

Beyond the Report—Testing Addendum:

# Achieve Higher Performance with NVM Express® (NVMe®) RAID

This document provides the system configuration details and step-by-step procedures for the Prowess Consulting benchmark testing on the following Dell™ platforms:

- 1 x Dell™ PowerEdge™ R760 server with two Dell PowerEdge RAID Controller 12 (PERC 12) H965i Front controllers
- 1 x PowerEdge R760 server with two Dell PERC 11 H755N Front controllers

For the full analysis, read the [technical research report](#).

Testing was concluded on March 26, 2024.

## Server Configurations

	Dell™ PERC 12 H965i Front Configuration	Dell™ PERC 11 H755N Front Configuration
Server	Dell™ PowerEdge™ R760	Dell™ PowerEdge™ R760
Processor	Intel® Xeon® Platinum 8460Y+ processor	Intel® Xeon® Platinum 8460Y+ processor
Number of CPUs	2	2
Cores/Threads per CPU	40,80	40/80
Cores/Threads Total	80,160	80,160
Frequency (Base/SCT/MCT)	2 GHz (max 4.0 GHz)	2 GHz (max 4.0 GHz)
Storage Controller 01	Marvell Technology Group Ltd. Dell™ Boot-Optimized Server Storage (BOSS)-N1 Monolithic	Marvell Technology Group Ltd. Dell™ Boot-Optimized Server Storage (BOSS)-N1 Monolithic
Disk	960 GB Dell™ NVM Express® (NVMe®) PE8010 RI M.2, SK hynix	960 GB Dell™ NVM Express® (NVMe®) PE8010 RI M.2, SK hynix
Number of Disks	2	2
Storage Controller 02	Broadcom / LSI PERC H965i Front (firmware: 8.4.0.0.18-29)	Broadcom / LSI PERC H755N Front (Firmware: 52.21.1-5149)
Disk	Dell™ Enterprise NVMe CM6 MU 1.6TB (KIOXIA Corporation) (Firmware: 2.2.0)	Dell™ Enterprise NVMe CM6 MU 1.6TB (KIOXIA Corporation) (Firmware: 2.2.0)
Number of Disks	8	8
Storage Controller 03	Broadcom / LSI PERC H965i Front (firmware: 8.4.0.0.18-29)	Broadcom / LSI PERC H755N Front (Firmware: 52.21.1-5149)
Disk	Dell™ Enterprise NVMe® CM6 MU 1.6TB (KIOXIA Corporation) (firmware: 2.2.0)	Dell™ Enterprise NVMe® CM6 MU 1.6TB (KIOXIA Corporation) (firmware: 2.2.0)
Number of Disks	8	8
Installed Memory	256 GB	256 GB
Memory DIMM	16 x 16 GB hynix HMCG78MEBRA174N	16 x 16 GB hynix HMCG78MEBRA174N
Memory Speed	4,800 megatransfers per second (MT/s) Single-Rank	4,800 megatransfers per second (MT/s) Single-Rank
Number of Memory DIMMs	16	16
Network	Broadcom® BCM57414 NetXtreme® E-Series 10 Gb/25 Gb remote direct memory access (RDMA) Ethernet controller Broadcom® NetXtreme® E-Series P2100D BCM57508 2 x 100 Gb QSFP PCIe®	Broadcom® BCM57414 NetXtreme® E-Series 10 Gb/25 Gb remote direct memory access (RDMA) Ethernet controller Broadcom® NetXtreme® E-Series P2100D BCM57508 2 x 100 Gb QSFP PCIe®
Operating System (OS)	Red Hat® Enterprise Linux® 9.2 (Plow)	Red Hat® Enterprise Linux® 9.2 (Plow)

OS Kernel	Linux® 5.14.0-284.30.1.el9_2.x86_64	Linux® 5.14.0-284.30.1.el9_2.x86_64
Microsoft® SQL Server® Version	Microsoft® SQL Server® 2022 (RTM-CU10) (KB5031778) -16.0.4095.4 (X64)	Microsoft® SQL Server® 2022 (RTM-CU10) (KB5031778) -16.0.4095.4 (X64)
BIOS Version	1.5.6	1.5.6

## Testing Summary

Prowess Consulting engineers ran tests comparing the following two configurations:

- 1 x PowerEdge R760 server with two Dell PERC 12 H965i Front controllers
- 1 x PowerEdge R760 server with two Dell PERC 11 H755N Front controllers

We used Microsoft® SQL Server® to measure RAID rebuild and restore times. Our testing showed that RAID rebuild and restore times on a PowerEdge R760 server with PERC 12 and NVMe Express® (NVMe®) RAID drives were lower than RAID rebuild and restore times running on the same configuration with PERC 11 and NVMe RAID drives.

