



The science behind the report:

Finish Hadoop workloads faster and with higher throughput with Dell PowerEdge R750 servers featuring Dell PowerEdge RAID Controllers (PERC 11)

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 [Finish Hadoop workloads faster and with higher throughput with Dell PowerEdge R750 servers featuring Dell PowerEdge RAID Controllers \(PERC 11\)](#).

We concluded our hands-on testing on December 14, 2022. During testing, we determined the appropriate hardware and software configurations and applied updates as they became available. The results in this report reflect configurations that we finalized on December 12, 2022 or earlier. Unavoidably, these configurations may not represent the latest versions available when this report appears.

Our results

To learn more about how we have calculated the wins in this report, go to <http://facts.pt/calculating-and-highlighting-wins>. Unless we state otherwise, we have followed the rules and principles we outline in that document.

Table 1: Results of our Hadoop testing for both platforms. Source: Principled Technologies.

	Dell™ PowerEdge™ R750	HPE ProLiant DL380 Gen9	% win
Median time (seconds)	253	347	27.0%
Median throughput (GB/s)	0.71	0.52	36.5%

System configuration information

Table 2: Detailed information on the systems we tested.

System configuration information	Dell PowerEdge R750	HPE ProLiant DL380 Gen9
BIOS name and version	1.8.2	P89 v2.80
Operating system name and version/ build number	VMware® ESXi™, 7.0.3, 20036589	VMware ESXi, 7.0.3, 19482537
Date of last OS updates/patches applied	11/17/2022	12/12/2022
Power management policy	Maximum Performance	Static High Performance Mode
Processor		
Number of processors	2	2
Vendor and model	Intel® Xeon® Gold 6348	Intel Xeon E5-2650 v4
Core count (per processor)	28	12
Core frequency (GHz)	2.60	2.20
Stepping	1	1
Memory module(s)		
Total memory in system (GB)	256	192
Number of memory modules	16	12
Vendor and model	Hynix HMA82GR7DJR8N-XN	N/A
Size (GB)	16	16
Type	PC4-25600	DIMM DDR4
Speed (MHz)	3,200	2,400
Speed running in the server (MHz)	3,200	2,400
Storage controller		
Vendor and model	PERC H755N Front	Smart Array P440ar
Cache size (GB)	8	2
Firmware version	52.21.0-4606	7.20
Driver version	7.718.02.00	N/A
Local storage (HDFS)		
Number of drives	6	6
Drive vendor and model	Dell Ent NVMe v2 AGN MU U.2 1.6TB	Toshiba PX05SVB096Y
Drive size (GB)	1.6	960
Drive information (speed, interface, type)	NVMe 1.3, SAS, SSD, 16GT/s	SAS, SSD, 12Gbps
Firmware version	N/A	7.20

System configuration information	Dell PowerEdge R750	HPE ProLiant DL380 Gen9
Local storage (OS)		
Number of drives	2	2
Drive vendor and model	Micron MTFDDAV480TDS	Toshiba PX05SVB096Y
Drive size (GB)	480	960
Drive information (speed, interface, type)	SATA, SSD, 6Gbps	SAS, SSD, 12Gbps
Network adapter #1		
Vendor and model	Broadcom® BCM5720	HP Ethernet 1Gb 4-port 331i Adapter
Number and type of ports	2 x 1GbE	4 x 1GbE
Driver version	22.00.6	20.18.31
Network adapter #2		
Vendor and model	Broadcom NetXtreme OCP 3.0	N/A
Number and type of ports	2 x 25GbE	N/A
Driver version	22.21.06.80	N/A
Cooling fans		
Number of cooling fans	6	6
Power supplies		
Vendor and model	Dell PWR SPLY,1400W, RDNT, DELTA	HP 720479-B21
Number of power supplies	2	2
Wattage of each (W)	1,400	800

How we tested

Testing overview

We tested two servers, one Dell PowerEdge R750 with PERC 11 and one HPE ProLiant DL380 Gen9 with P440ar. We set up and configured the Dell PowerEdge R750 server remotely in a Dell lab; we installed and configured the HPE ProLiant DL380 Gen9 in the PT lab. We installed and configured the latest HPE-customized VMware vSphere 7.3 on the DL380 server. We installed the hypervisor on two SSD drives and attached six SSD drives the respective RAID controller on each server. We created a RAID5 logical drive on these six SSD drives. We deployed a Hadoop cluster with one manager node and four workers on each server. We configured the RAID5 logical drive as storage for Hadoop Distributed Filesystem (HDFS). We ran the TeraSort workload within the HiBench suite and measured run-time and throughput, and report the median of three runs.

