



The science behind the report

Get answers faster by upgrading to a Dell EMC PowerEdge R740xd running Microsoft SQL Server on Linux 2017 Standard on Red Hat Enterprise Linux 7.5

This document describes what we tested, how we tested, and what we found. To learn how these facts translate into real-world benefits, read the report [Analyze data faster by upgrading to a Dell EMC PowerEdge R740xd running Microsoft SQL Server 2017 Standard on Red Hat Enterprise Linux 7.5](#).

On April 17, 2018, we finalized the hardware and software configurations we tested. Updates for current and recently released hardware and software appear often, so unavoidably these configurations may not represent the latest versions available when this report appears. For older systems, we chose configurations representative of typical purchases of those systems. We concluded hands-on testing on May 3, 2018.

Our results

The table below presents our findings in detail.

	Query time (s)	Drive read latency (ms)	Drive read throughput (MB/s)
Dell EMC™ PowerEdge™ R740xd	799	0.35	160
Dell EMC PowerEdge R720xd	10,169	36	439
Percentage improvement with upgrade	92%	99%	63%

System configuration information

The table below presents detailed information on the systems we tested.

Server configuration information	Dell EMC PowerEdge R740xd	Dell EMC PowerEdge R720xd
BIOS name and version	Dell 1.3.7	Dell 2.6.1
Non-default BIOS settings	None	Virtualization enabled
Operating system name and version/build number	Red Hat® Enterprise Linux® 7.5, kernel version 3.10.0-862.el7.x86_64	Windows Server® 2012 R2 Standard, Version 6.3, Build 9600
Date of last OS updates/patches applied	04/17/2018	03/21/18
Power management policy	Performance	Performance
Processor		
Number of processors	2	2
Vendor and model	Intel® Xeon® Gold 6130	Intel Xeon E5-2640
Core count (per processor)	16	6
Core frequency (GHz)	2.10	2.50
Stepping	H0	C2
Memory module(s)		
Total memory in system (GB)	192	192
Number of memory modules	12	12
Vendor and model	Hynix HMA82GR7AFR8N-VK	Hynix HMT42GR7MFR4C-PB
Size (GB)	16	16
Type	DDR4-2666	PC3-12800R
Speed (MHz)	2,666	1,333
Speed running in the server (MHz)	2,666	1,333
Storage controller		
Vendor and model	Dell PERC H730P Mini	Dell PERC H710P Mini
Cache size (GB)	1GB	1GB
Firmware version	25.5.4.0006	21.3.5-0002
Driver version	6.604.06.00	6.805.03.00
Local storage (type A)		
Number of drives	2	24
Drive vendor and model	Dell ST300MP0026	Dell ST9300653SS
Drive size (GB)	300	300
Drive information (speed, interface, type)	15K, 12Gbps, SAS, HDD	15K, 6Gbps, SAS, HDD

Server configuration information	Dell EMC PowerEdge R740xd	Dell EMC PowerEdge R720xd
Local storage (type B)		
Number of drives	12	N/A
Drive vendor and model	Dell THNSF8960CCSE	N/A
Drive size (GB)	960	N/A
Drive information (speed, interface, type)	6Gbps, SATA, SSD	N/A
Network adapter		
Vendor and model	QLogic BCM57800 10 GbE	Intel Gigabit 4P I350-t rNDC
Number and type of ports	2 x 1GbE, 2 x 10GbE	4 x 1GbE
Driver version	7.13.104.0	12.14.7.0
Cooling fans		
Vendor and model	Nidec UltraFlo 4VXP3-X30	San Ace 60 9GA0612P1J611
Number of cooling fans	6	6
Power supplies		
Vendor and model	Dell D1600E-S0	Dell E1100E-S0
Number of power supplies	2	2
Wattage of each (W)	1,600	1,100

How we tested

We ran the TPC-H-like test from the HammerDB suite on a Dell EMC PowerEdge R720xd with 24 15K SAS HDDs. We began by splitting up the disks into three RAID10 volumes. We used two disks for our OS volume, 16 disks for our Data volume, and six disks for our Log volume. We then moved the TempDB and TempLog files to the Data and Log volumes, respectively. We started testing with the PowerEdge R720xd, tuning the MAXDOP until the benchmark ran with shortest total completion time (including refresh).

We ran the same test on the PowerEdge R740xd server configured with two 300GB, 12Gbps, 15K SAS HDDs and 12 960GB, 6Gbps SATA SSDs. We used the two HDDs in a RAID1 configuration for our OS volume, 10 SSDs in RAID10 for our Data volume, and the remaining two SSDs in RAID1 for our Log volume.

