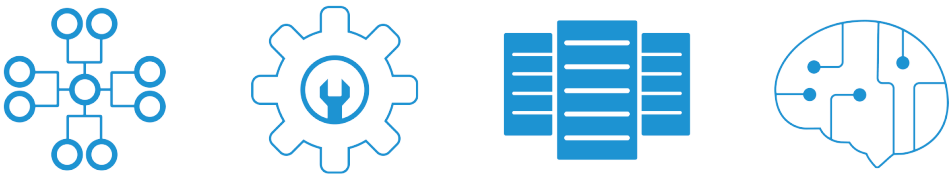
Apache Hadoop® with Apache Spark™ Data Analytics Using Micron® 9300 and 5210 SSDs

Reference Architecture

Sujit Somandepalli, Principle Storage Solutions Engineer
Tony Ansley, Principle Technical Marketing Engineer



Contents

Executive Summary

This document describes an example configuration of a performance-optimized big data analytics solution using Apache Hadoop® Distributed File System (HDFS) and Apache Spark™ data analytics software in a scalable cluster configuration using Micron® NVMe™ and SATA SSDs, standard x86 architecture rack-mount servers and 100 GbE networking.

It details the hardware and software building blocks used to construct this reference architecture (including the Red Hat® Enterprise Linux® OS configuration, network switch configurations, and Apache software configuration and tuning parameters) and shows the performance test results and measurement techniques for a scalable 4-node HDFS with Spark architecture.

This tiered NVMe + SATA solution is optimized to maximize performance in a compact, rack-efficient design to enable:

- Faster deployment: The configuration has been pre-validated and is thoroughly documented to enable faster deployment.
- Balanced design: The right combination of NVMe and SATA SSDs, DRAM, processors, and networking ensures subsystems are balanced and performance-matched.
- Broad use: Complete tuning and performance characterization across multiple IO profiles enables broad deployment across multiple uses.

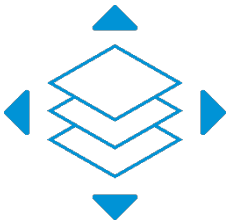
The result is exceptional performance results based on tests executed using industry standard data analytics benchmarks:

Benchmark	Throughput (MB/s)
Sort	1,450 MB/s
TeraSort	470 MB/s
Wordcount	5,230 MB/s

Why Micron for this Solution

SSDs and DRAM can represent up to 80 percent of the value of today's advanced server/storage solutions. Micron is a leading designer, manufacturer, and supplier of advanced storage and memory technologies with extensive in-house software, application, workload, and system design experience.

Micron's silicon-to-systems approach provides unique value in our reference architectures, ensuring these core elements are engineered to perform in highly demanding applications like Hadoop and holistically balanced at the platform level. This reference architecture solution leverages decades of technical expertise as well as direct, engineer-to-engineer collaboration with industry leaders.



Micron Reference Architectures

Micron reference architectures are optimized, pre-engineered, enterprise-leading solution templates co-developed between Micron and industry-leading hardware and software companies.

Designed and tested at Micron's Storage Solutions Center, they provide end users, system builders, independent software vendors (ISVs) and OEMs with a proven template to build next-generation solutions with reduced time investment and risk.

Architecture Overview

The data analytics solution offered here includes an Apache HDFS storage cluster built from large numbers of x86 industry standard server nodes providing scalability, fault-tolerance, and performant storage. Apache Spark is also distributed across each node to perform data analytics processing within the HDFS file system. The solution uses Apache Hadoop YARN™ for assignment and management of analytics jobs deployed to each Spark instance.

