DELL POWEREDGE C5220: HADOOP MAPREDUCE PERFORMANCE

Dell™ PowerEdge™ C5220 delivered excellent performance

with Hadoop MapReduce

Every year, the amount of data that businesses must process grows enormously. The ability to sort, filter, and analyze this data is becoming more and more vital to many businesses in analyzing their customers and their market segment. Additionally, businesses need an infrastructure that is powerful and flexible, but also compact and scale-friendly. The Dell PowerEdge C5220 server is an ideal solution to pair with Apache Hadoop, a powerful multi-node data analysis application. With the PowerEdge C5220, organizations can scale out to their data processing requirements and successfully handle these ever-increasing data volumes, finding new value in their big data.

To test the Hadoop performance capabilities of the Dell PowerEdge C5220, we configured eight Dell PowerEdge C5220 servers into a Hadoop cluster and ran the MapReduce benchmark (mrbench) on the platform. We found that eight Dell PowerEdge C5220 servers, all contained within the single shared infrastructure design of the Dell PowerEdge C5000 chassis, could run our mrbench tests of varying sizes, map processes, and reduce processes, in times averaging just 15.9 to 25.6 seconds, making this platform ideal for scale-out data-analysis application workloads.

A PRINCIPLED TECHNOLOGIES TEST REPORT

DELL POWEREDGE C5220 AND HADOOP – BETTER TOGETHER

Today's marketplace has moved beyond the structural confines of relational data and up to the category of "big data." Companies with hyperscale data centers are dealing with tremendous amounts of existing data, with massive data growth on the horizon. They require a reliable and powerful hardware platform, along with a reliable and fast software platform on which to process this big data. The Dell PowerEdge C series servers provide this solid infrastructure for companies to offer their data processing capabilities. Integrating many PowerEdge C servers together in a Hadoop cluster allows these companies to quickly and efficiently process large datasets.

In our lab tests, we configured eight Dell PowerEdge C5220 microservers into a Hadoop cluster and ran the MapReduce benchmark on the cluster to demonstrate the capabilities of the Dell PowerEdge C platform. Below we discuss the Dell PowerEdge C5220 microserver, our test topology, and our results in more detail.

Figure 1: The Dell PowerEdge C5220 microserver provides power and reliability in a tight and dense form factor, within the C5000 shared infrastructure.

Dell PowerEdge C5220

The Dell PowerEdge C5220 microserver is a highly dense small-form-factor server mounted on a sled, which is housed in a Dell PowerEdge C5000 shared infrastructure enclosure, offering up to 12 microservers in a 3U rack density. These high physical densities offer massive scale-out configurations, and are ideal for service providers and large data centers that need optimized space, power, and cooling infrastructure. Each server, all cold-aisle accessible for maintenance, features processing power from the Intel® Xeon® E3-1200 series, up to 32GB RAM

Highlights of the Dell PowerEdge C5220

- Up to 12 PowerEdge C5220 microservers in a single 3U C5000 chassis
- Cold aisle serviceability
- Shared infrastructure uses less floor space, power, and cooling
- Dense form factor ideal for service providers, hosting platforms, and hyperscale environments
- Up to 6TB raw storage in a single microserverEmbedded chassis management control