Installing Microsoft® Windows Server 2012 R2 on the Dell EMC PowerEdge R720xd server

1. Insert the installation media into the CD/DVD drive, and restart the server.
2. When the option appears, press F11 to enter the Boot Manager.
3. Select BIOS Boot Menu.
4. Select SATA Optical Drive, and press Enter.
5. When prompted to boot from DVD, press any key.
6. When the installation screen appears, click My language is English (United States).
7. Leave the language, time/currency format, and input method as default, and click Next.
8. Click Install now.
9. When the installation prompts you, enter the product key.
10. Select Windows Server 2012 Standard (Server with a GUI), and click Next.
11. Check I accept the license terms, and click Next.
12. Click Custom: Install Windows only (advanced).
13. Select Drive 0 Unallocated Space, and click Next. Windows will begin and restart automatically.
14. At the Settings page, fill in the Password and Reenter Password fields with the same password.
15. Log in with the password you set up in the previous step.

Configuring Windows Update on the Dell EMC PowerEdge R720xd server

1. In the left pane of the Server Manager window, click Local Server.
2. In the main frame, next to Windows Update, click Not configured.
3. In the Windows Update window, in the main pane, click Let me choose my settings.
4. Under Important updates, select Never check for updates (not recommended), and click OK.
5. In the left pane, click Check for updates, and install all available updates.
6. Close the Windows Update window.

Disabling the Windows Update Service on the Dell EMC PowerEdge R720xd server

1. After you have installed all updates, press the Start Menu button, and type `Services`.
2. Scroll down, and right-click on the Windows Update service.
3. Click Properties.
4. Choose Disabled as the startup type.
5. Stop the service if it is running.
6. Click Apply.
7. Click OK to close the window.

Formatting a Windows NTFS volume for Microsoft SQL Server on the Dell EMC PowerEdge R720xd server

1. Right-click the start icon, and click Disk Management.
2. Right-click the disk, and click Online.
3. Right-click the disk again and click Initialize.
4. Right click an unallocated area, and select New Simple Volume.
5. Click Next.
6. Click Next.
7. Assign a letter path, and click Next.
8. Set the Allocation Unit Size to 64K, and name the Volume Label.
9. Click Next.
10. Click Finish.
11. Repeat steps 1 through 10 for all data and log volumes.

Installing Microsoft SQL Server 2008 R2 on the PowerEdge R720xd server

1. Mount the installation DVD for SQL Server 2008 R2.
2. Click Run SETUP.EXE. If Autoplay does not begin the installation, navigate to the SQL Server 2008 DVD, and double-click it.
3. In the left pane, click Installation.
4. Click New SQL Server stand-alone installation or add features to an existing installation.
5. Select the Enter the product key radio button, and enter the product key. Click Next.
6. To accept the license terms, click the checkbox, and click Next.
7. To check for updates, click Use Microsoft Update, and click Next.
8. To install the setup support files, click Install
9. If no failures appear, click Next.
10. At the Setup Role screen, choose SQL Server Feature Installation, and click Next.
11. At the Feature Selection screen, select the following, and click Next:
 - a. Database Engine Services
 - b. Full-Text and Semantic Extractions for Search
 - c. Client Tools Connectivity
 - d. Client Tools Backwards Compatibility
 - e. Management Tools—Basic
 - f. Management Tools—Complete
12. At the Installation Rules screen, after the check completes, click Next.
13. At the Instance configuration screen, leave the default selection of default instance, and click Next.
14. At the Server Configuration screen, choose NT Service\SQLSERVERAGENT for SQL Server Agent, and choose NT Service\MSSQLSERVER for SQL Server Database Engine.
15. Change the Startup Type to Automatic.

16. Click Next.
17. At the Database Engine Configuration screen, select the authentication method you prefer. In our tests, we selected Mixed Mode.
18. Enter and confirm a password for the system administrator account.
19. Click Add Current user. This may take several seconds.
20. Click Next.
21. At the Error and usage reporting screen, click Next.
22. At the Installation Configuration Rules screen, check to ensure there are no failures or relevant warnings, and click Next.
23. At the Ready to Install screen, click Install.
24. After installation completes, click Close.
25. Close the installation window.

Moving and expanding tempdb on the PowerEdge R720xd server

1. Open SQL Server Management Studio.
2. Expand Databases→System Databases
3. Right-click tempdb, and choose Properties.
4. Set location of tempdev to Data volume and templog to Log volume.
5. Click Files and enter a new Initial Size (MB) for both files. The size required will depend on the scale of your TPC-H database. We set tempdev to 50 GB and templog to 20 GB to ensure more than enough space.
6. When the files are finished expanding, click OK.