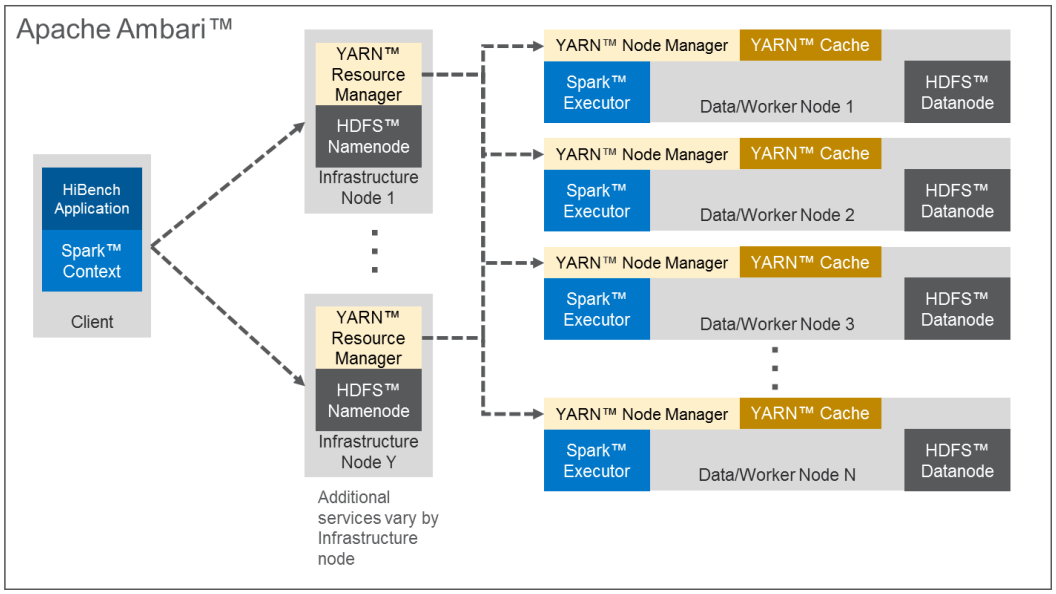


Figure 1: Architecture Overview—Data analytics with Apache Spark, HDFS and YARN

This data analytics reference architecture uses a series of open-source components from the Apache Foundation. The following sections describe each component and its role in the solution.

Apache Hadoop File System (HDFS)

Designed for reliable storage of very large files distributed across multiple host servers within a scalable cluster, HDFS provides hosting of processed data. HDFS is based on the Google File System introduced by Alphabet in 2003. HDFS allows the configuration of variable block size and replication factors on a file-by-file basis, allowing greater flexibility, fault-tolerance and availability.

HDFS uses a distributed architecture consisting of namenodes and datanodes. Each namenode manages the file system and regulates access to files by clients. It also determines the mapping of blocks to the datanodes. Each datanode in the cluster provides the actual storage devices to HDFS. The datanodes are responsible for serving read and write requests from filesystem clients. They also perform block creation, deletion and replication upon instruction by the namenodes.

Apache Hadoop YARN

Apache YARN is a general-purpose, distributed application management framework that supersedes the classic Apache Hadoop MapReduce framework for processing data in enterprise Hadoop clusters.

YARN breaks up the functionalities of resource management and job scheduling/monitoring into separate daemons. The idea is to have a global ResourceManager and a per-application ApplicationMaster. An application is either a single job or multiple jobs.

The ResourceManager and the NodeManager form the data-computation framework. The ResourceManager is the ultimate authority that arbitrates resources among all the applications in the system. The NodeManager is the per-machine framework agent that is responsible for jobs, monitoring their resource usage and reporting back to the ResourceManager.

YARN Cache

The YARN shared cache provides the facility to upload and manage shared application resources to HDFS in a safe and scalable manner. YARN applications can leverage resources uploaded by other applications or previous runs of the same application without having to re-upload and localize identical files multiple times. YARN saves network resources and reduces application startup time.

It is critical that this shared cache be high-performance and low-latency. SSDs make great options for YARN Cache, resulting in better application performance; high-performance NVMe SSDs, in particular, provide low latency and high-performance access to shared resources.

Apache Spark

Apache Spark is a unified analytics engine for large-scale data processing. Spark is a fast, general-purpose cluster computing platform that allows applications to run as independent sets of processes on a cluster of compute nodes, coordinated by a driver program (SparkContext) for the application. (The SparkContext can connect to several types of cluster managers including YARN used in this reference architecture).

Once connected, Spark acquires executors on nodes in the cluster, which are processes that run computations and store data for your application. Spark then sends the application code to these executors. For this reference architecture, YARN is the cluster manager.

Apache Ambari™

Apache Ambari provides a consolidated solution through a graphical user interface for provisioning, monitoring, and managing the Hadoop cluster. Operations supported by Ambari include:

- Graphical wizard-based installation of Hadoop services, ensuring the applications of consistent configuration parameters to each Hadoop node based on its role in the Hadoop cluster.
- Centralized graphics console to allow for easily starting, stopping or reconfiguring Hadoop services.
- Dashboards containing monitoring information related to the health and status of the Hadoop cluster including all resources consumed.