capacity, and up to four drives per server. The high performance levels it delivers, coupled with a simple, modular, and easy-to-maintain design, make the PowerEdge C5220 microserver an optimal choice to deploy for extremely dense compute fabrics handling big data deployments, large software as a service (SaaS) environments, and cloud deployments. Figure 2 presents a view of the Dell PowerEdge C5220.
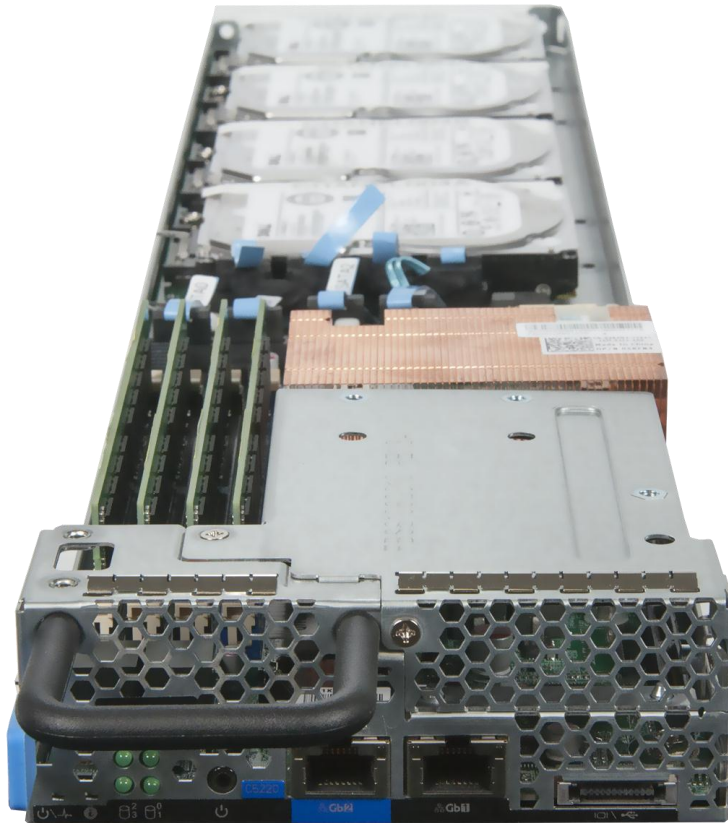

Figure 2: The PowerEdge C5220 microserver.

The Dell PowerEdge C5220 can deliver many benefits to your large cloud deployment, and lets you:

- Maintain your environment easily with cold-aisle accessible, swappable microserver modules and connections.
- Make better use of expensive data center space by increasing the rack density with up to twelve server nodes fitting into a standard 3U rack slot.
- Utilize up to 6 TB of raw disk storage per microserver with SAS drives or 4 TB of raw disk storage per microserver with SATA drives – up to 72 TB and 48 TB respectively – across the entire C5000 shared infrastructure in just a 3U form factor

- Reuse or repurpose servers easily when workloads change with hot-swap server nodes – you no longer need to experience downtime by replacing the entire server chassis.

Designed with power-efficiency and maintainability in, the Dell PowerEdge C5220 maximizes operating efficiency with a shared-infrastructure design. To learn more about the Dell PowerEdge C5220 and the entire Dell PowerEdge C Series, visit http://www.dell.com/us/enterprise/p/poweredge-cloud-servers.

WHAT WE TESTED

To test the ability of the PowerEdge C5220 microserver to handle large data processing tasks, we used Hadoop, specifically Cloudera Distribution Including Apache Hadoop (CDH). Below, we briefly discuss Hadoop and the benchmark tool we used, MapReduce benchmark (mrbench).

Hadoop

Hadoop, developed by Apache Software Foundation, is an open-source distributed application that enables the analysis of large volumes of data for specific purposes. Using Hadoop's framework, IT organizations and researchers can build applications that tailor the data analysis to specific needs for each company, even using unstructured data. Many different markets—among them finance, IT, and retail—use Hadoop due to its ability to handle heterogeneous data, both structured and unstructured.

Hadoop can run across any number of machines using varied hardware, spreading data across all available hardware resources using a distributed file system, Hadoop Distributed File System (HDFS), and replicating data to minimize loss if a hardware malfunction occurs. The software is able to detect hardware failures, and to work around said failures to allow uninterrupted access to data. Because of its ability to run on different hardware, a Hadoop cluster is scalable and flexible – it can be expanded to encompass growing databases and companies. It is also cost-effective as it allows companies to utilize commodity hardware effectively.

MapReduce

MapReduce is a framework within Hadoop that provides the ability to process extremely large datasets in parallel across the Hadoop cluster, shortening the overall processing time greatly. MapReduce breaks input data down into chunks to be processed across the Hadoop cluster. When an application is run on a Hadoop cluster, MapReduce perfoms "map" tasks that process data in parallel. The data is then sent to "reduce" tasks that reduce the information into a final result. This allows for faster data processing using multiple nodes, while still producing a single, comprehensive, accurate result.