Enabling Lock Pages in Memory

1. Click the Start menu, type `gpedit.msc`, and hit Enter.
2. Once the Group Policy console opens, expand Computer Configuration→Windows Settings→Security Settings→Local Policies.
3. Select User Rights Assignment.
4. In the right-hand pane, double-click Lock pages in memory.
5. Click Add user or group.
6. Add the NT SERVICE\MSSQLSERVER account, and click OK.

Creating a TCPH Database

1. Open SQL Server Management Studio.
2. Right-click Databases→New Database.
3. Name the database. We named ours `tpch`.
4. Click Add to add seven additional database files for a total of eight.
5. Name the database files, and click OK.

Installing HammerDB on Windows

1. Download the latest version of HammerDB from www.hammerdb.com/download.html
2. Double-click the exe file, choose English, and click OK.
3. Click Yes.
4. Click Next.
5. Choose a destination location, and click Next.
6. Click Next.
7. Click Finish.

Installing and configuring Red Hat Enterprise Linux 7.5 on the Dell EMC PowerEdge R740XD

1. Boot the server from the Red Hat Enterprise Linux 7.5 installation DVD
2. Modify the default disk partitioning to use only the volume allocated to the operating system and to create the following partitions:
 - a. 50GiB for `/root`
 - b. 200GiB for `/var`
 - c. 23.67 GiB for `/opt`
 - d. 200 MiB for `/boot`
 - e. 4096 MiB for swap.

3. Choose Minimal Install.
4. Disable kdump.
5. Enable the first active network port, and configure it. (For example, assign static IP address of 10.220.64.11/16 with gateway of 10.220.0.1, and DNS server of 10.41.0.10).
6. Set the hostname to `linux-db-01`.
7. Set the time zone to Eastern/US; enable NTP to sync only from 10.40.0.1
8. Click Install.
9. Set the root password.
10. Reboot when the install has finished.
11. Log in as root.
12. Connect to the Red Hat repository and update the operating system software:


```
subscription-manager register
subscription-manager attach --auto
yum -y update
```
13. Stop and disable a few unnecessary services:


```
for i in postfix.service firewalld.service ; do
  systemctl disable $i
  systemctl stop $i
done
```
14. Disable SELinux:


```
sed -i 's/^SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config
```
15. Set the tuning profile to throughput-performance:


```
tuned-adm profile throughput-performance
```
16. Modify kernel parameters as suggested by the guide at the following URL: <https://docs.microsoft.com/en-us/sql/linux/sql-server-linux-performance-best-practices?view=sql-server-linux-2017>

```
cat >> /etc/sysctl.d/99-sysctl.conf <<EOF
kernel.numa_balancing=0
vm.max_map_count=262144
EOF
```
17. Prepare data and log volumes for SQL Server:


```
for disk in /dev/sd{b,c} ; do
echo $disk
  parted $disk mklabel gpt
  parted $disk mkpart mkpart primary "1 -1"
  mkfs.xfs ${disk}1
done

mkdir -p /opt/db/{data,logs}
cat >> /etc/fstab <<EOF
#
# MSSQL data and log areas
/dev/sdb1 /opt/db/data xfs defaults 0 0
/dev/sdc1 /opt/db/logs xfs defaults 0 0
EOF

mount -a
chown mssql:mssql /opt/db/{data,logs}
```
18. Reboot the server:


```
shutdown -r now
```

Installing Microsoft SQL Server 2017 on Red Hat Enterprise Linux 7.5

We closely followed the installation instructions in the guide, [Quickstart: Install SQL Server and create a database on Red Hat](#).

1. Log in as root.
2. Obtain and add the Microsoft's RHEL7 software repositories for SQL Server 2017 on Linux:


```
curl -o /etc/yum.repos.d/mssql-server.repo \
  https://packages.microsoft.com/config/rhel/7/mssql-server-2017.repo
curl -o /etc/yum.repos.d/msprod.repo \
  https://packages.microsoft.com/config/rhel/7/prod.repo
```

3. Install Microsoft SQL Server (including tools and the full-text searches for SQL Server) and Linux dependencies. Note that the installation requires user interaction to accept EULAs.

```
yum remove -y unixODBC-utf16 unixODBC-utf16-devel dos2unix
yum install -y mssql-server mssql-tools mssql-server-fts unixODBC-devel
```

4. Perform initial set up of Microsoft SQL Server by running the following command. Choose Standard edition; accept the license; and set the password for the account named SA to `Passw0rd@`

```
export PATH=/opt/mssql/bin:/opt/mssql-tools/bin:$PATH
mssql-conf setup
```

