls_count_ware>
            <mssqls_num_vu>1</mssqls_num_vu>
            <mssqls_dbase>tpcc</mssqls_dbase>
            <mssqls_imdb>false</mssqls_imdb>
            <mssqls_bucket>1</mssqls_bucket>
            <mssqls_durability>SCHEMA_AND_DATA</mssqls_durability>
        </schema>
        <driver>
            <mssqls_total_iterations>1000000</mssqls_total_iterations>
            <mssqls_raiseerror>false</mssqls_raiseerror>
            <mssqls_keyandthink>false</mssqls_keyandthink>
            <mssqls_checkpoint>false</mssqls_checkpoint>
            <mssqls_driver>timed</mssqls_driver>
            <mssqls_rampup>10</mssqls_rampup>
            <mssqls_duration>20</mssqls_duration>
            <mssqls_allwarehouse>true</mssqls_allwarehouse>
            <mssqls_timeprofile>false</mssqls_timeprofile>
            <mssqls_async_scale>false</mssqls_async_scale>
            <mssqls_async_client>10</mssqls_async_client>
            <mssqls_async_verbose>false</mssqls_async_verbose>
            <mssqls_async_delay>1000</mssqls_async_delay>
        </driver>
    </tpcc>
    <tpch>
        <schema>
          <mssqls_scale_fact>1</mssqls_scale_fact>
          <mssqls_maxdop>2</mssqls_maxdop>
          <mssqls_tpch_dbase>tpch</mssqls_tpch_dbase>
          <mssqls_num_tpch_threads>1</mssqls_num_tpch_threads>
          <mssqls_colstore>false</mssqls_colstore>
        </schema>
        <driver>
          <mssqls_total_querysets>1</mssqls_total_querysets>
          <mssqls_raise_query_error>false</mssqls_raise_query_error>
          <mssqls_verbose>false</mssqls_verbose>
          <mssqls_refresh_on>false</mssqls_refresh_on>
          <mssqls_update_sets>1</mssqls_update_sets>
          <mssqls_trickle_refresh>1000</mssqls_trickle_refresh>
          <mssqls_refresh_verbose>false</mssqls_refresh_verbose>

        </driver>

    </tpch>

</mssqlserver>
```

Configure Windows Performance Monitor

1. Open the Computer Management app in the Windows Server VM.
2. Expand **Performance**, and then expand **Data Collector Sets**.
3. Right-click **User-defined**, select **New**, and then select **Data Collector Set**.
4. Enter a name in the **Name** field.
5. Select **Create manually (Advanced)**, and then select **Next**.
6. Select the **Performance counter**, **Event trace data**, and **System configuration information** check boxes.
7. Select **Next**.
8. Select **Add**.
9. Select each of the following items from the **Available counters** section, and then select **Add**.

   Processor Information(_Total)\% Processor Time

   Memory\Available MBytes

   PhysicalDisk(_Total)\Avg. Disk sec/Read

   PhysicalDisk(_Total)\Avg. Disk sec/Write

   PhysicalDisk(_Total)\Disk Reads/sec

   PhysicalDisk(_Total)\Disk Writes/sec

   Processor(_Total)\% Processor Time

   SQLServer:General Statistics\User Connections

   SQLServer:SQL Statistics\Batch Requests/sec

   SQLServer:SQL Statistics\SQL Compilations/sec

   SQLServer:SQL Statistics\SQL Re-Compilations/sec

   System\Processor Queue Length

   Process(_Total)\% Privileged Time

   Process(_Total)\% User Time

   Process(_Total)\IO Read Bytes/sec

   Process(_Total)\IO Write Bytes/sec

   Processor(_Total)\% Privileged Time

   Processor(_Total)\% User Time

   Process(_Total)\% Processor Time

   Average Disk Sec/Transfer

   Average Disk Queue Length

   Current Disk Queue Length

   %Disk Time

   %Idle Time

   Buffer Manager: Page Reads/Sec

   Page Writes/Sec

   Current Disk Queue Length

Avg. Disk Bytes / Transfer

Avg. Disk sec / Transfer

Disk Bytes / sec

Disk Transfers / sec

10. Select **OK**, and then select **Finish**.

Create the SQL Server Database

Using SQL Server Management Studio, connect to the SQL Server instance and create the 20 TB and 30 TB databases using the following SQL commands:

20 TB TPC-C SQL Server Database Creation Script

```
CREATE DATABASE [tpcc]

 CONTAINMENT = NONE

 ON  PRIMARY

( NAME = N'tpcc', FILENAME = N'E:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\
MSSQL\Data\tpcc.mdf' , SIZE = 10240GB , FILEGROWTH = 10%),

( NAME = N'tpcc1', FILENAME = N'E:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\
MSSQL\Data\tpcc1.ndf' , SIZE = 10240GB , FILEGROWTH = 10%)