TEST RESULTS IN MORE DETAIL

We used mrbench to create datasets containing varying numbers of records, which each contained 6 bytes, for datasets varying in size from 96 bytes to 9,600,000 bytes. We tested two mrbench configurations; in the first, we applied 16 maps and one reduction process to those input recordsets, and in the second, we applied 160 maps and 16 reduction processes to those input recordsets.

Figure 4 shows the time in milliseconds the cluster took to complete each job when we applied 16 maps and one reduction process to those input recordsets.

Task	Number of records	Total bytes	Time to complete (milliseconds)
Task 1	16	96	14,955
Task 2	160	960	14,559
Task 3	1,600	9,600	14,515
Task 4	16,000	96,000	14,851
Task 5	160,000	960,000	14,846
Task 6	1,600,000	9,600,000	15,878

Figure 3: The time spent for each job of the Hadoop MapReduce benchmark.

Figure 4 shows the time in milliseconds the cluster took to complete each job when we applied 160 maps and 16 reduction processes to those input recordsets.

Task	Number of records	Total bytes	Time to complete (milliseconds)
Task 1	16	96	22,374
Task 2	160	960	22,057
Task 3	1,600	9,600	22,496
Task 4	16,000	96,000	21,648
Task 5	160,000	960,000	22,122
Task 6	1,600,000	9,600,000	25,594

Figure 4: The time spent for each job of the Hadoop MapReduce benchmark.

CONCLUSION

Companies that wish to process big data in scale-out environments require powerful and dense hardware that complete large data processing tasks, such as Hadoop analysis, quickly and efficiently. Selecting the right server for your underlying hardware infrastructure is critical at hyperscale.

In our tests, an eight-node Dell PowerEdge C5220 Hadoop cluster was able to efficiently process multiple MapReduce scenarios. While each scenario and each environment vary, our eight-node Hadoop MapReduce test shows that even a small number of Dell PowerEdge C5220 microservers can process varying amounts of data quickly.

APPENDIX A – SERVER CONFIGURATION INFORMATION

Figure 5 provides detailed configuration information for the test servers.

System	Dell PowerEdge C5220
General	
Number of processor packages	1
Number of cores per processor	4
Number of hardware threads per core	1
System power management policy	Balanced
CPU	
Vendor	Intel
Name	Xeon
Model number	E3-1220
Stepping	D2
Socket type	LGA1155
Core frequency (GHz)	3.20
Bus frequency	5 GT/s
L1 cache	32 KB + 32 KB (per core)
L2 cache	256 KB (per core)
L3 cache (MB)	8
Platform	
Vendor and model number	Dell PowerEdge C5220
Motherboard model number	1KTH4B0
BIOS name and version	Dell 1.0.11
BIOS settings	Defaults
Memory module(s)	
Total RAM in system (GB)	32
Vendor and model number	Samsung M391B1G73AH0-YH9
Type	PC3L-10600E
Speed (MHz)	1,333
Speed running in the system (MHz)	1,333
Timing/Latency (tCL-tRCD-tRP-tRASmin)	9-9-9-36
Size (GB)	8
Number of RAM module(s)	4
Chip organization	Double-sided
Rank	Dual
Operating system	
Name	CentOS 6.2, x86_64
File system	ext4
Kernel	2.6.32-220.13.1.el6.x86_64
Language	English
Updates	All as of 4/12/2012
Graphics	
Vendor and model number	AST2050

System	Dell PowerEdge C5220
Graphics memory (MB)	128
RAID controller	
Vendor and model number	Intel C204 SATA Raid Controller
Driver version	N/A
Cache size (GB)	None
Hard drive	
Vendor and model number	Dell ST9500620NS
Number of disks in system	4
Size (GB)	500
Buffer size (MB)	64
RPM	7.2K
Type	SATA
Ethernet adapters	
Vendor and model number	Intel 82580DB Gigabit Network Connection
Type	Integrated
Driver	Intel(R) Gigabit Ethernet Network Driver; igb, 3.0.6-k
USB ports	
Number	1 internal
Type	2.0

Figure 5: Configuration details for the two test servers.