5. Perform additional configuration of SQL Server, including setting the location of its default data and log directories, and setting its memory limit to approximately 95 percent of the available memory.

```
mssql-conf set filelocation.defaultdatadir /opt/db/data/
mssql-conf set filelocation.defaultlogdir /opt/db/logs/
mssql-conf set memory.memorylimitmb 183880
mssql-conf set telemetry.customerfeedback false
```

6. Restart SQL Server.

```
systemctl restart mssql-server
```

7. We used Microsoft SQL Server Management Studio v17.6 to perform these final configuration step, though we could have performed them with the Linux SQL command-line program.
 - a. Set the process affinity for the Processors and I/O by right clicking on the Server Instance and selecting Properties→Processors.
 - b. Set the Processor Affinity to use all 32 hyperthreads on NUMA node 0 (CPUS 0-31) and the first 16 hyperthreads on NUMA node 1 (CPUS 32-47). This is the maximum number of CPUs permitted under a Standard Edition license.
 - c. Select Automatically set I/O affinity mask for all processors.
 - d. Click OK.
8. We resized the existing datafile and created seven additional data files for the instance's tempdb so that there were eight 8GB data files total.
 - a. Expose the "tempdb" database under Databases→System Databases.
 - b. Right-click tempdb→Properties→Files.
 - c. Resize the sole data file to 8,192MB with 0 autogrowth and ensured it was located in directory /opt/db/data.
 - d. Create seven addition data files, tempdb2.ndf, ..., tempdb8.ndf, of size 8,92MB with 0 autogrowth in directory /opt/db/data.
 - e. Click OK.

Installing HammerDB on Linux

We used HammerDB version 2.23.

1. Log onto the Linux server.
2. Install additional RPMs for X Windows and Tcl/Tk.

```
yum install -y xterm libX11 xauth libXss libXScrnSaver-devel
```
3. Download and install Hammer DB version 2.23.

```
./HammerDB-2.23-Linux-x86-64-Install
```

Generating the HammerDB TPC-H 300-scale database

We used the following steps to generate a 300-scale TPC-H-like database using HammerDB. For the 12G solution, we generated the data locally. For Linux, we obtained the TPC-H-like data from a full backup file of a 300-scale dataset from another SQL Server 2017 instance.

1. Open HammerDB, and click Options→Benchmark.
2. Choose MSSQL Server and TPC-H.
3. Expand SQL Server→TPC-H→Schema Build.
4. Double-click Options.
5. Choose 300 scale, and the number of virtual users corresponding to the number of CPU cores. Click OK.
6. For the 14G dataset, check the box for Clustered Columnstore.
7. Double-click Build. This build could take several hours to complete.
8. Using SQL Server Management Studio, perform a full backup of the database for restoring between test runs.

Performing the test on Windows

Using the legacy server as our base system, we ran the TPC-H-like test while increasing the DOP by one after each run until the time to finish increased. We determined that a DOP of 6 was the ideal performance point on the legacy system, and MAXDOP of 8 was the ideal for the 14G solution.

1. Start HammerDB.
2. Set the database server to SQL Server, and the workload to TPC-H.
3. Open the Options panel for the Schema Build: SQL Server→TPC-H→Schema Build→Options.
 - a. Change the ODBC Driver to ODBC Driver 10.0 for SQL Server.
 - b. Change the MAXDOP to 6.
 - c. Select 300 as the Scale Factor.
 - d. Click OK.
4. Open the Options panel for the Virtual Users: SQL Server→TPC-H→Virtual User→Options.
 - a. Use one Virtual User.
 - b. Select the following:
 - i. Show Output
 - ii. Log Output to Temp
 - iii. Use Unique Log Name
 - c. Click OK.
5. Open the Options panel for the Driver: SQL Server→TPC-H→Driver Script→Options.
 - a. Set Total Query Sets per User to 1
 - b. Unselect "Verbose Output" and "Refresh Function"
 - c. Click OK.
6. Load the Driver Script: SQL Server→TPC-H→Driver Script→Load.
7. Start a custom Perfmon data collector set to capture CPU, RAM, and disk performance counters.
8. Click the green arrow.
9. Note the name of the log file. To start the run, click OK.
10. When the run finishes, stop Perfmon.
11. Save the HammerDB results text file and Perfmon output.
12. Open SQL Manager, delete the TPC-H database, and restore the database from the backup file.
13. Reboot the server.
14. Repeat steps 1 through 13 twice more to account for variance.

Performing the test on Linux

Using the same TPC-H-like test methodology as the 12th generation solution, we determined that MAXDOP of 8 was the ideal for the 14th generation solution.