 LOG ON

( NAME = N'tpcc_log', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\
MSSQL\Data\tpcc_log.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log1', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL15.
MSSQLSERVER\MSSQL\Data\tpcc_log1.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log2', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL15.
MSSQLSERVER\MSSQL\Data\tpcc_log2.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log3', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL15.
MSSQLSERVER\MSSQL\Data\tpcc_log3.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log4', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL15.
MSSQLSERVER\MSSQL\Data\tpcc_log4.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%)

GO

ALTER DATABASE [tpcc] SET COMPATIBILITY_LEVEL = 150

GO

ALTER DATABASE [tpcc] SET ANSI_NULL_DEFAULT OFF

GO

ALTER DATABASE [tpcc] SET ANSI_NULLS OFF
```

```
GO

ALTER DATABASE [tpcc] SET ANSI_PADDING OFF

GO

ALTER DATABASE [tpcc] SET ANSI_WARNINGS OFF

GO

ALTER DATABASE [tpcc] SET ARITHABORT OFF

GO

ALTER DATABASE [tpcc] SET AUTO_CLOSE OFF

GO

ALTER DATABASE [tpcc] SET AUTO_SHRINK OFF

GO

ALTER DATABASE [tpcc] SET AUTO_CREATE_STATISTICS ON(INCREMENTAL = OFF)

GO

ALTER DATABASE [tpcc] SET AUTO_UPDATE_STATISTICS ON

GO

ALTER DATABASE [tpcc] SET CURSOR_CLOSE_ON_COMMIT OFF

GO

ALTER DATABASE [tpcc] SET CURSOR_DEFAULT  GLOBAL

GO

ALTER DATABASE [tpcc] SET CONCAT_NULL_YIELDS_NULL OFF

GO

ALTER DATABASE [tpcc] SET NUMERIC_ROUNDABORT OFF

GO

ALTER DATABASE [tpcc] SET QUOTED_IDENTIFIER OFF

GO

ALTER DATABASE [tpcc] SET RECURSIVE_TRIGGERS OFF

GO

ALTER DATABASE [tpcc] SET  DISABLE_BROKER

GO
```

```
ALTER DATABASE [tpcc] SET AUTO_UPDATE_STATISTICS_ASYNC OFF

GO

ALTER DATABASE [tpcc] SET DATE_CORRELATION_OPTIMIZATION OFF

GO

ALTER DATABASE [tpcc] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [tpcc] SET READ_COMMITTED_SNAPSHOT OFF

GO

ALTER DATABASE [tpcc] SET  READ_WRITE

GO

ALTER DATABASE [tpcc] SET RECOVERY FULL

GO

ALTER DATABASE [tpcc] SET  MULTI_USER

GO

ALTER DATABASE [tpcc] SET PAGE_VERIFY CHECKSUM

GO

ALTER DATABASE [tpcc] SET TARGET_RECOVERY_TIME = 60 SECONDS

GO

ALTER DATABASE [tpcc] SET DELAYED_DURABILITY = DISABLED

GO

USE [tpcc]

GO

ALTER DATABASE SCOPED CONFIGURATION SET LEGACY_CARDINALITY_ESTIMATION = Off;

GO

ALTER DATABASE SCOPED CONFIGURATION FOR SECONDARY SET LEGACY_CARDINALITY_ESTIMATION = Primary;

GO

ALTER DATABASE SCOPED CONFIGURATION SET MAXDOP = 0;

GO
```

```
ALTER DATABASE SCOPED CONFIGURATION FOR SECONDARY SET MAXDOP = PRIMARY;

GO

ALTER DATABASE SCOPED CONFIGURATION SET PARAMETER_SNIFFING = On;

GO

ALTER DATABASE SCOPED CONFIGURATION FOR SECONDARY SET PARAMETER_SNIFFING = Primary;

GO

ALTER DATABASE SCOPED CONFIGURATION SET QUERY_OPTIMIZER_HOTFIXES = Off;

GO

ALTER DATABASE SCOPED CONFIGURATION FOR SECONDARY SET QUERY_OPTIMIZER_HOTFIXES = Primary;

GO

USE [tpcc]

GO

IF NOT EXISTS (SELECT name FROM sys.filegroups WHERE is_default=1 AND name = N'PRIMARY')
ALTER DATABASE [tpcc] MODIFY FILEGROUP [PRIMARY] DEFAULT

GO
```

30 TB TPC-C SQL Server Database Creation Script

```
CREATE DATABASE [tpcc]

 CONTAINMENT = NONE

 ON  PRIMARY

( NAME = N'tpcc', FILENAME = N'E:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\
MSSQL\Data\tpcc.mdf' , SIZE = 10240GB , FILEGROWTH = 10%),

( NAME = N'tpcc1', FILENAME = N'E:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\
MSSQL\Data\tpcc1.ndf' , SIZE = 10240GB , FILEGROWTH = 10%),