Additional Services

Several additional services and client software are installed on the infrastructure nodes as part of the installations of Spark, HDFS, and Ambari. These services are not within scope of this reference architecture. These services are available if needed for your specific needs.

Design Overview

This design uses the Intel® Purley platform with Intel Xeon® 8168 processors. This combination provides the high CPU performance required for a performance-optimized big data cluster and yields an open, cost-effective Hadoop platform.

The Micron 9300 NVMe SSD offers tremendous performance with low latencies required for a YARN cache. Micron 5210 ION SATA SSDs provide high capacity and excellent performance at a lower cost than previously available for SSDs. Each datanode contains one 3.84TB 9300 SSD and twelve 7.68TB 5210 SSDs. This entire reference architecture takes up 16RU (including the four infrastructure nodes) and can be easily scaled up 2U and 96TB at a time.

Mellanox® ConnectX®-4 100 GbE network cards installed in each node handle all inter-node and client-cluster communications.

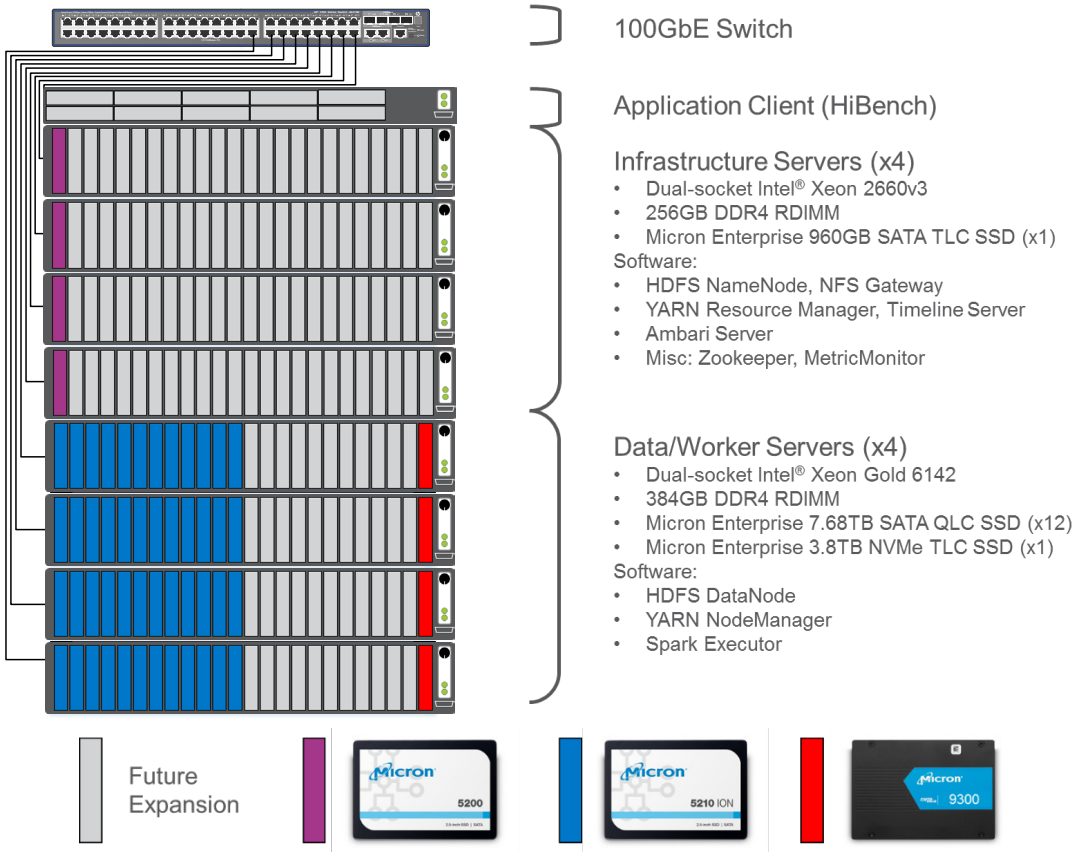


Figure 2: Reference Architecture Design Overview

Software Components

This section details the software versions used in the reference architecture. Refer to Figure 1 for location of specific software components and roles.