The following sections describe the steps we took to configure the test environment and run the test.

Installing vSphere 7.0 on a HPE ProLiant DL380 Gen9 server

1. Open a new browser tab, and connect to the IP address of the HPE server iLO.
2. Log in with the iLO credentials.
3. On the left panel, click Remote Console.
4. Click HTML5 console to launch remote console.
5. In the console menu bar, select Virtual Media, and then Virtual Storage.
6. In the top menu, click CD/DVD, and select local .iso file.
7. Select the HPE custom ESXi 7.0 iso file from local computer.
8. In the top menu, select Power, and select Cold Boot.
9. The system will boot to the mounted image and the Loading ESXi installer screen will appear. When prompted, press Enter to continue.
10. To accept and continue, press F11.
11. Under Storage Device, select the installation drive, and press Enter to continue.
12. For keyboard layout, select US Default, and press Enter to continue.
13. Enter the root password twice, and press Enter to continue.
14. At the Confirm Install window, press F11 to install.
15. At the Installation Complete window, press Enter to reboot.
16. After reboot, press F2 to configure system.
17. Log in with root user/password and press Enter for OK.
18. Scroll to Configure Management Network, and press Enter.
19. Scroll to IPv4 Configuration, and press Enter.
20. Scroll to Static IPv4, and use spacebar to select it.
21. Set the IPv4 address, Subnet Mask, and Default gateway.
22. To continue, press Enter for OK.
23. Scroll to IPv6 Configuration, and press Enter.
24. Scroll to Disable IPv6, and use spacebar to select it.
25. To continue, press Enter for OK.
26. Scroll to DNS Configuration, and press Enter.
27. Scroll to manually configure DNS, and use spacebar to select it.
28. Add Primary DNS Server, Alternate DNS Server, and provide the hostname for the system.
29. Scroll to Custom DNS Suffixes, and press Enter.
30. Add the suffix that is required for testing, and press Enter for OK.
31. To accept the changes, press Esc.
32. To confirm changes, press Y. The system will reboot.
33. After reboot, press F2 to configure system.
34. Log in with root user/password, and press Enter for OK.
35. Scroll to Troubleshooting Options, and press Enter.
36. Select Enable ESXi Shell, and press Enter to enable it.
37. Select Enable SSH, and press Enter to enable it.
38. Scroll to Restart Management Agents, and press Enter.
39. To confirm, press F11 for OK.
40. To exit, press Esc.
41. To log out, press Esc.

Installing vCenter Server Appliance 7.0

1. Download VMware vCenter 7.0 Update 3 from the VMware support portal <https://my.vmware.com>.
2. Mount the image on your local system, and browse to the `vcsa-ui-installer` folder. Expand the folder for your OS, and launch the installer if it doesn't automatically begin.
3. When the vCenter Server Installer wizard opens, click Install.
4. To begin installation of the new vCenter server appliance, click Next.
5. Check the box to accept the license agreement, and click Next.
6. Enter the IP address of the infrastructure server with ESXi 7.0 Update 3. Provide the root password, and click Next.
7. To accept the SHA1 thumbprint of the server's certificate, click Yes.
8. Accept the VM name, and provide and confirm the root password for the VCSA. Click Next.
9. Set the size for environment you're planning to deploy. We selected Medium. Click Next.
10. Select the datastore on which to install vCenter. Accept the datastore defaults, and click Next.
11. Enter the FQDN, IP address information, and DNS servers you want to use for the vCenter server appliance. Click Next.
12. To begin deployment, click Finish.
13. When Stage 1 has completed, click Close. To confirm, click Yes.
14. Open a browser window, and connect to `https://[vcenter.FQDN]:5480/`.
15. On the Getting Started - vCenter Server page, click Set up.
16. Enter the root password, and click Log in.
17. Click Next.
18. Enable SSH access, and click Next.
19. To confirm the changes, click OK.
20. Enter `vsphere.local` for the Single Sign-On domain name. Enter a password for the administrator account, confirm it, and click Next.
21. Click Next.
22. Click Finish.