Additionally, we used Flexible I/O (fio), RocksDB, and dbench running on Red Hat® Enterprise Linux® 9.2 to benchmark storage.

- Flexible I/O (fio) is an open-source benchmarking tool used for measuring the performance of storage devices and filesystems.
- RocksDB is an open-source, embeddable, persistent key-value store optimized for fast storage.
- dbench is a benchmarking tool designed to evaluate the performance of file systems and disk I/O.

Our testing showed that fio, RocksDB, and dbench running on a PowerEdge R760 server with PERC 12 and NVMe RAID drives performed better than fio, RocksDB, and dbench running on the same configuration with PERC 11 and NVMe RAID drives.

## Testing Procedures

Our engineers completed the following procedures to test SQL Server on a PowerEdge R760 server using two PERC 12 H965i Front RAID controllers with NVMe storage and on a PowerEdge R760 server using two PERC 11 H755N Front RAID controllers with NVMe storage.

### Configuring Firmware and RAID for Dell™ PERC 12

We first configured firmware and RAID on the PowerEdge R760 server with Dell PERC 12.

1. Launch the server into the Integrated Dell™ Remote Access Controller (iDRAC).
2. Select **Install firmware updates**.
3. On the **Firmware update** page, select the **Dell website**, and then click **Next**.
4. On the **Firmware update: Launch the firmware update** page, click **Next**.
5. If prompted for **Proxy**, click **Yes**.
6. Select the **Updates** page, review the updates, and click **Apply**.
7. Reboot the server when directed.
8. Log in to iDRAC.
9. Select **Storage > Overview**.
10. Select the first RAID controller.
11. From the **Actions** drop-down menu, select **Create Virtual Disk**.
  - a. On the **Set up virtual disk** page, select or enter the following:
    - i. **Name: Data**
    - ii. **Controller: PERC H965i Front (Embedded)**
    - iii. **Layout: RAID-5**
    - iv. **Media Type: SSD**
    - v. **Physical Disk Selection: New Group**

- vi. **Security: Disabled**
        - vii. **Stripe Element Size: 64 KB**
        - viii. **Read Policy: No Read Ahead**
        - ix. **Write Policy: Force Write Back**
        - x. **Disk Cache Policy: Default**
      - b. Select the **Physical Disk** page, select eight disks, and click **Next**.
      - c. On the **Virtual Disk Settings** page, click **Next**.
      - d. On the **Confirmation** page, click **Add to Pending**.
      - e. Click **Apply Now**.
      - f. On the **Information** page, click **OK**.
12. Select the second RAID controller.
13. From the **Actions** drop-down menu, select **Create Virtual Disk**.
  - a. On the **Set up virtual disk** page, select or enter the following:
    - i. **Name: Logs**
    - ii. **Controller: PERC H965i Front (Embedded)**
    - iii. **Layout: RAID-10**
    - iv. **Media Type: SSD**
    - v. **Physical Disk Selection: New Group**
    - vi. **Security: Disabled**
    - vii. **Stripe Element Size: 64 KB**
    - viii. **Read Policy: No Read Ahead**
    - ix. **Write Policy: Force Write Back**
    - x. **Disk Cache Policy: Default**
14. Reboot the server, if needed, to begin the RAID creation job.
15. From iDRAC, select the virtual remote connection to the server.
16. Select **Virtual Media**.
17. On the **Map CD/DVD** page, select **Map device**.
18. In the **Location** field, select the **Red Hat Enterprise Linux ISO**, and then click **Map device**.
19. Click **Close**.
20. Click **Boot**, and then select **Virtual CD/DVD/ISO**.
21. Reboot the server.
22. When booted into the Red Hat Enterprise Linux installation media, select **Install Red Hat Enterprise Linux 9.2**.
23. From the **Welcome to Red Hat Enterprise Linux** page, leave the default settings and click **Continue**.
24. Select **Date and Time**, update to **Pacific time zone**, and then click **Done**.
25. Select the **Root password**, enter the same password twice, and then click **Done**.
26. Select **Software selection**, select **Server**, and then click **Done**.
27. Select **Installation Destination**.
28. Select the Dell™ Boot Optimized Server Storage (BOSS) disk location.
29. From **Storage Configuration**, click **Custom**.
  - a. Click **Done**.
  - b. Under **New Red Hat Enterprise Linux 9.2 installation**, click **Click here to create them automatically** to create the mount points.
  - c. Update the mount points as follows:
    - **/home: 100 GiB**
    - **wap: 16 GiB**
    - **/root: remaining disk space**
  - d. Click **Done**.
30. If prompted, click **Accept Changes**.