Cloudera™ HDP

Cloudera HDP helps enterprises implement a Hadoop data management solution to gain insights from structured and unstructured data. Based on the open-source Apache Hadoop framework paired with structured support services, HDP provides a suite of Apache applications and functionality designed for big data solutions. Core components of HDP are described later in this document.

For this reference architecture, Cloudera HDP 3.1.1 provides all data analytics and storage services. For more information about Cloudera HDP visit https://hortonworks.com/products/data-platforms/hdp/.

CentOS® Linux

CentOS Linux, a high-performance operating system based on Red Hat® Enterprise Linux®, is used by IT environments for production solutions that scale. This reference architecture uses version 7.6.1810 on all nodes.

Software by Node Type

Table 2 shows the software and version numbers used in the infrastructure and data/compute.

Operating System	CentOS Linux	7.6
Hadoop	Cloudera HDP	3.1.1 HDFS 3.1.1 YARN 3.1.1 Zookeeper 3.4.6 Spark 2.3.2 Ambari 2.7.3
NIC Driver	Mellanox	12.17.2020

Table 2: Hadoop Infrastructure and Data/Compute Nodes—Software

Micron Components

Micron® 5210 ION SATA SSD

The Micron 5210 ION is the world's first SSD to market with groundbreaking quad-level cell (QLC) NAND technology, delivering fast capacity for less. It is ideal for handling the demands of real-time analytics, big data, media streaming, block/object stores, SQL/NoSQL and the data lakes that feed artificial intelligence (AI) and machine learning, such as Apache Hadoop. Available in capacities from 1.92TB to 7.68TB, the 5210 ION is a highly read-centric solution for large, primarily read workloads such as big-data analytics.

This reference architecture uses the 5210 ION 7.68TB SSD. Table 3 summarizes the specifications.

Model	5210 ION	**Interface**	6 Gb/s SATA
Form Factor	2.5-inch	**Capacity**	7.68TB
NAND	Micron 3D QLC	**MTTF**	2M device hours
Sequential Read[1]	540 MB/s	**Random Read**	83,000 IOPS
Sequential Write[1]	350 MB/s	**Random Write**	6500 IOPS

Table 3: 5210 ION Specifications Summary

Micron® 9300 NVMe™ SSD

The Micron 9300 series of NVMe™ SSDs is our flagship performance family and our third generation of NVMe SSDs. The 9300 family has the right capacity for demanding workloads, ranging from 3.2TB to 15.36TB in mixed-use, read-intensive designs. The 9300 is also the first offering from Micron to provide "no compromise" read and write performance by offering balanced 3500 MB/s throughput on certain models[2] .

This reference architecture uses the 9300 PRO 3.84TB SSD. Table 4 summarizes the specifications.

Model	9300 PRO	**Interface**	PCIe Gen 3 x4
Form Factor	U.2	**Capacity**	3.84TB
NAND	Micron® 3D TLC	**MTTF**	2M device hours
Sequential Read[1]	3.5 GB/s	**Random Read**	850,000 IOPS
Sequential Write[1]	3.5 GB/s	**Random Write**	310,000 IOPS
Endurance	144.8PB	**Status**	Production

Table 4: 9300 PRO Specifications Summary

[1] MB/s measured using 128K transfers, IOPS measured using 4K transfers. All data is steady state. Complete MTTF details are available in the product data sheet.

[2] Offered on 9300PRO 7.68TB and 15.36TB and 9300MAX 6.4TB and 12.8TB models.

Server Platforms

This reference architecture used standard 2RU, x86 architecture dual-socket servers. While implementation of this reference architecture consisted of a specific vendor-provided configuration, the vendor and model number specifics are not pertinent to the design. The descriptions below provide the pertinent details of the server configuration(s) to allow the reader to provide any server vendor with the requirements required to match the results presented in this reference archtiecture. Contact your server vendor regarding availability.