APPENDIX B – HOW WE TESTED

We installed and configured CentOS on an auxiliary server, which provided both basic services and Hadoop management for the C5220 nodes in the Hadoop cluster. We installed and configured CentOS on the C5220 nodes, using the C5000's baseboard management controller (BMC) virtual console feature. We installed the Cloudera Hadoop Manager on the auxiliary manager server and used it to install Hadoop on the C5220 nodes.

Configuring the auxiliary server

Below, we review the steps necessary to configure the management server, which runs the necessary auxiliary services and the Cloudera Manager.

1. Install the base-installation of CentOS 6.2 x86_64 following the steps in the section CentOS base installation.
2. Log onto the server as root.
3. Configure one network interface, e.g. eth0, to access a network connected to the Internet with, for example, IP address 10.1.1.10.
4. Configure one network interface, e.g., eth1, to access the network of C5220 nodes with, for example, IP address 192.168.1.10 and CIDR prefix of 24.
5. Install CentOS updates:
   ```
   yum update
   ```
6. Reboot the server:
   ```
   shutdown -r now
   ```
7. After the system reboots, log onto the server as root.
8. Install Squid as a HTTP proxy for the C5220 nodes:
   ```
   yum install squid
   ```
9. Modify /etc/squid/squid.conf configuration to add these settings:
   ```
   acl ftp proto FTP
   http_access allow ftp
   ## cluster network
   acl intranet src 192.168.1.0/21
   http_access allow intranet
   ```
10. Install dnsmasq to supply hostname resolution for the nodes:
    ```
    yum install dnsmasq
    ```
11. Edit the configuration /etc/dnsmasq.conf and modify these lines as shown:
    ```
    interface=eth1
    no-dhcp-interface=eth1
    ```
12. Add the IP addresses and hostnames of the auxiliary server and C5220 nodes to /etc/hosts on the auxiliary server. For example:
    ```
    192.168.1.10    had01-ctrl01
    192.168.1.31    cl01n01
    192.168.1.32    cl01n02
    192.168.1.33    cl01n03
    192.168.1.34    cl01n04
    192.168.1.35    cl01n05
    192.168.1.36    cl01n06
    192.168.1.37    cl01n07
    192.168.1.38    cl01n08
    ```
13. Start the DNS server:
    ```
    chkconfig dnsmasq on
    ```

```
service dnsmasq start
```
14. Install the ntpd time server to provide a common time source for the nodes:
```
yum install ntpd
```
15. Modify the configuration file /etc/ntp.conf to use Internet time sources, and to be a time service on the nodes
network. For example:
```
# Hosts on local network are less restricted.
restrict 192.168.1.0 mask 255.255.255.0
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
server 0.pool.ntp.org
```
16. Start the NTP server:
```
chkconfig ntpd on
service ntpd start
```
17. Generate an SSH public-key pair with, for example, no password:
```
ssh-keygen -t dsa -N ""
```
18. Reboot the server.

Configuring the C5220 Hadoop nodes

1. Install the base-installation of CentOS 6.2 x86_64 following the steps in the section CentOS base installation.
2. Configure one network interface, e.g. em1, to access the Hadoop network of C5220 nodes with, for example, IP
address 192.168.1.31 and CIDR prefix of 24.
3. Configure the system to synchronize its time with the NTP service on the auxiliary server. Replace the server
lines in /etc/ntp.conf with this:
```
server 192.168.1.10
```
4. Configure the yum updater to use the cluster's HTTP proxy:
```
echo "proxy=http://192.168.1.10:3128/" >> /etc/yum.conf
```
5. Copy the auxiliary server's public SSH key to the node to allow easy remote management:
```
mkdir /root/.ssh
chmod 700 /root/.ssh
scp 192.168.1.10:/root/.ssh/id_dsa.pub /root/.ssh/authorized_keys
```
6. Install CentOS updates:
```
yum update
```
7. Reboot the server:
```
shutdown -r now
```
8. Repeat steps 1 through 7 for the remaining nodes.