31. Click **Network and Host name**.
  - a. In the **Host Name** field, enter a name for the server.
  - b. From the network adapter list, enable connected network interface controllers (NICs), and then click **Done**.
32. Click **Connect to Red Hat**.
  - a. Enter account information to connect to Red Hat services, click **Register**, and then click **Done**.
33. Click **Begin Installation**.
34. When the installation is complete, click **Reboot system**.
35. Log in to the server.
36. Run the following command to get the IP:

```
ip a
```

37. Run the following command to install updates:

```
dnf update -y && dnf upgrade -y
```

38. Run the following commands to enable additional Red Hat Enterprise Linux repository access:

```
subscription-manager repos --enable codeready-builder-for-rhel-9-$(arch)-rpms  
dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm -y  
dnf install -y epel-release
```

39. Run the following command to install fio:

```
dnf install fio atop nmon tmux systat dstat -y
```

40. Run the following command to update the file limitations for the OS:

```
nano /etc/security/limits.d/99-mssql-server.conf
```

41. Copy the following and save the file:

```
mssql hard nfile 32727  
mssql soft nfile 16000
```

42. Run the following commands to set SQL Server parameters. Note: Set memory.memorylimitmb to 90 percent of available system memory:

```
mssql-conf traceflag 3979 on  
mssql-conf set control.writethrough 1  
mssql-conf set control.alternatewritethrough 0  
mssql-conf set telemetry.customerfeedback false  
mssql-conf set network.tlsprotocols 1.2  
mssql-conf set memory.memorylimitmb 235929
```

43. Run the following command to precondition the disks:

```
fio --ioengine=libaio --invalidate=1 --fsync_on_close=1 --direct=1 --output=steady.tmp --norandommap --  
numjobs=8 --name=/dev/sdb --filename=/dev/sdb --rw=write --bs=4k --iodepth=32 --randrepeat=0 --size=2000G  
fio --ioengine=libaio --invalidate=1 --fsync_on_close=1 --direct=1 --output=steady.tmp --norandommap --  
numjobs=8 --name=/dev/sdc --filename=/dev/sdc --rw=write --bs=4k --iodepth=32 --randrepeat=0 --size=2000G
```

44. Run the following command to install the SQL Server Red Hat Enterprise Linux Tuned profile:

```
dnf install tuned-profiles-mssql
```

45. Run the following command to set the SQL Server Tuned profile:

```
tuned-adm profile mssql
```

46. Run the following command to verify the SQL Server Tuned profile:

```
tuned-adm active
```

47. Run the following command to download the SQL Server 2022 Red Hat Enterprise Linux repository config file:

```
curl -o /etc/yum.repos.d/mssql-server.repo https://packages.microsoft.com/config/rhel/9/mssql-server-2022.repo
```

48. Run the following command to install SQL Server 2022:

```
dnf install -y mssql-server
```

49. Run the following command to configure SQL Server 2022:

- a. At the **Enter your edition** prompt, enter **1**, and then press **Enter**.
- b. At the **Do you accept the license terms** prompt, enter **Yes**, and then press **Enter**.
- c. At the **Enter the SQL Server system administrator password** prompt, enter a password, and then press **Enter**.
- d. At the **Confirm the SQL Server system administrator password** prompt, reenter the password, and then press **Enter**.