1. Log onto the Linux server via a VNC client or with X11 forwarding enabled.
2. Before each run, delete the old data, and load the TPC-H-like data from the backup file.

```
sqlcmd -S localhost -U SA -P Passw0rd@ -I ~/Restore_TCPH.sql

cat ~/Restore_TCPH.sql
USE [master]
GO
IF EXISTS (SELECT name FROM sys.databases WHERE name = N'tpch')
DROP DATABASE [tpch]
GO
RESTORE DATABASE [tpch] FROM DISK = N'/opt/db/data/tpch_scale_300_columnstore.bak'
WITH REPLACE, FILE = 1,
MOVE N'tpch01' TO N'/opt/db/data/tpch01.mdf',
MOVE N'tpch02' TO N'/opt/db/data/tpch02.ndf',
MOVE N'tpch03' TO N'/opt/db/data/tpch03.ndf',
MOVE N'tpch04' TO N'/opt/db/data/tpch04.ndf',
MOVE N'tpch05' TO N'/opt/db/data/tpch05.ndf',
MOVE N'tpch06' TO N'/opt/db/data/tpch06.ndf',
MOVE N'tpch07' TO N'/opt/db/data/tpch07.ndf',
MOVE N'tpch08' TO N'/opt/db/data/tpch08.ndf',
MOVE N'tpch_log' TO N'/opt/db/logs/tpch_log.ldf',
NOUNLOAD, STATS = 10
GO
3. Clear the filesystem data cache.
echo 3 > /proc/sys/vm/drop_caches
```

4. Start the system performance gathering scripts.

```
bash sar_iostat.sh 750
```

```
cat sar_iostat.sh
#!/bin/bash
if [[ -z "$1" ]]; then
    echo "usage: $0 #_of_5s_steps"
fi
m=${HOSTNAME}_$(date "+%Y%m%d%H%M%S")
sar -o sar-out_${m}.sa 5 $1 >/dev/null 2>&1 &
S_TIME_FORMAT=ISO \
    iostat -dxmtty -p sdaa sda sdb sdc 5 $1 > iostat-out_${m}.txt 2>&1 &
```

5. Change directory to the location of Hammer DB.
6. Start the Hammer DB GUI client.

```
./hammerdb.tcl
```
7. Set the database server to SQL Server and the workload to TPC-H.
8. Open the Options panel for the Schema Build: SQL Server→TPC-H→Schema Build→Options.
 - a. Change the ODBC Driver to ODBC Driver 17 for SQL Server.
 - b. Change the password to `Passw0rd@.`
 - c. Change the MAXDOP to 8.
 - d. Select Clustered Column Store.
 - e. Select 300 as the Scale Factor.
 - f. Click OK.
9. Open the Options panel for the Virtual Users: SQL Server→TPC-H→Virtual User→Options.
 - a. Use ne Virtual User.
 - b. Select the following:
 - i. Show Output
 - ii. Log Output to Temp
 - iii. Use Unique Log Name
 - c. Click OK.
10. Open the Options panel for the Driver: SQL Server→TPC-H→Driver Script→Options.
 - a. Set Total Query Sets per User to 1.
 - b. Unselect Verbose Output and Refresh Function.
 - c. Click OK.
11. Load the Driver Script: SQL Server→TPC-H→Driver Script→Load.
12. Click the green arrow.
13. Note the name of the log file. To start the run, click OK.
14. At the end of the run, copy the HammerDB log file from /tmp to the home directory.

Read the report at <http://facts.pt/1s0pmi> ►

This project was commissioned by Dell Technologies.



Facts matter.®

Principled Technologies is a registered trademark of Principled Technologies, Inc.
All other product names are the trademarks of their respective owners.

DISCLAIMER OF WARRANTIES; LIMITATION OF LIABILITY:

Principled Technologies, Inc. has made reasonable efforts to ensure the accuracy and validity of its testing, however, Principled Technologies, Inc. specifically disclaims any warranty, expressed or implied, relating to the test results and analysis, their accuracy, completeness or quality, including any implied warranty of fitness for any particular purpose. All persons or entities relying on the results of any testing do so at their own risk, and agree that Principled Technologies, Inc., its employees and its subcontractors shall have no liability whatsoever from any claim of loss or damage on account of any alleged error or defect in any testing procedure or result.

In no event shall Principled Technologies, Inc. be liable for indirect, special, incidental, or consequential damages in connection with its testing, even if advised of the possibility of such damages. In no event shall Principled Technologies, Inc.'s liability, including for direct damages, exceed the amounts paid in connection with Principled Technologies, Inc.'s testing. Customer's sole and exclusive remedies are as set forth herein.