HDP Compute/Data Node Hardware

Compute/data nodes used 2RU servers with the following specifications:

Server Architecture	Intel dual-socket, PCIe Gen3 (“Purley”) architecture
CPU (x2)	Intel Xeon Gold 6142: 16 cores, 32 threads, 2.6 GHz base
DRAM (x12)	Micron 32GB DDR4-2666 MT/s, 384GB total per node
Capacity (x12)	Micron 5210 ION SATA SSDs, 7.68TB each
Cache (x1)	Micron 9300 PRO NVMe SSD, 3.84TB
SATA (OS)	64GB SATA Disk-on-Motherboard
Network Adapter	1x Mellanox ConnectX-4 100 GbE dual-port (MCX416A-CCAT)

Table 5: Compute/Data Node Hardware Details

HDP Infrastructure Node Hardware

Infrastructure nodes used 2RU servers with the following specifications:

Server Architecture	Intel dual-socket, PCIe Gen 3 (“Grantley”) architecture
CPU (x2)	Intel Xeon 2660v3: 10 cores, 20 threads, 2.6 GHz base
DRAM (x12)	Micron 32GB DDR4-2666 MT/s, 256GB total per node
OS/Data (x1)	Micron 5100 SATA SSD, 960GB
Network Adapter	1x Mellanox ConnectX-4 100 GbE dual-port (MCX416A-CCAT)

Table 6: Infrastructure Node Hardware Details

Switches

The Mellanox SN2700 builds on a foundation of 25 GbE and 100 GbE standards to provide a high-quality switch suitable for high-performance CLOS spine/leaf or top-of-rack solutions. Providing up to 32x 100 GbE ports per switch, the SN2700 is a great solution for large-scale big-data analytics and AI/ML infrastructures.

Model	Mellanox SN2700
Software	Onyx 3.7.1000

Table 7: Network Switches (Hardware and Software)

Network Interface Cards

The ConnectX-4 EN network controller with two ports of 100 Gb/s Ethernet connectivity and advanced offload capabilities delivers high bandwidth, low latency and high computation efficiency for high performance, data-intensive and scalable HPC, Cloud, data analytics, database, and storage platforms.

Measuring Performance

Test Methodology

Testing was performed using several component tests from HiBench, a recognized benchmark suite for big data solutions. Each compute/data node provided storage via the HDFS storage services. All analytics were performed using Spark as an alternative to legacy Hadoop MapReduce.

Server Storage Configuration

Configuration of the solution followed these general procedures:

1. Secure erase all data and cache drives.
2. Format data and cache drives using XFS.
3. Mount each Micron 5210 data drive to `/hadoop/ssd<x>`, where `<x>` indicates a unique number from 0 to 11, using the following mount options:
   a. `noatime`
   b. `nodiratime`
   c. `norelatime`
   d. `discard`
4. Mount each Micron 9300 cache drive to `/hadoop/yarn` on each compute/data node.

   See the tuning parameters determined for each test and documented in Appendix A.
5. Execute multiple iterations of each test; provide results as the statistical mean of each test. Actual results from each test run will differ from those reported below.

   Data set sizes were determine for each test listed in the table below.

Sort	350GB
TeraSort	640GB
Wordcount	1.64TB

Table 8: Data Set Sizes for Each Test

Benchmark Tests

HiBench is a big data benchmark suite that helps evaluate different big data frameworks in terms of speed, throughput and system resource utilization. Consisting of 19 benchmarks divided into six categories, this reference architecture focused on the Micro Benchmark subset, which focuses on evaluating a typical Hadoop big data analytics use case. Micro Benchmark consists of a set of workloads including Sort, TeraSort, WordCount, DFSIO, and enhanced DFSIO.

Sort

The HiBench Sort benchmark performs a sort algorithm against data—a representative workload for a Hadoop Map-Reduce job—that transforms an initial data set from one state to another. In this instance, this reference architecture replaced MapReduce with Apache Spark. Data generated for Sort results from executing the Hadoop RandomTextWriter program.

WordCount

WordCount counts the occurrence of each word in the input data. As with Sort, this data is also generated using RandomTextWriter. WordCount is another example of a typical real-world MapReduce job—evaluating a dataset and deriving a small set of information about that dataset.

TeraSort

TeraSort is another sorting algorithm similar to the Sort benchmark. One major difference between Sort and TeraSort involves differences in the dataset used. TeraSort uses a large dataset consisting of relatively small records generated by the TeraGen program included in the Hadoop distribution. The resulting dataset consists of 10 billion 100-byte records.

All tests used HiBench version 7.0 which was modified to include address some minor incompatibilities between HiBench and HDP 3.1.1.

Baseline Testing

To provide context for the performance of SSD-based configurations and test runs, we executed each test using an identical server configuration for compute/data nodes using 12 8TB 7.2K RPM hard disk drives (HDDs) in place of the 12 Micron 5210 ION SSDs. This class of HDD is a common storage implementation for large big data analytics deployments and provides a suitable baseline for SSD performance comparisons.