50. Run the following command to check the SQL Server status:

51. Run the following command to set mssql-conf into the PATH variable:

```
echo 'export PATH="$PATH:/opt/mssql/bin"' >> ~/.bash_profile
echo 'export PATH="$PATH:/opt/mssql/bin"' >> ~/.bashrc
source ~/.bashrc
```

52. Run the following command to stop the SQL Server service:

```
systemctl stop mssql-server
```

53. Run the following command to create directories for SQL Server database and log files:

```
mkdir -p /mssql/data /mssql/logs
```

54. Run the following command to retrieve the storage device names:

```
lsblk
```

55. Run the following commands to create a file system on the two storage devices:

```
mkfs.xfs -L logs /dev/sda -f; mkfs.xfs -L data /dev/sdb -f
```

56. Run the following command to mount the storage devices to the file directories:

```
mount /dev/sda /mssql/logs; mount /dev/sdb /mssql/data
```

57. Run the following command to add additional directories to support the backup and TempDB locations:

```
mkdir -p /mssql/data/backup /mssql/logs/tempdb /mssql/logs/logs
```

58. Run the following commands to set permissions on the SQL Server database and log file directories:

```
chown -R mssql:mssql /mssql
chmod 777 -R /mssql
```

59. Run the following commands to set the directory location for the SQL Server databases:

```
mssql-conf set filelocation.defaultdatadir /mssql/data; mssql-conf set filelocation.defaultlogdir /
mssql/logs; mssql-conf set filelocation.defaultbackupdir /mssql/data/backup
```

60. Run the following commands to install SQL Server tools:

```
curl https://packages.microsoft.com/config/rhel/9/prod.repo | sudo tee /etc/yum.repos.d/mssql-release.
repo
sudo yum install -y mssql-tools unixODBC-devel
echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bash_profile
```

61. Run the following command to enter SQL commands:

```
sqlcmd -S localhost -U sa
```

62. Run the following commands to set the TempDB database location:

```
ALTER DATABASE tempdb MODIFY FILE
(NAME = tempdev, FILENAME = '/mssql/logs/tempdb/tempdb.mdf', SIZE = 1024, FILEGROWTH = 8192MB)
GO
ALTER DATABASE tempdb MODIFY FILE
(NAME = templog, FILENAME = '/mssql/logs/tempdb/templog.ldf', SIZE = 1024, FILEGROWTH = 8192MB)
GO
ALTER DATABASE tempdb
ADD FILE (NAME = tempdb2, FILENAME = '/mssql/logs/tempdb/tempdb2.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
ALTER DATABASE tempdb
ADD FILE (NAME = tempdb3, FILENAME = '/mssql/logs/tempdb/tempdb3.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
```

```

ALTER DATABASE tempdb
ADD FILE (NAME = tempdb4, FILENAME = '/mssql/logs/tempdb/tempdb4.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
ALTER DATABASE tempdb
ADD FILE (NAME = tempdb5, FILENAME = '/mssql/logs/tempdb/tempdb5.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
ALTER DATABASE tempdb
ADD FILE (NAME = tempdb6, FILENAME = '/mssql/logs/tempdb/tempdb6.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
ALTER DATABASE tempdb
ADD FILE (NAME = tempdb7, FILENAME = '/mssql/logs/tempdb/tempdb7.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
ALTER DATABASE tempdb
ADD FILE (NAME = tempdb8, FILENAME = '/mssql/logs/tempdb/tempdb8.ndf', SIZE = 1024, FILEGROWTH =
8192MB);
GO

```

63. Run the following command to delete the old TempDB files:

```
rm -rf /var/opt/mssql/data/temp*
```

### Testing with fio, RocksDB, and dbench

The following outlines the fio, RocksDB, and dbench testing:

1. Run the following command to install Phoronix prerequisites:

```
dnf install php-cli php-xml php-json
```

2. Run the following command to install the dbench benchmark:

```
./phoronix-test-suite install-dependencies pts/dbench
./phoronix-test-suite install pts/dbench
```