Creating a vSphere cluster in vCenter

1. Open a browser, and enter the address of the vCenter server you deployed. For example: `https://[vcenter.FQDN]/ui`
2. In the left panel, select the vCenter server, right-click, and select New Datacenter.
3. Provide a name for the new data center, and click OK.
4. Select the data center you just created, right-click, and select New Cluster.
5. Give a name to the cluster. Click OK.
6. In the cluster configuration panel, under Add hosts, click Add.
7. Enter the IP address and root credentials for the HPE DL380 server. Click Next.
8. Click Next.
9. Click Finish.

Creating a datastore on the disk array

1. Open a browser, and enter the address of the vCenter server you deployed. For example: `https://[vcenter.FQDN]/ui`
2. In the left panel, right-click the DL380 server, select Storage, and click New Datastore.
3. Select VMFS, and click Next.
4. Assign a name to the new datastore, select the RAID5 logical drive, and click Next.
5. Click Finish.
6. Repeat steps 1 through 5 to create a new datastore on the remote R750 server.

The following sections describe the steps we took to configure the Hadoop environment on each of the servers under test.

Installing and configuring a base CentOS VM

1. Log into vCenter, and from the Menu drop-down menu, click Storage.
2. Select the RAID 5 datastore we created on the server under test, and click Files.
3. Click Upload Files, and upload the CentOS stream 8 ISO image.
4. Right-click the server under test, and click New Virtual Machine.
5. Click Next.

6. Enter a name for the VM, and click Next.
7. Click Next.
8. Select the RAID5 Datastore, and click Next.
9. Click Next.
10. From the Guest OS Family drop-down menu, select Linux.
11. From the guest OS version drop-down menu, select CentOS 8 (64 bit), and click Next.
12. Assign the VM 2 vCPUs, 8 GB of memory, and a 512GB hard disk.
13. From the New CD/DVD Drive dropdown, select Datastore ISO File, and select the CentOS ISO you uploaded to the datastore previously. Ensure Connect At Power On is checked, and click Next.
14. Click Finish.
15. Power on the VM, and click Launch Remote Console.
16. Select Install CentOS Linux 8, and press Enter.
17. Select English as the language, and click Continue.
18. On the Installation summary screen, click Network & Host Name.
19. Turn on the switch for the ethernet port, and click Done.
20. Click Root Password, and set password for root user.
21. Click Installation Source, select `http://` from the dropdown menu, and enter `mirror.centos.org/centos/8-stream/BaseOS/x86_64/os/`.
22. Click Done.
23. Click Software Selection, select Server with GUI, and click Done.
24. Click Installation Destination, and select the disk for OS.
25. Click Begin Installation.
26. When installation is complete, click Reboot System.

Configuring CentOS 8 and installing Apache Hadoop and Spark

1. Log into the base CentOS VM via ssh as user root.
2. Modify SSH to allow a pre-shared key login:

```
mkdir -p /root/.ssh
chmod 700 /root/.ssh
cd /root/.ssh
ssh-keygen -t rsa -q
cp id_rsa.pub authorized_keys
echo "StrictHostKeyChecking=no" > config
```

3. Set the hostname by typing the following command:

```
hostnamectl set-hostname [HOSTNAME]
```

4. Modify your hosts file to add your hostname to your IP address.
5. Turn off and disable your firewall by typing the following commands:

```
systemctl stop firewalld
systemctl disable firewalld
```

6. Edit your selinux to disable its enforcing by typing the following commands:

```
setenforce 0
vi /etc/selinux/config (modify "enforcing" to "disabled" in the file)
```

7. Update your OS by typing the following command:

```
yum upgrade -y
```

8. Install the prerequisites via yum by typing the following command:

```
yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
yum install -y nmon python3 vim tar wget java-1.8.0-openjdk maven git blas64 lapack64
python2 bc curl
```