Cloudera Hadoop (CDH3u3) installation using Cloudera manager (3.7.5)

1. Log onto the auxiliary server as root.
2. Download and transfer the free edition of the Cloudera Manager installer, version 3.7.5, for CentOS from
Cloudera.com to the auxiliary server.
3. Run the Cloudera Manager installer:
```
./cloudera-manager-installer.bin
```
4. On the Cloudera Manager README screen, read the installation overview, and select Next.
5. On the Cloudera Manager (Free Edition) License screen, review the license, and select Next.
6. On the next screen, select yes to accept this license.
7. On the Oracle Binary Code License Agreement screen, review the license, and select Next.

---

8. On the next screen, select Yes to accept this license.
9. The auxiliary server now has a Web interface for managing Hadoop. Select OK on the Next Step screen.
10. On the installer's Finish screen, select OK to exit the installer.
11. Using a Web browser, open the page http://<clouderamanager>:7180/ , where <clouderamanager> is the IP address of the auxiliary manager server.
12. Copy the SSH keys, previously generated in the Configuring the auxiliary server section, from the auxiliary server to the local desktop running the Web browser.
13. Log onto the Cloudera manager with username `admin` and password `admin`.
14. On the Thank you for choosing Cloudera Manager and Cloudera's Distribution including Apache Hadoop (CDH) screen, click Continue.
15. On the register your CDH3 installation, click Skip Registration.
16. On the Specify hosts for your CDH3 cluster installation screen, enter a list of hostnames or IP addresses in the text box at the bottom, such as 192.168.1.3[1-8]. Click Find Hosts.
17. The upper text box will contain a list of potential Hadoop hosts. Select the applicable nodes, and click Continue.
18. On the provide SSH login credentials screen, select All hosts use the same public key, click Choose file for the Public Key File, and browse to and select the SSH public-key file for the auxiliary server's public key (see step 12).
19. Repeat step 18 to select the auxiliary server's Private Key File.
20. Click Start Installation.
21. After the Hadoop installation on the nodes has finished, switch to a console session on the auxiliary server.
22. Format two of the remaining four disks on each node as part of the Hadoop file system.

```
# Note: in the following example, the contents of disks /dev/sdc and /dev/sdd
#       will be destroyed
# partition /dev/sdc and /dev/sdd and create EXT4 file systems

for i in $(seq 31 38); do
   ssh 192.168.1.$i parted -s /dev/sdc mklab gpt \; parted -s /dev/sdd mklab gpt
   ssh 192.168.1.$i parted /dev/sdc mkpart primary  '"1 -1"' \; \
                    parted /dev/sdd mkpart primary  '"1 -1"'
   ssh 192.168.1.$i mkfs.ext4 /dev/sdc1 \; mkfs.ext4 /dev/sdd1
done

# Modify fstab so that these file systems are mounted at boot time

for i in $(seq 31 38); do
   ssh 192.168.1.$i '(echo "/dev/sdc1 /dfs/d1 ext4 defaults,noatime 1 2";\
                      echo "/dev/sdd1 /dfs/d2 ext4 defaults,noatime 1 2") \
                    >> /etc/fstab'
done

# Create the default Hadoop HDFS directories

for i in $(seq 31 38); do
   ssh 192.168.1.$i mkdir -p '/dfs/{d1,d2,n1,n2,s1,s2}' '/mapred/{local,jt}'
   ssh 192.168.1.$i mount '/dfs/d{1,2}'
done

for i in $(seq 31 38); do
   ssh 192.168.1.$i chmod 700 '/dfs/{d1,d2,n1,n2,s1,s2}' \;
                    chmod 755 '/mapred/{local,jt}'
```

```
        ssh 192.168.1.$i chown hdfs:hadoop '/dfs/{d1,d2,n1,n2,s1,s2}' \;
                        chown mapred:hadoop '/mapred/{local,jt}'
    done
```
23. Return to the Cloudera Manager in the browser.
24. On the Services screen, click Add a Service.
25. On the Select the type of service you want screen, select HDFS, and click Continue.
26. Accept the default host assignments for HDFS (eight DataNodes and one each of NameNode, Secondary NameNode and Balancer nodes) and click Continue.
27. Review the configuration changes, and click Continue.
28. On the Services screen, click Add a Service.
29. On the Select the type of service you want screen, select MapReduce, and click Continue.
30. On the Select the set of dependencies for your new service, select HDFS, and click Continue.
31. Accept the default host assignments for HDFS (eight TaskTrackers and one JobTracker nodes) and click Continue.
32. Review the configuration changes, and click Continue.