3. Run the following command to run the dbench benchmark:

```
./phoronix-test-suite benchmark dbench
```

4. Use the following parameters:

- **Disk Test Configuration: 7**
- **Would you like to save these test results: Y**
- **Enter a name for the result file:** *Name*
- **Enter a unique name to describe this test run/configuration:** *Details*
- **New Description:** *Details*

5. Run the following command to install the RocksDB benchmark:

```
./phoronix-test-suite install-dependencies pts/rocksdb
./phoronix-test-suite install pts/rocksdb
```

6. Run the following command to run the RocksDB benchmark:

```
./phoronix-test-suite benchmark rocksdb
```

7. Use the following parameters:

- **Disk Test Configuration: Random Fill, Random Read, and Read Random Write Random**
- **Would you like to save these test results: Y**
- **Enter a name for the result file:** *Name*
- **Enter a unique name to describe this test run/configuration:** *Details*
- **New Description:** *Details*

8. Run fio with random read and random write with the following settings:

- **Random Test Worker Sizes: 8, 16, 32**
- **Random Tests IO Sizes: 4KiB, 8KiB, 64KiB, 128KiB**
- **Queue depths: 1, 4, 16, 32**

### Configuring Firmware and RAID for Dell PERC 11

After configuring the firmware and RAID on the PowerEdge R760 server with PERC 12, we then configured the firmware and RAID on the PowerEdge R760 server with PERC 11.

1. Use Secure Shell (SSH) to connect to the server.
2. Run the following command to stop and disable the SQL Server services:

```
systemctl stop mssql-server && systemctl disable mssql-server
```

3. Run the following command to copy the files from the paths where the SQL Server files are located to the OS disk:

```
cp -R /mssql/data /tmp/mssql
```

4. Log in to iDRAC.
5. Select **Storage > Overview**.
6. Select the first RAID controller.
7. From the **Actions** drop-down menu, select **Create Virtual Disk**.
  - a. On the **Set up virtual disk** page, select or enter the following:

- i. **Name: Data**
- ii. **Controller: PERC H755N Front (Embedded)**
- iii. **Layout: RAID-5**
- iv. **Media Type: SSD**
- v. **Physical Disk Selection: New Group**
- vi. **Security: Disabled**
- vii. **Stripe Element Size: 64 KB**
- viii. **Read Policy: No Read Ahead**
- ix. **Write Policy: Write Through**
- x. **Disk Cache Policy: Disabled**

- b. Select the **Physical Disk** page, select eight disks, and then click **Next**.
- c. On the **Virtual Disk Settings** page, click **Next**.
- d. On the **Confirmation** page, click **Add to Pending**.
- e. Click **Apply Now**.
- f. On the **Information** page, click **OK**.

8. Select the second RAID controller.
9. From the **Actions** drop-down menu, select **Create Virtual Disk**.
  - a. On the **Set up virtual disk** page, select or enter the following:

- i. **Name: Logs**
- ii. **Controller: PERC H755N Front (Embedded)**
- iii. **Layout: RAID-10**
- iv. **Media Type: SSD**
- v. **Physical Disk Selection: New Group**
- vi. **Security: Disabled**
- vii. **Stripe Element Size: 64 KB**
- viii. **Read Policy: No Read Ahead**
- ix. **Write Policy: Write Through**
- x. **Disk Cache Policy: Disabled**

10. Reboot the server, if needed, to begin the RAID creation job.
11. Log in to the server utilizing SSH.
12. Run the following commands to precondition the disks:

```
fio --ioengine=libaio --invalidate=1 --fsync_on_close=1 --direct=1 --output=steady.tmp --norandommap --numjobs=8 --name=/dev/sdb --filename=/dev/sdb --rw=write --bs=4k --iodepth=32 --randrepeat=0 --size=2000G  
fio --ioengine=libaio --invalidate=1 --fsync_on_close=1 --direct=1 --output=steady.tmp --norandommap --numjobs=8 --name=/dev/sdc --filename=/dev/sdc --rw=write --bs=4k --iodepth=32 --randrepeat=0 --size=2000G
```