9. Download Hadoop and Spark by typing the following commands:

```
wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.4/hadoop-3.3.4.tar.gz
wget https://dlcdn.apache.org/spark/spark-3.3.1/spark-3.3.1-bin-hadoop3.tgz
```

10. Modify your bash profile ~/.bashrc and add the following lines:

```
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.322.b06-11.el8.x86_64/jre
PATH=$PATH:$HOME/bin:/opt/yarn/hadoop-3.3.4/bin
HADOOP_HOME=/opt/yarn/hadoop-3.3.4
```

11. Add in the Hadoop users by typing the following commands:

```
groupadd hadoop
useradd -g hadoop yarn
useradd -g hadoop hdfs
useradd -g hadoop mapred
```

12. Create default Hadoop directories, and set their permissions by typing the following commands:

```
mkdir -p /var/data/hadoop/hdfs/nn
mkdir -p /var/data/hadoop/hdfs/snn
mkdir -p /var/data/hadoop/hdfs/dn
chown hdfs:hadoop /var/data/hadoop/hdfs/ -R
mkdir -p /var/data/hadoop/hdfs/tmp
mkdir -p /var/log/hadoop/yarn
chown yarn:hadoop /var/log/hadoop/yarn/ -R
mkdir -p /opt/yarn
```

13. Extract the Hadoop and Spark compressed files by typing the following commands:

```
cd /opt/yarn
tar xvzf /root/hadoop-3.3.4.tar.gz
tar -xvzf ~/spark-3.3.1-bin-hadoop3.tgz
```

14. Move into the Hadoop directory and make a yarn directory by typing the following commands:

```
cd hadoop-3.3.4/
mkdir logs
chmod g+w logs
chown yarn:hadoop . -R
```

15. Append the following line to /opt/yarn/spark-3.3.1-bin-hadoop3/conf/spark-env.sh on each VM:

```
export SPARK_LOCAL_DIRS=/var/data/hadoop/hdfs/tmp
```

16. Navigate into the Hadoop configuration directory by typing the following command:

```
cd etc/hadoop/
```

17. Modify the Hadoop configuration files with the following settings:

core-site.xml

```
<configuration>
  <property>
    <name>fs.default.name</name>
    <value>hdfs://[MANAGER IP ADDRESS]:9000</value>
  </property>
  <property>
    <name>hadoop.http.staticuser.user</name>
    <value>hdfs</value>
  </property>
</configuration>
```

hdfs-site.xml

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>3</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/var/data/hadoop/hdfs/nn</value>
  </property>
  <property>
    <name>fs.checkpoint.dir</name>
    <value>file:/var/data/hadoop/hdfs/snn</value>
  </property>
  <property>
    <name>fs.checkpoint.edits.dir</name>
    <value>file:/var/data/hadoop/hdfs/snn</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/var/data/hadoop/hdfs/dn</value>
  </property>
</configuration>
```

mapred-site.xml

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>yarn.app.mapreduce.am.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME</value>
  </property>
  <property>
    <name>mapreduce.map.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME</value>
  </property>
  <property>
    <name>mapreduce.reduce.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME</value>
  </property>
</configuration>
```

yarn-site.xml

```
<configuration>
  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>[MANAGER_HOSTNAME_HERE]</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
  </property>
</configuration>
```

hadoop-env.sh

Uncomment the JAVA_HOME line and add the following information:

```
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.322.b06-11.el8.x86_64/jre
```

18. Power off the instance.

Creating Manager and Worker nodes for Hadoop and Spark

1. Log into vCenter, and right-click the base CentOS VM.
2. Select Clone and Clone to Virtual Machine.
3. Enter a name for the virtual machine, and click Next.
4. Select a host for the virtual machine, and click Next.
5. Select the RAID5 Datastore, and click Next.
6. Select Customize this virtual machine's hardware, and click Next.
7. For the Manager VM, select 2 vCPUs and 8 GB memory, and click Next.
8. For Worker VMs, select 8 vCPUs and 32GB memory. Click Next.
9. Click Finish.
10. Power on the virtual machine when clone is done.