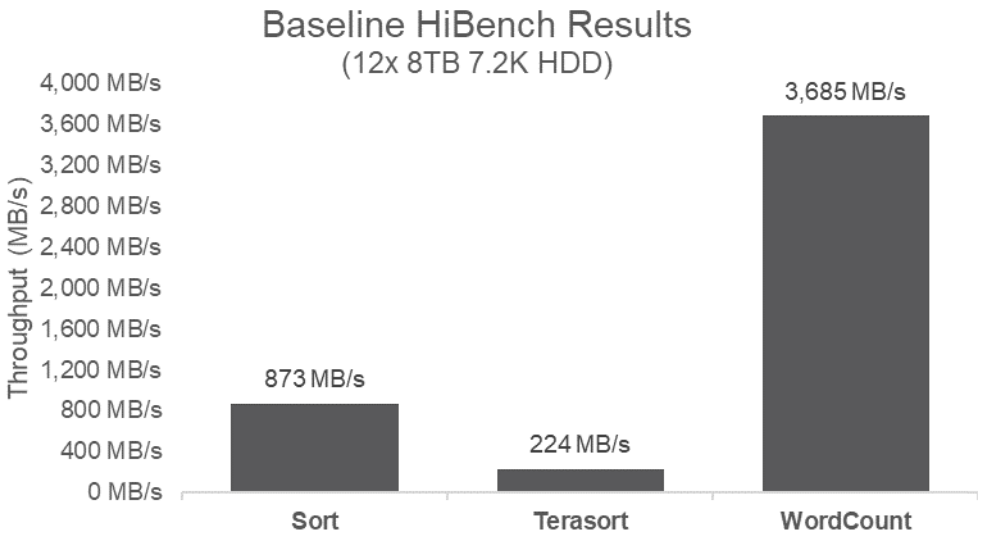


Figure 3: Baseline Test Results for 12 8TB 7200 RPM Hard Disk Drives

Test Results and Analysis

The servers used HDFS 3x replication for all data/compute nodes data storage. Overall, there is a clear advantage to using SSDs in an all-flash configuration versus legacy high-capacity HDDs. As shown in Figure 4, the execution of each test is significantly faster than the baseline using 12 HDDs. Depending on the benchmark considered, an all-SATA all-flash configuration can be anywhere from 30-41% faster than HDDs. Adding an NVMe SSD as a YARN cache improves the overall performance advantage over HDDs to between 30% and 52%.