Running the MapReduce benchmark

1. Log onto one of the Hadoop nodes as root.
2. Switch login shell to the mapred user:
   ```
   su - mapred
   ```
3. Invoke mrbench, providing the necessary parameters. In our tests, we ran mrbench with varying sizes of data, maps, and reduces, which we specify in the report. For example, to process one file containing 16,000 lines of data with 160 maps and 16 reduce processes, use the following parameters:
   ```
   hadoop jar hadoop-test.jar  mrbench -maps 160  -reduces 16 -inputLines 16000
   ```

CentOS 6.2 base installation

1. Insert and boot from the CentOS-6.2-x86_64-bin-DVD1 installation DVD.
2. At the welcome screen, select Install or upgrade an existing system, and press Enter.
3. At the Media test screen, select Skip, and press Enter.
4. At the CentOS 6 title screen, click Next.
5. At the Choose an Installation Language screen, select English, and click Next.
6. At the Keyboard Type screen, select U.S. English, and click Next.
7. At the Storage Devices screen, select Basic Storage Devices, and click Next.
8. If a warning for device initialization appears, select Yes, discard any data.
9. At the Name the Computer screen, type the host name, and click Configure Network.
10. At the Network Connections screen, select the server's main or management network interface, and click Edit.
11. At the Editing network interface screen, check Connect Automatically.
12. On the same screen, Select the IPv4 Settings tab, change the Method to Manual, and click Add.
13. On the same screen, enter the IP address, Netmask, Gateway, and DNS server. Click Apply.
14. Click Close on the Network Connections screen, and click Next on the Name the Computer screen.
15. At the Time zone selection screen, select the appropriate time zone, and click Next.
16. Enter the root password in the Root Password and Confirm fields, and click Next.
17. At the Partition selection screen, select Replace Existing Linux System(s), and click Next.
18. If a warning appears, click Write changes to disk.

---

19. At the default installation screen, click Next to begin the installation.
20. At the Congratulations screen, click Reboot.
21. After the system reboots, log in as root.
22. Disable SELinux by editing the file /etc/selinux/config, and change the line `SELINUX=enforcing` to `SELINUX=disabled`. These changes take effect after rebooting.
23. Disable these unused services by running the following command-line script:

```
CHK_OFFs="auditd autofs cups ip6tables iptables nfslock netfs portreserve postfix\
          qpidd rhnsd rhsmcertd rpcgssd rpcidmapd rpcbind"
for i in ${CHK_OFFs}; do
  chkconfig $i off
  service $i stop
done
```

ABOUT PRINCIPLED TECHNOLOGIES

Principled Technologies®

Principled Technologies, Inc.
1007 Slater Road, Suite 300
Durham, NC, 27703
www.principledtechnologies.com

We provide industry-leading technology assessment and fact-based marketing services. We bring to every assignment extensive experience with and expertise in all aspects of technology testing and analysis, from researching new technologies, to developing new methodologies, to testing with existing and new tools.

When the assessment is complete, we know how to present the results to a broad range of target audiences. We provide our clients with the materials they need, from market-focused data to use in their own collateral to custom sales aids, such as test reports, performance assessments, and white papers. Every document reflects the results of our trusted independent analysis.

We provide customized services that focus on our clients' individual requirements. Whether the technology involves hardware, software, Web sites, or services, we offer the experience, expertise, and tools to help our clients assess how it will fare against its competition, its performance, its market readiness, and its quality and reliability.

Our founders, Mark L. Van Name and Bill Catchings, have worked together in technology assessment for over 20 years. As journalists, they published over a thousand articles on a wide array of technology subjects. They created and led the Ziff-Davis Benchmark Operation, which developed such industry-standard benchmarks as Ziff Davis Media's Winstone and WebBench. They founded and led eTesting Labs, and after the acquisition of that company by Lionbridge Technologies were the head and CTO of VeriTest.