Configuring and starting the Hadoop cluster

1. Set the hostname on the master and each of the worker nodes:

```
hostnamectl set-hostname <HOST_NAME>
```

2. Add the FQDN, hostname, and IP address of the manager VM and all worker VMs to the /etc/hosts file on the manager and all worker nodes.
3. Verify that you can do passwordless SSH into each VM.
4. On the Manager VM, format the hdfs filesystem:

```
hdfs namenode -format
```

5. Start the Hadoop services and Spark on the manager node:

```
/opt/yarn/hadoop-3.3.4/bin/hdfs --daemon start namenode
/opt/yarn/hadoop-3.3.4/bin/hdfs --daemon start secondarynamenode
/opt/yarn/hadoop-3.3.4/bin/yarn --daemon start resourcemanager
/opt/yarn/hadoop-3.3.4/bin/yarn --daemon start nodemanager
/opt/yarn/spark-3.3.1-bin-hadoop3/sbin/start-master.sh
```

- Start the Hadoop services and Spark on each of the worker nodes:

```
/opt/yarn/hadoop-3.3.4/bin/hdfs --daemon start datanode
/opt/yarn/spark-3.3.1-bin-hadoop3/sbin/start-slave.sh spark://[MANAGER IP ADDRESS]:7077
```

- Verify HDFS is up and running with all four data nodes:

```
hdfs dfsadmin -report
```

- Verify Spark is up and running by going to the remote console of the Manager VM. Open a web browser and type in address localhost:8080. Verify all four worker nodes have joined the Workers list.
- Perform the following steps on the manager node to install and configure HiBench.

- Create the directories you will use for HiBench by typing the following commands:

```
hdfs dfs -mkdir -p /user/root
hdfs dfs -mkdir /HiBench
hdfs dfs -chown -R root:hadoop /HiBench
hdfs dfs -chown root /user/root
```

- Navigate to your home directory and download HiBench by typing the following commands:

```
cd ~
git clone https://github.com/intel-hadoop/HiBench.git
```

- Install HiBench for Spark 3.0 by typing the following commands:

```
cd HiBench/
mvn -Dspark=3.0 -Dscala=2.12 clean package | tee hibench_build.log
cd conf/
```

- Modify the HiBench configuration files with the following information:

hadoop.conf

```
# Hadoop home
hibench.hadoop.home      /opt/yarn/hadoop-3.3.4

# The path of hadoop executable
hibench.hadoop.executable ${hibench.hadoop.home}/bin/hadoop

# Hadoop configuration directory
hibench.hadoop.configure.dir ${hibench.hadoop.home}/etc/hadoop

# The root HDFS path to store HiBench data
hibench.hdfs.master      hdfs://[MANAGER IP ADDRESS]:9000

# Hadoop release provider. Supported value: apache, cdh5, hdp
hibench.hadoop.release   apache
```

spark.conf

```
# Spark home
hibench.spark.home      /opt/yarn/spark-3.3.1-bin-hadoop3/

# Spark master
# standalone mode: spark://xxx:7077
# YARN mode: yarn-client
hibench.spark.master    spark://[MANAGER IP ADDRESS]:7077
```

- Modify /root/HiBench/conf/hibench.conf to set the data scale profile to huge:

```
hibench.scale.profile      huge
```

Running the TeraSort tests

In this section, we list the steps to run the TeraSort benchmark on the Hadoop cluster under test.

1. Log into the manager node via SSH.
2. Navigate to the HiBench prepare directory for TeraSort workload, and generate the input dataset:

```
cd /root/HiBench/bin/workloads/micro/terasort/prepare
./prepare.sh
```

3. Navigate to the Spark directory:

```
cd /root/HiBench/bin/workloads/micro/terasort/spark
```

4. Clear the cache on the manager and worker nodes before running a test:

```
sync; echo 3 > /proc/sys/vm/drop_caches
```

5. Run the benchmark:

```
./run_test.sh 2>&1 | tee <test_log>
```

6. Run the same test three times on each server.

Read the report at <https://facts.pt/AbwL4Nf> ▶

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.