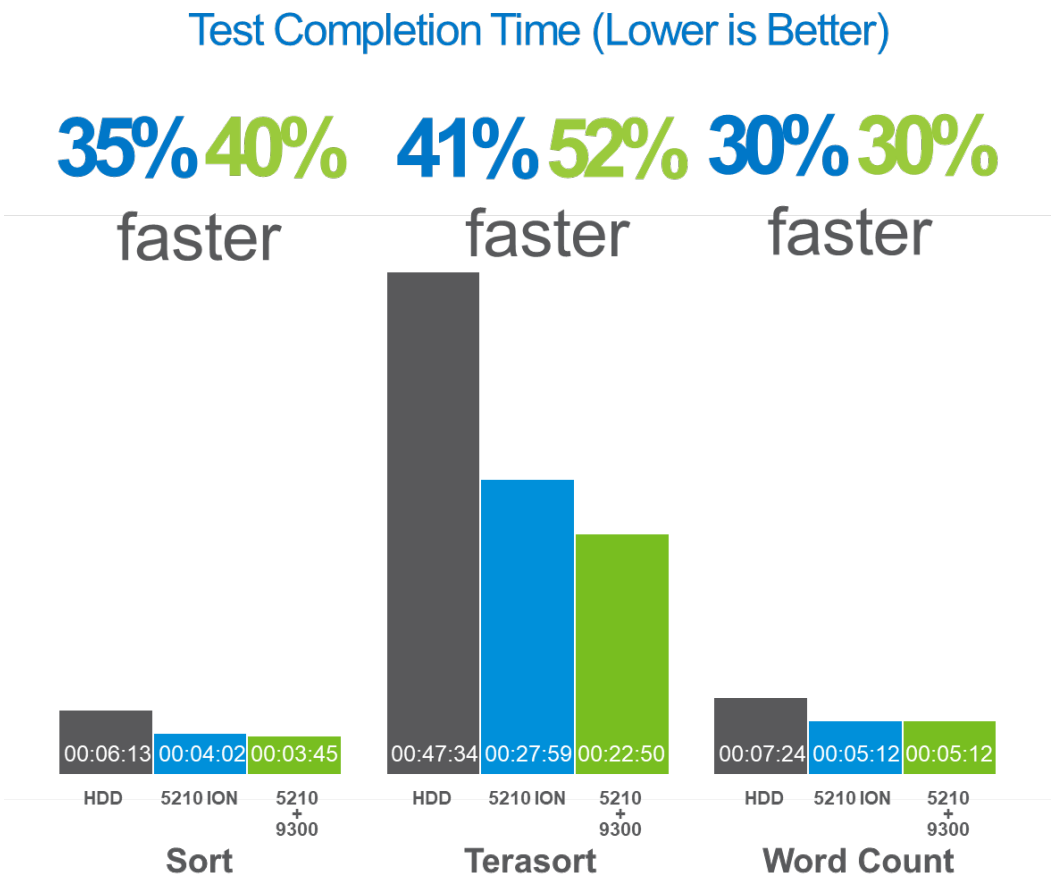


Figure 4: Summary of HiBench Test Results – Execution Time in hh:mm:ss

HiBench Sort Results

Figures 5a and 5b provide the mean performance results for the HiBench Sort benchmark.

A 66% improvement is seen in throughput generated by the all-5210 SSD configuration over the baseline, with 5210 throughput measured over 1.3 GB/s. The throughput level resulted in the execution time for the 5210 configuration to complete 35% faster than the baseline.

Adding a single 9300 NVMe SSD as a YARN cache improved these results significantly. The cached configuration showed an average throughput of over 1.45 GB/s, resulting in an execution time of over 40% faster than the baseline.

Figures 5a-5b: HiBench Sort Results: Execution Time (Shorter is Better) and Throughput (Taller is Better)

HiBench WordCount Results

Figures 6a and 6b provide the mean performance results for the HiBench WordCount benchmark.

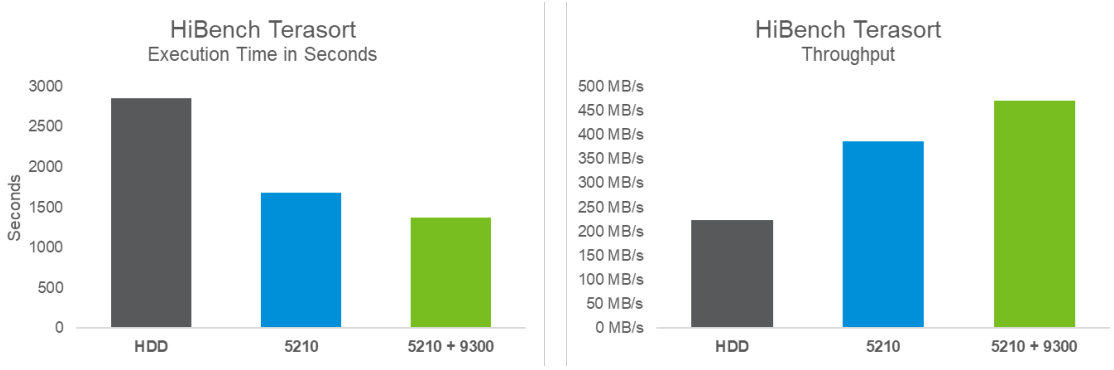
Figures 6a-6b: HiBench WordCount Results: Execution Time (Shorter is Better) and Throughput (Taller is Better)

WordCount experienced a 42% improvement in throughput generated by the all-5210 SSD configuration over the baseline, with 5210 throughput measured over 5,220 MB/s. The throughput level resulted in the execution time for the 5210 configuration to complete 30% faster than the HDD baseline.

Adding a single 9300 NVMe SSD as a YARN cache saw a similar 42% improvement over the baseline. The cached configuration showed an average throughput of over 5,230 MB/s, resulting in an execution time of over 30% faster than the baseline.

HiBench TeraSort Results

Figures 7a and 7b provide the mean performance results for the HiBench TeraSort benchmark.