( NAME = N'tpcc2', FILENAME = N'E:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\
MSSQL\Data\tpcc2.ndf' , SIZE = 10240GB , FILEGROWTH = 10%)

 LOG ON

( NAME = N'tpcc_log', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\
MSSQL\Data\tpcc_log.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log1', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.
MSSQLSERVER\MSSQL\Data\tpcc_log1.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log2', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.
MSSQLSERVER\MSSQL\Data\tpcc_log2.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log3', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.
MSSQLSERVER\MSSQL\Data\tpcc_log3.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log4', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.
MSSQLSERVER\MSSQL\Data\tpcc_log4.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log5', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.
MSSQLSERVER\MSSQL\Data\tpcc_log5.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%),

( NAME = N'tpcc_log6', FILENAME = N'F:\Program Files\Microsoft SQL Server\MSSQL11.
MSSQLSERVER\MSSQL\Data\tpcc_log6.ldf' , SIZE = 1073741824KB , FILEGROWTH = 10%)

GO

ALTER DATABASE [tpcc] SET COMPATIBILITY_LEVEL = 110

GO

ALTER DATABASE [tpcc] SET ANSI_NULL_DEFAULT OFF

GO

ALTER DATABASE [tpcc] SET ANSI_NULLS OFF

GO

ALTER DATABASE [tpcc] SET ANSI_PADDING OFF

GO

ALTER DATABASE [tpcc] SET ANSI_WARNINGS OFF

GO

ALTER DATABASE [tpcc] SET ARITHABORT OFF

GO

ALTER DATABASE [tpcc] SET AUTO_CLOSE OFF

GO

ALTER DATABASE [tpcc] SET AUTO_CREATE_STATISTICS ON

GO

ALTER DATABASE [tpcc] SET AUTO_SHRINK OFF

GO

ALTER DATABASE [tpcc] SET AUTO_UPDATE_STATISTICS ON

GO
```

```
ALTER DATABASE [tpcc] SET CURSOR_CLOSE_ON_COMMIT OFF

GO

ALTER DATABASE [tpcc] SET CURSOR_DEFAULT  GLOBAL

GO

ALTER DATABASE [tpcc] SET CONCAT_NULL_YIELDS_NULL OFF

GO

ALTER DATABASE [tpcc] SET NUMERIC_ROUNDABORT OFF

GO

ALTER DATABASE [tpcc] SET QUOTED_IDENTIFIER OFF

GO

ALTER DATABASE [tpcc] SET RECURSIVE_TRIGGERS OFF

GO

ALTER DATABASE [tpcc] SET  DISABLE_BROKER

GO

ALTER DATABASE [tpcc] SET AUTO_UPDATE_STATISTICS_ASYNC OFF

GO

ALTER DATABASE [tpcc] SET DATE_CORRELATION_OPTIMIZATION OFF

GO

ALTER DATABASE [tpcc] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [tpcc] SET READ_COMMITTED_SNAPSHOT OFF

GO

ALTER DATABASE [tpcc] SET  READ_WRITE

GO

ALTER DATABASE [tpcc] SET RECOVERY FULL

GO

ALTER DATABASE [tpcc] SET  MULTI_USER

GO

ALTER DATABASE [tpcc] SET PAGE_VERIFY CHECKSUM
```

```
GO

ALTER DATABASE [tpcc] SET TARGET_RECOVERY_TIME = 0 SECONDS

GO

USE [tpcc]

GO

IF NOT EXISTS (SELECT name FROM sys.filegroups WHERE is_default=1 AND name = N'PRIMARY')
ALTER DATABASE [tpcc] MODIFY FILEGROUP [PRIMARY] DEFAULT

GO
```

Create the HammerDB Database Schema

1. Start the HammerDB application.
2. In the **Benchmark** pane, expand **SQL Server**.
3. Expand **Schema Build**, and then select **Options**.
4. Set **Number of Warehouses** to **1000**.
5. Set **Virtual Users to build schema** to **100**.
6. Select **OK**, and then select **Build**.

Configure the HammerDB Test

1. In the HammerDB window, load the preconfigured driver script.
2. Expand **Driver Script**.
3. Select **Load**.
4. Expand **Virtual Users**.
5. Set **Virtual Users** to **50**.
6. Select the **Log Output to Temp** check box.
7. Select the **User Unique Log Name** check box.
8. Select the **Log Timestamps** check box.
9. Select **OK**.
10. Before running HammerDB, verify that you are capturing host and VM performance data.
11. To capture host data:
    a. Connect to the ESXi host using SSH.
    b. Run the following command for each test pass
    ```
    esxtop -b -a -d 10 > filename.csv
    ```
    c. On the VM, start the data collector you previously configured.
12. Select **Run** to run the test.
13. Complete the test for 50, 100, 500, and 1,000 virtual users.

The analysis in this document was done by Prowess Consulting and commissioned by Dell Technologies.
Prowess and the Prowess logo are trademarks of Prowess Consulting, LLC.