13. Run the following command to create a file system on the new volumes:

```
mkfs.xfs /dev/sda -f -L logvolume && mkfs.xfs /dev/sdb -f -L datavolume
```

14. Run the following command to mount the new file systems on the existing folder locations:

```
mount /dev/sda /mssql/log && mount /dev/sdb /mssql/data
```

15. Run the following command to recreate the working directories:

```
mkdir -p /mssql/data/backup /mssql/log/log /mssql/log/tempdb
```

16. Run the following commands to set permissions on the new directories:

```
chown -R mssql:mssql /mssql  
chmod -R 777 /mssql/
```

17. Run the following commands to enable and start SQL Server services:

```
systemctl enable mssql-server.service  
systemctl start mssql-server.service  
systemctl status mssql-server.service
```

### Testing with fio, RocksDB, and dbench

The following steps outline the fio, RocksDB, and dbench testing:

1. Run the following command to install Phoronix prerequisites:

```
dnf install php-cli php-xml php-json
```

2. Run the following command to install the dbench benchmark:

```
./phoronix-test-suite install-dependencies pts/dbench  
./phoronix-test-suite install pts/dbench
```

3. Run the following command to run the dbench benchmark:

```
./phoronix-test-suite benchmark dbench
```

4. Use the following parameters:

- **Disk Test Configuration: 7**
- **Would you like to save these test results: Y**
- **Enter a name for the result file:** *Name*
- **Enter a unique name to describe this test run/configuration:** *Details*
- **New Description:** *Details*

5. Run the following command to install the RocksDB benchmark:

```
./phoronix-test-suite install-dependencies pts/rocksdb  
./phoronix-test-suite install pts/rocksdb
```

6. Run the following command to run the RocksDB benchmark:

```
./phoronix-test-suite benchmark rocksdb
```

7. Use the following parameters:

- **Disk Test Configuration: Random Fill, Random Read, and Read Random Write Random**
- **Would you like to save these test results: Y**
- **Enter a name for the result file:** *Name*
- **Enter a unique name to describe this test run/configuration:** *Details*
- **New Description:** *Details*

8. Run fio with random read and random write with the following settings:

- **Random Test Worker Sizes: 8, 16, 32**
- **Random Tests IO Sizes: 4KiB, 8KiB, 64KiB, 128KiB**
- **Queue depths: 1, 4, 16, 32**

### Test Database Build

We used the following steps to build the test database to capture restore times:

1. Configure an Open Database Connectivity (ODBC) data source for SQL Server:
  - a. Launch ODBC Data Source Administrator (32-bit).
  - b. Click **Add**.



- c. Select **SQL Server Native Client 11.0**, and then click **Finish**.
  - d. In the **Create a New Data Source to SQL Server** window, in the **Name** field, enter a unique name to identify the SQL Server database.
  - e. In the **Server** field, enter a name or IP address to connect to SQL Server, and then click **Next**.
  - f. In the **How should SQL Server verify the authenticity of the login ID** field, select **With SQL Server authentication**.
  - g. In the **Login ID** field, enter **SA**.
  - h. In the **Password** field, enter the SA password.
  - i. Click **Next**.
  - j. Click **Next** again.
  - k. Click **Finish**.
  - l. Click **Test Data Source** to verify connectivity.
2. Download and install the following SQL Server client tools:
  - a. Client Tools Connectivity
  - b. Client Tools Backwards Compatibility
  - c. Client Tools SDK
3. Copy the provided **MSTPCC.470.tar** file to the test client.
4. Unzip the .tar file to **C:\MSTPCC.470**.
5. In File Explorer, browse to **C:\MSTPCC.470\SETUP\scripts\1400.war\database**.
6. Open **createdb.sql**, and then update the **FILENAME** path for each file.
7. In File Explorer, browse to **C:\MSTPCC.470\BC-TPCC-v1.6.0**.
8. Double-click **BenchCraft-TPCC.msi** to install BenchCraft, accepting the default settings.
9. Open a command prompt window or PowerShell® session.
10. Run the following command to build a TPC-C® database:

```
setup.cmd /S:<local host ip> /U:sa /P:<password> /W:1400 /B:bulddb /D:normal /V:true  
setup.cmd /S:<local host ip> /U:sa /P:<password> /W:1400 /B:objects /D:normal /V:true  
setup.cmd /S:<local host ip> /U:sa /P:<password> /W:1400 /B:bulkload /D:normal /V:true
```
11. Once the database build is complete, take a backup of the TPC-C database.
12. Launch the BenchCraft application:
  - a. Click **New Profile**.
  - b. Click **Browse**, and then select the **TPCC\_SQLServer.bdll** plugin file.
  - c. Click **Next**.
  - d. Select **ThreadGroup-1**, and then enter the following values for each setting:
    - i. **Thread Count: 100**
    - ii. **W\_ID Range End: 1400**
    - iii. **W\_ID Maximum: 1400**
    - iv. **Server Name or IP:** *IP address or name of the Linux SQL Server*
    - v. **User Name: SA**
    - vi. **Password:** *SA password*
13. Save the profile.
14. Select the **Drivers** page.
15. Select **Launch**.
16. Wait 5 minutes before pressing **F1** to run the load.
17. Run the load for 45 minutes or until the load is at a steady state, and then take the last 15 minutes of the load as the scored value.
18. Once the test is complete, delete and restore the TPC-C database.
19. Double-click the saved profile to launch the BenchCraft workload utility to run the workload again.