Figures 7a-7b: HiBench TeraSort Results: Execution Time (Shorter is Better) and Throughput (Taller is Better)

TeraSort experienced a 72% improvement in throughput generated by the all-5210 SSD configuration over the baseline, with 5210 throughput measured over 386 MB/s. The throughput level resulted in the execution time for the 5210 configuration to complete 41% faster than the baseline.

Adding the 9300 NVMe SSD as a YARN cache improved these results significantly. The cached configuration showed an average throughput of over 471 MB/s, resulting in an execution time of over 52% faster than the baseline.

	Baseline	5210 ION	9300 + 5210
Sort	373.50	241.59	225.06
TeraSort	2,853.98	1,678.98	1,369.84
Wordcount	443.71	311.91	311.74

Table 9: HiBench Benchmark Completion Times (seconds)

	Baseline	5210 ION	9300 + 5210
Sort	373.50 MB/s	241.59 MB/s	225.06 MB/s
TeraSort	2,853.98 MB/s	1,678.98 MB/s	1,369.84 MB/s
Wordcount	443.71 MB/s	311.91 MB/s	311.74 MB/s

Table 10: HiBench Benchmark Performance Results

Summary

Micron designed this reference architecture to illustrate the advantages of deploying big data solutions based on a commercially available release of the Apache Hadoop solution framework. Capable of generating significant performance improvements over legacy HDD-based big data environments, Micron SSDs such as the Micron 5210 ION, based on our ground-breaking quad-level cell NAND technology, provide cost-effective flash solutions to meet business needs for a high-performance, real-time analytics capable solution.

Micron reference architectures provide a significant advantage for customers looking to lower risk by providing a well-defined guide to deploy Hadoop using prescriptive deployment information along with valuable performance characterization, enabling you to plan a solution to meet performance needs, while also empowering you to choose the hardware you want.

About Micron

Micron Technology (Nasdaq: MU) is a world leader in innovative memory solutions. Through our global brands—Micron, Crucial® and Ballistix®—our broad portfolio of high-performance memory technologies, including DRAM, NAND and NOR memory, is transforming how the world uses information. Backed by more than 35 years of technology leadership, Micron's memory solutions enable the world's most innovative computing, consumer, enterprise storage, data center, mobile, embedded, and automotive applications. Micron's common stock is traded on the Nasdaq under the MU symbol. To learn more about Micron Technology, Inc., visit micron.com.

About Cloudera

Founded in 2008 by some of Silicon Valley's leading minds from Google, Yahoo!, Oracle, and Facebook, Cloudera believes that open source, open standards are best for business. A 2018 merger with Hortonworks – created in 2011 by members of the original Hadoop team at Yahoo! – strengthens that belief and remains central to their values, continuing their investments in the open source community. Headquartered in Silicon Valley, California, Cloudera has offices around the globe. To learn more about Cloudera, visit cloudera.com.

Appendix A: Configuration Details

HDFS Settings

Mountpoints	hadoop/ssd1, /hadoop/ssd2, /hadoop/ssd3, /hadoop/ssd4, /hadoop/ssd5, /hadoop/ssd6, /hadoop/ssd7, /hadoop/ssd8, /hadoop/ssd9, /hadoop/ssd10, /hadoop/ssd11, /hadoop/ssd12
Failed Disk Tolerance	2
Max Data Transfer Threads	12040

YARN Settings

Memory allocated for all YARN containers on a node	350GB
Minimum Container Size	10GB
Maximum Container Size	350GB (warning can be safely ignored)
Number of Virtual Cores	64
Minimum Container Size (vCores)	16
Yarn.nodemanager.local-dirs	/hadoop/yarn
Yarn.nodemanager.log-dirs	/hadoop/yarn

Spark Settings

Spark2-env	SPARK_EXECUTOR_INSTANCES="28" #Number of workers to start (Default: 2)
	SPARK_EXECUTOR_CORES="56" #Number of cores for the workers (Default: 1).
	SPARK_EXECUTOR_MEMORY="128G" #Memory per Worker (e.g. 1000M, 2G) (Default: 1G)
	SPARK_DRIVER_MEMORY="32G" #Memory for Master (e.g. 1000M, 2G) (Default: 512 Mb)

Benchmark software and workloads used in performance tests may have been optimized for performance on specified components and have been documented here where possible. Performance tests, such as HIBench, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Rev. A 6/19 CCM004-676576390-11334