## Testing Under RAID Rebuild

We used the following steps to run the BenchCraft test while the RAID is under rebuild:

1. For PERC 12 testing:
  - a. During the RAID rebuild, perform the following tasks:
    - i. Set the disk as offline.
    - ii. Set the disk as online and rebuilding.
    - iii. Verify the status of the rebuild.
    - iv. Run the Microsoft BenchCraft benchmark tool.
    - v. For each RAID configuration, RAID 5 and RAID 10, update the controller and disk identifiers in the PERCCLI2 command:

```
/opt/MegaRAID/perccli/perccli2 /c1/e304/s7 set offline  
/opt/MegaRAID/perccli/perccli2 /c1/e304/s7 start rebuild  
/opt/MegaRAID/perccli/perccli2 /c1/e304/s7 show rebuild
```

2. For PERC 11 testing:
  - a. During the RAID rebuild, perform the following tasks using the instructions below:
    - i. Set the disk offline.
    - ii. Set the disk as online and rebuilding.
    - iii. Verify the status of the rebuild.
    - iv. Run the Microsoft BenchCraft benchmark tool.
    - v. For each RAID configuration, RAID 5 and RAID 10, update the controller and disk identifiers in the PERCCLI64 command:

```
/opt/MegaRAID/perccli/perccli64 /c1/e252/s7 set offline  
/opt/MegaRAID/perccli/perccli64 /c1/e252/s7 start rebuild  
/opt/MegaRAID/perccli/perccli64 /c1/e252/s7 show rebuild
```

## Measuring Time to Restore the Test Database

This section covers the steps our engineers used to capture the time to restore the test database.

1. Run the following command to create the restore\_sqlbackup.sh bash script:

```
nano /tmp/restore_sqlbackup.sh
```

2. Copy the following content into the script:

```
#!/bin/bash  
ts=$(date +%s)  
echo $ts >> /testlogs/sqlbackup.txt  
sqlcmd -S localhost -U SA -P TestPassword1 -C -Q "RESTORE DATABASE [tpcc] FROM DISK = N'/mssql/data/  
backup/tpcc_backup.bak' WITH FILE = 1, NOUNLOAD, REPLACE, STATS = 1" >> /testlogs/sqlbackup.txt  
ts1=$(date +%s)  
echo $ts1 >> /testlogs/sqlbackup.txt
```

3. Save the script.
4. Run the following command to capture the time to restore the test database:

```
./restore_sqlbackup.sh
```

5. Subtract the end time from the start time to determine the time to restore the database backup.



The analysis in this document was done by Prowess Consulting and commissioned by Dell Technologies.

Results have been simulated and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance.

Prowess Consulting and the Prowess logo are trademarks of Prowess Consulting, LLC.

Copyright © 2024 Prowess Consulting, LLC. All rights reserved.

Other trademarks are the property of their respective owners.